BMSD 2013

Third International Symposium on

Business Modeling and Software Design

Proceedings

Noordwijkerhout, The Netherlands • 8 - 10 July, 2013

Organized by:

IICREST

In Cooperation with:

SIKS

CTIT Centre for Telematics and Information Technology

ARISTOTLE UNIVERSITY OF THESSALONIKI

AMAKOTA

BMSD 2013

Proceedings of the
Third International Symposium on
Business Modeling and Software Design

Noordwijkerhout, The Netherlands

July 8-10, 2013

Organized by
IICREST - Interdisciplinary Institute for Collaboration and Research on Enterprise Systems and Technology

In Cooperation with

SIKS - the Dutch Research School for Information and Knowledge Systems

CTIT - Center for Telematics and Information Technology

AUTH - Aristotle University of Thessaloniki

AMAKOTA Ltd.

Edited by Boris Shishkov

Graphics Production by Bozhana Yankova

Compiled in Portugal

Printed in Bulgaria

ISBN: 978-989-8565-56-3

Depósito Legal: 359896/13

http://www.is-bmsd.org

secretariat@iicrest.org

BRIEF CONTENTS

KEYNOTE SPEAKERS

Marco Aiello

University of Groningen

The Netherlands

Kecheng Liu

University of Reading

United Kingdom

Leszek Maciaszek

Wroclaw University of Economics

Poland

Chair and Program Committee

Chair

Boris Shishkov, IICREST, Bulgaria

Program Committee

Marco Aiello, University of Groningen, The Netherlands

Mehmet Aksit, University of Twente, The Netherlands

Antonia Albani, University of St. Gallen, Switzerland

Ognyan Andreev, Technical University of Sofia, Bulgaria

Paulo Anita, Delft University of Technology, The Netherlands

Rumen Arnaudov, Technical University of Sofia, Bulgaria

Colin Atkinson, University of Mannheim, Germany

Csaba Boer, TBA, The Netherlands

Boyan Bontchev, Sofia University St. Kliment Ohridski, Bulgaria

Frances Brazier, Delft University of Technology, The Netherlands

Barrett Bryant, University of North Texas, USA

Cinzia Cappiello, Politecnico di Milano, Italy

Jorge Cardoso, University of Coimbra, Portugal

Kuo-Ming Chao, Coventry University, UK

Ruzanna Chitchyan, University of Leicester, UK

Samuel Chong, Capgemini, UK

Dimitar Christozov, American University in Bulgaria - Blagoevgrad, Bulgaria

José Cordeiro, Polytechnic Institute of Setúbal, Portugal

Dumitru Dan Burdescu, University of Craiova, Romania

Joop De Jong, Delft University of Technology, The Netherlands

Jan L. G. Dietz, Delft University of Technology, The Netherlands

Teduh Dirgahayu, Universitas Islam Indonesia, Indonesia

Lyubka Doukovska, Bulgarian Academy of Sciences, Bulgaria

Chiara Francalanci, Politecnico di Milano, Italy

Boris Fritscher, University of Lausanne, Switzerland

J. Paul Gibson, T&MSP - Telecom & Management SudParis, France

PROGRAM COMMITTEE (CONT.)

Eduardo Gonçalves da Silva, University of Twente, The Netherlands

Rafael Gonzalez, Javeriana University, Colombia

Clever Ricardo Guareis de Farias, University of São Paulo, Brazil

Markus Helfert, Dublin City University, Ireland

Philip Huysmans, University of Antwerp, Belgium

Ilian Ilkov, IBM, The Netherlands

Ivan Ivanov, SUNY Empire State College, USA

Dmitry Kan, AlphaSense Inc., Russia

Dimitris Karagiannis, University of Vienna, Austria

Marite Kirikova, Riga Technical University, Latvia

Samuel Kounev, Karlsruhe Institute of Technology, Germany

José Paulo Leal, University of Porto, Portugal

Kecheng Liu, University of Reading, UK

Leszek Maciaszek, Wroclaw University of Economics, Poland

Jelena Marincic, University of Twente, The Netherlands

Michele Missikoff, Institute for Systems Analysis and Computer Science, Italy

Dimitris Mitrakos, Aristotle University of Thessaloniki, Greece

Preslav Nakov, Qatar Computing Research Institute - Qatar Foundation, Qatar

Ricardo Neisse, European Commission Joint Research Center, Italy

Bart Nieuwenhuis, University of Twente, The Netherlands

Selmin Nurcan, University Paris 1 Pantheon Sorbonne, France

Olga Ormandjieva, Concordia University, Canada

Mike Papazoglou, Tilburg University, The Netherlands

Robert Parhonyi, Inter Access, The Netherlands

Marcin Paprzycki, Polish Academy of Sciences, Poland

Oscar Pastor, Universidad Politécnica de Valencia, Spain

Erik Proper, Public Research Centre - Henri Tudor, Luxembourg

Ricardo Queirós, IPP, Portugal

Jolita Ralyte, University of Geneva, Switzerland

Gil Regev, EPFL / Itecor, Switzerland

Ella Roubtsova, Open University, The Netherlands

Irina Rychkova, University Paris 1 Pantheon Sorbonne, France

Shazia Sadiq, University of Queensland, Australia

Program Committee (cont.)

Best Papers Selection

The authors of around ten selected papers presented at BMSD 2013 will be invited by Springer-Verlag to submit revised and extended versions of their papers for publication in a Springer LNBIP Series book.

FOREWORD

This book contains the proceedings of BMSD 2013 – the Third International Symposium on Business Modeling and Software Design, held in Noordwijkerhout, The Netherlands, on July 8 – 10, 2013. The proceedings consists of 33 high-quality research and experience papers that have not been published previously. These papers have undergone a detailed peer-review process and were selected based on rigorous quality standards.

The symposium has been organized and sponsored by the Interdisciplinary Institute for Collaboration and Research on Enterprise Systems and Technology (IICREST), in cooperation with the Dutch Research School for Information and Knowledge Systems (SIKS), the Center for Telematics and Information Technology (CTIT), Aristotle University of Thessaloniki (AUTH), and AMAKOTA Ltd.

The third edition of BMSD has been inspired by two previous very successful editions of the symposium, namely BMSD 2011 (Sofia, Bulgaria) and BMSD 2012 (Geneva, Switzerland). We have all succeeded in establishing and maintaining high scientific quality and stimulating collaborative atmosphere. Touching upon business modeling and its relation to software design, BMSD is characterized by a high level of interaction among leading scientists, engineers, and practitioners in the area. The gap between information systems and underlying business models continues to pose challenges to systems engineers, IT architects, and software developers. The scientific areas of interest to BMSD 2013 are: (i) Business Models and Requirements; (ii) Business Models and Services; (iii) Business Models and Software; (iv) Information Systems Architectures. Each year, a specific theme is chosen, for making presentations and discussions more focused. The theme of BMSD 2013 is: **Enterprise Engineering and Software Generation**.

Business / Enterprise models and architectures are in the focus of a great number of researchers and practitioners not only because enterprise complexity increases but also because it is becoming more and more evident that information systems development failures would be inevitable without adequate underlying business models. Within the last 15 years, numerous research activities have been devoted to bringing together Business Modeling and Software Design – we saw Enterprise Ontology, Model-Driven Engineering, and so on. This impacted the Requirements Engineering discipline. All those efforts have aimed at closing the gap between business and IT systems, acknowledging that not grasping correctly and exhaustively a business system would inevitably lead to consequent software failures. Recently, Service-Orientation + Cloud Computing as well as Autonomic Computing + Context-Aware Computing have appeared as promising paradigms in this regard. Nevertheless, the software development Community still misses a practically usable and widely accepted business-IT alignment methodology that is theoretically rooted. BMSD 2013 has addressed this and other related challenges, by considering a large number of research topics: from more conceptual ones, such as enterprise engineering and business - IT alignment, business process (workflow) modeling and modeling languages, business process verification and execution, maturity models, value modeling, normalized enterprise systems, intelligent systems, ontology reasoning and Semiotics, to more technical ones, such as software specification, use cases, (web) services - choreography modeling and variability issues, database clusters, and 'e-applications' (in Healthcare and Transport), from more business-oriented ones, such as enterprise architecture management, enterprise resource planning, and requirements specification, key performance indicators to IT architectures –related topics. We believe that all these research contributions highlight challenging (technical) problems and present innovative solutions relevant to the above-mentioned scientific areas.

The 33 published papers (including several Invited Papers) were selected from 56 submissions (compared to 46 submissions in 2012 and 58 submissions in 2011) and 13 of these papers were selected for a 30-minutes oral presentation (Full Papers); in addition, 20 papers were selected for a 20-minutes oral presentation (Short Papers and Special Sessions Papers). Hence, the full-paper acceptance ratio of 23% shows a high level of quality which we intend to maintain and reinforce in the following editions of the symposium. Further, the BMSD'13 authors are from: Belgium, Bulgaria, Finland, Germany, Japan, Luxembourg, The Netherlands, Norway, Russia, Saudi Arabia, Switzerland, Thailand, Turkey, and United Kingdom (listed alphabetically); this makes 14 countries in total (compared to 11 countries having been represented in 2012 and 10 countries – in 2011); 7 countries, nevertheless, have been represented in all 3 BMSD editions so far, these are: Belgium, Bulgaria, Germany, The Netherlands, Switzerland, Russia, and United Kingdom. This clearly indicates for a strong European influence which we aim at maintaining and developing further while at the same time adding more countries and regions to the 'BMSD Map'.

The current proceedings' Publisher is SCITEPRESS and we deliver not only printed proceedings but also an electronic version of the proceedings - all presented papers will be made available at the SCITEPRESS Digital Library by September, 2013. Furthermore, the proceedings will be submitted for indexation by DBLP (Computer Science Bibliography). Finally, the authors of around ten selected papers presented at BMSD 2013 will be invited by Springer-Verlag to submit revised and extended versions of their papers for publication in a Springer LNBIP (Lecture Notes in Business Information Processing) Series book.

The high quality of the BMSD 2013 program is enhanced by three Keynote Lectures, delivered by distinguished guests who are renowned experts in their fields, including (alphabetically): Marco Aiello (University of Groningen, The Netherlands), Kecheng Liu (University of Reading, United Kingdom), and Leszek Maciaszek (Wroclaw University of Economics, Poland). In addition, the Keynote Speakers and other BMSD'13 participants will take part in a panel discussion and also in informal discussions focused on community building and project acquisition. These high points in the symposium program would definitely contribute to maintaining the event's high quality and its stable and motivated Community; probably it would be interesting mentioning that two thirds of those who attended BMSD 2012 are among the BMSD'13 participants.

Building an interesting and successful program for the symposium required the dedicated efforts of many people. Firstly, we must thank the Authors, whose research and development achievements are recorded here. Secondly, the Program Committee members each deserve credit for the diligent and rigorous peer-reviewing and paper selection process. Further, we appreciate the willingness of SCITEPRESS to publish the current proceedings and we turn especially to Vitor Pedrosa with sincere words of gratitude for his brilliant collaboration and perfect work with regard to the proceedings preparation. We would also like to compliment the excellent organization provided by the IICREST team; it did all necessary preparations for a stimulating and productive event, supported by its logistics partner, AMAKOTA Ltd. Last but not least, we thank the Keynote Speakers for their invaluable contribution and for taking the time to synthesize and deliver their talks.

We wish you all an inspiring symposium and an enjoyable stay in the beautiful Noordwijk area in The Netherlands. We look forward to seeing you next year in Luxembourg, for the Fourth International Symposium on Business Modeling and Software Design (BMSD 2014), details of which will be made available at http://www.is-bmsd.org.

Boris Shishkov
IICREST, Bulgaria

CONTENTS

KEYNOTE SPEAKERS

FULL PAPERS

SHORT PAPERS

SPECIAL SESSION ON E-HEALTH SERVICES AND TECHNOLOGIES (EHST)

SPECIAL SESSION ON FUTURE INTERNET APPLICATIONS FOR TRAFFIC SURVEILLANCE AND MANAGEMENT (FIATS-M)

SPECIAL SESSION ON INTELLIGENT SYSTEMS AND BUSINESS ANALYSIS

KEYNOTE SPEAKERS

Smartness and the Power Grid: An Information Systems Perspective

Marco Aiello
University of Groningen, Groningen, The Netherlands
M.Aiello@rug.nl

Abstract: The Smart Power Grid promises to not only provide for a more reliable distribution infrastructure, but also give the end-users better pricing, information, and freedom. The promise is fueled by a pervasive digitalization of the energy production and distribution network that will finally involve both utilities, governments, and end-users. The real advantages of the smart grid will be available to all, only if the physical infrastructure of energy distribution is supported by adequate information systems. In this talk, I will review the current state and possible evolutions of the concept of a smart grid, I will point to the data that future information systems will need to manage and, finally, indicate possible uses for such information.

BRIEF BIOGRAPHY

Marco Aiello is Full Professor of Distributed Systems at the Johann Bernoulli Institute for Mathematics and Computer Science of the University of Groningen, The Netherlands. Member of Energy Academy Europe, he focuses on ICT research related to Smart Grids and in particular on the low voltage distribution network and on buildings connected to it. He is the Technical Manager of the EU project GreenerBuildings and PI of the Dutch Energy Smart Offices. He was also the PI of a Dutch project on variability modeling and service orientation for governmental business processes (SaS-LeG). Prior to joining the University of Groningen he was a Lise Meitner fellow at the Technical University of Vienna (Austria) from which he obtained the habilitation. Before he has been assistant professor at the University of Trento (Italy). He holds a Ph.D. from the University of Amsterdam (2002) and a degree in Computer Engineering cum Laude from the University of Rome, La Sapienza (1997). More information is available at www.cs.rug.nl/~aiellom and www.distributedsystems.nl.

Organisational Semiotics for Co-Design of Business and IT Systems

Kecheng Liu
University of Reading, Reading, UK
K.Liu@reading.ac.uk

Abstract: There is often a tension between IT and business systems, because of the changing business requirements. An IT system, that at one time may be highly supportive, after some time, could become inadequate to business operations and be regarded as a legacy system. Such a problem may be caused by a number of factors. One is that IT systems and business processes are not treated as one integral unit; and therefore the misalignment between the business and IT systems may occur. Calibration and alignment of IT and business systems have to be regularly performed. Much effort in industry and academia has been made in searching for solutions, through investigation of, e.g., flexible architecture of IT systems, evolutionary information systems and co-evolution of IT systems and business processes. But the results have often been far from being satisfactory. The co-design of business and IT systems introduced in the keynote is an approach towards this direction. An IT system is viewed as a part of the business organisation, and the design of the business organisation will derive the design of IT system, with the IT system design being a by-product. The organic integration of IT into the business processes will allow both systems to evolve naturally. The co-design approach from the perspective of organisational semiotics in this talk will be presented as a methodological foundation for modelling the business organisation. An organisation is analysed and modelled as the informal, formal and automated components which interact and support each other. Modelling the organisation will involve the solicitation and representation of the norms that govern the behaviour in each part. The presentation of this method of co-design will be followed by an illustration of the method applied in integrating the healthcare enterprise, with a discussion on future research.

BRIEF BIOGRAPHY

Professor Kecheng Liu, Fellow of the British Computer Society, holds a professorial chair of Applied Informatics at the University of Reading, UK. He is Director of Informatics Research Centre, and Head of School of Business Informatics, Systems and Accounting, a constituent school within Henley Business School. He has published 14 books and over 200 papers in the fields of business informatics. As a key international figure in business informatics and organisational semiotics, he was one of the founders and also the chair of a series of international workshops and conferences on Informatics and Semiotics in Organisations. His research interests span from information systems methodology, requirements engineering, pervasive informatics, intelligent systems enabling sustainable working and living, information management in healthcare and, recently, pragmatic web. He has supervised 50 PhD students spreading in many countries and regions such as Chile, Brazil, The Netherlands, Portugal, Saudi Arab, Iran, Singapore, Hong Kong, China and the UK. He has been visiting professor in a number of prestigious Chinese Universities, including Renmin, Beijing Institute of Technology, and Shanghai University of Finance and Economics.

Dependency Analysis for Developing Maintainable Systems – Hierarchy is Not Old Hat

Leszek Maciaszek
Wroclaw University of Economics, Wroclaw, Poland
Leszek.Maciaszek@ue.wroc.pl

Abstract: Complexity is defined as the degree to which a software system is difficult to understand, maintain and evolve. The main difficulty stems from complex interactions (dependencies) between system components/services. The dependencies can be enforced in the architectural design and can be managed by analyzing the implementation code. This keynote addresses software complexity, offers a meta-architecture that minimizes software dependencies, and presents a method to monitor software dependencies. The aim is to produce systems possessing the quality of maintainability. Architectural intent for maintainable systems is invariably some sort of hierarchical layered structure. The holon abstraction - introduced by Arthur Koestler to interpret the structures and processes in living systems - is applied as an approach to restraining software complexity.

BRIEF BIOGRAPHY

Leszek A. Maciaszek is the Director of Institute of Business Informatics and Head of Department of MIS Engineering at Wroclaw University of Economics. He holds also an Honorary Research Fellow position at Macquarie University - Sydney, Australia. He is internationally recognized mostly for his work in database technology, software engineering and systems analysis and design. He has worked as a Visiting Professor/Scientist in more than 20 universities/research centres in countries on four continents; has authored about 150 peer-reviewed publications (including Prentice-Hall and Addison-Wesley books, some translated from English to Chinese, Russian and Italian); has initiated a number of yearly international conferences, including ENASE and FedCSIS; has served as an expert, reviewer and advisor to international corporations, government bodies and ministries (currently a member an advisory council to the Minister of Administration and Digitization of the Republic of Poland); has been a reviewer and evaluator to European Commission of FP7 projects.

FULL PAPERS

Challenges of Modelling Landscapes
Pragmatics Swept Under the Carpet?

Marija Bjeković and Hend*erik* A. Proper
Public Research Centre Henri Tudor, Luxembourg, Luxembourg
Radboud University Nijmegen, Nijmegen, The Netherlands
EE-Team[1]*, Luxembourg, Luxembourg*
{*marija.bjekovic, erik.proper*}*@tudor.lu*

Keywords: Model, Modelling Language, Model Integration.

Abstract: In enterprise modelling, a wide range of models and languages is used to support different purposes. If left uncontrolled, this can easily result in a fragmented perspective on the enterprise, its processes and IT support. On its turn, this negatively affects traceability, the ability to do crosscutting analysis, and the overall coherence of models. Different strategies are suggested to achieve model integration. They mainly address syntactic-semantics aspects of models/languages, and only to a limited extent their pragmatics. In actual use, the 'standardising' and 'integrating' effects of traditional approaches (e.g. UML, ArchiMate) erodes. This is typically manifested by the emergence of local 'dialects', 'light weight versions', as well as extensions of the standard to cover 'missing aspects'. This paper aims to create more awareness of the factors that are at play when creating integrated modelling landscapes. Relying on our ongoing research, we develop a fundamental understanding of the driving forces and challenges related to modelling and linguitic variety within modelling landscapes. In particular, the paper discusses the effect of *a priori* fixed languages in modelling and model integration efforts, and argues that they bring about the risk of neglecting the pragmatic richness needed across practical modelling situations.

1 INTRODUCTION

Enterprise models play an important role in the design and operations of enterprises (Bubenko et al., 2010). More specifically, enterprise models can be used to study the current state of an enterprise, analyse problems with regard to the current situation, sketch potential future scenarios, design future states of the enterprise, communicate with stakeholders, manage change, etc. ((Davies et al., 2006), (Bubenko et al., 2010), (Anaby-Tavor et al., 2010)).

Next to the fact that enterprise models are created for different purposes, it is necessary to do so from different perspectives, such as business processes, value exchanges, products and services, information systems, etc. In the field of information systems engineering, the use of a multi-perspective approach has long since been advocated, e.g. (Wood–Harper et al., 1985), (Zachman, 1987). For enterprise modelling, there is even a broader set of perspectives to consider ((Frank, 2002), (Winter and Fischer, 2007), (Greefhorst et al., 2006), (Wagter et al., 2012)).

The collection of models that jointly represent the different perspectives of one enterprise, are often expressed using different modelling languages, including UML (OMG, 2003), BPMN (OMG, 2008), ArchiMate (Iacob et al., 2012), i* (Yu and Mylopoulos, 1996), e3Value (Gordijn and Akkermans, 2003), SBVR (OMG, 2006), etc. Throughout this paper, we will use the term *enterprise modelling landscape*, or simply *modelling landscape*, to refer to the variety of models and corresponding modelling languages used in a specific enterprise modelling effort[2].

Since the models included in an enterprise modelling landscape provide different views on the *same* enterprise, it is quite natural to expect that the sets of models form a coherent whole; i.e. linked where relevant and consistent as a whole. A plethora of models

[1] The Enterprise Engineering Team (EE-Team) is a collaboration between Public Research Centre Henri Tudor, Radboud University Nijmegen and HAN University of Applied Sciences (`www.ee-team.eu`).

[2] The scope of a particular enterprise modelling effort can be only one project, cross-project considerations, entire enterprise etc.

and modelling languages may easily result in a fragmented perspectives on the enterprise, which is likely to have a negative impact on the traceability across different models, the ability to do crosscutting analysis, to manage inconsistency[3] and to ensure overall coherence of e.g. the design of the enterprise. The fact that different models are usually expressed in different modelling languages makes it even harder to maintain the coherence.

The traditional approach of dealing with fragmentation in modelling landscapes is to create an integrated/unified modelling language, such as UML and ArchiMate. However, in actual use, one can observe how the 'standardising' and 'integrating' effect of such languages erodes. This typically manifests itself in terms of local 'dialects', 'light weight versions', or several extensions of an existing standard that are intended to deal with 'missing aspects'. This point is illustrated by the advent of domain or purpose-specific (modelling) languages that allow for the creation of models that are tuned to the needs of specific domains or purposes. At the same time, there exists a number of approaches that aim to alleviate this fragmenting effect by assuring the links between the different languages definitions, see e.g. ((Frank, 2002), (Vernadat, 2002), (Anaya et al., 2010)). Typically, these links are defined based on the standardised definition of the language, in particular its semantics (as in e.g. (Anaya et al., 2010)).

As we will discuss throughout this paper, the drivers of language standardisation are predominantly of technical nature (Hoppenbrouwers, 2003). There are clear benefits of language standardisation. For instance, it is generally considered as a necessary condition for CASE tool development. In addition, it is a first step towards *automating* some model manipulations, e.g. model transformations (including code generation). Nonetheless, the potential benefits of standardised languages tend to be quickly generalised to the entire modelling endeavour. What goes practically unquestioned, in aiming for language standardisation, is whether fixed languages can be used at all in different modelling contexts and with different stakeholder groups. For instance, in ((Kaidalova et al., 2012), (Bubenko et al., 2010)) it is observed that the choice of formalism should be related the given modelling task and audience. For example, when the language chosen is rather too formal for stakeholders, it can hinder the modelling process.

We will argue, that standardising/fixing a modelling language leads to a situation in which the *pragmatic* richness that is needed across various modelling situations in practice is neglected. This brings about the risk of *sweeping pragmatics under the carpet*. Indeed, various surveys ((Malavolta et al., 2012), (Kaidalova et al., 2012)) and empirical studies ((Anaby-Tavor et al., 2010), (Karlsen, 2011), (Briand et al., 1995), (Elahi et al., 2008)) reporting on practical experiences with enterprise modelling, point at the need for flexibility in modelling. At the same time they observe a lack in flexibility of tools and the underlying (fixed) languages to aptly fit the needs of specific modelling situations. This often leads to different levels of discipline in which the standard language is obeyed to, e.g. resulting in dialect-like variations of the original language ((Bubenko et al., 2010), (Malavolta et al., 2012), (Briand et al., 1995), (Elahi et al., 2008), (Karlsen, 2011)), or workarounds (e.g. using ad hoc notes and annotations) to compensate for the missing elements in the language/tool (Delen et al., 2005). This may even go as far as the use of home-grown, organisation-specific semi-structured models/languages instead of the standard notation ((Anaby-Tavor et al., 2010), (Malavolta et al., 2012), (Karlsen, 2011)).

In our view, this indicates a lack of fundamental understanding of the role of language in modelling, and more specifically, the place of fixed language in attempts to integrate models. In this paper, based on our ongoing research, we intend to shed more light on this topic. We will therefore start in Section 2 with a discussion on the potential benefits of standardising modelling languages. In Sections 3 and 4 we then explore the effects of their use in modelling and integration respectively, also identifying more explicitly the risk of *sweeping pragmatics under the carpet* of the modelling landscape. We then continue in Section 5 with a fundamental discussion of *models* and *modelling*. This understanding is used then to discuss in Section 6 the role of modelling languages, and their potential standardisation. In the conclusion, we synthesise the insights and suggest the direction to explore more realistic strategies for creating and managing modelling landscapes.

2 STANDARDISATION

In our field, we typically deal with *linguistic models* (Karagiannis and Höfferer, 2006), i.e. models expressed in a modelling language. A further distinction can be made between textual and graphical languages (Harel and Rumpe, 2004). Given their significant usage in enterprise modelling efforts, our discussion focusses on graphical languages. Traditionally, a

[3] Allow inconsistencies between models, e.g. due to different views by differing stakeholders, in a controlled way.

modelling language is defined in terms of an *abstract syntax*, a *concrete syntax* and *semantics*.

The *abstract syntax* defines the basic elements and rules for creating models. The abstract syntax of graphical modelling language is commonly given by means of the meta-model. The meta-model actually represents the *conceptual foundation* of the modelling language, i.e. a specific classification of concepts to be used in discourse about the 'world' (Falkenberg et al., 1998). As such, the meta-model provides a particular *ontological position* filtering the view on the 'world' that one chooses to take (Falkenberg et al., 1998). It is also argued in (Falkenberg et al., 1998) that all other aspects of the modelling language depend on the concepts contained in the meta-model.

The *concrete syntax* or *notation* deals with the (visual) representation of the modelling language on the medium, by defining the visual symbols and rules for their combination (and their correspondence to the abstract syntax of the language. The medium itself can for example be restricted to a specific form, such as graphical, textual, or video, but the notation in general can also be restricted in terms of fonts, icons and layout rules. See e.g. (Moody, 2009).

The *semantics* deals with the meaning of a modelling language. The conventional way of defining semantics is in terms of a *semantic domain* and a *semantic mapping* (Harel and Rumpe, 2004). According to (Harel and Rumpe, 2004), the semantic domain captures the "*decisions about the kinds of things language should express*" (Harel and Rumpe, 2004, p. 68). The semantic mapping, in turn, establishes the correspondence from syntactic elements to the semantic domain. However, this approach to defining semantics is required for the mechanical manipulation of models, e.g. by computer tools, since they can only manipulate semantics in terms of syntactic representations (Harel and Rumpe, 2004). We propose to label this aspect of semantics as *syntactic semantics*. In the realm of human use of language, *meaning* is approached differently by taking into account the entire context in which the linguistic communication is embedded and the *function* of linguistic utterances in that context[4]. We propose to label this aspect of semantics as *pragmatic semantics*. When stakeholders use modelling language in modelling, they address the semantics from this perspective. We will discuss this topic further in Section 6.

As illustrated in Figure 1, the fixed i.e. *standardised* definition of the modelling language *a priori* identifies and restricts the *intended* sets of models the

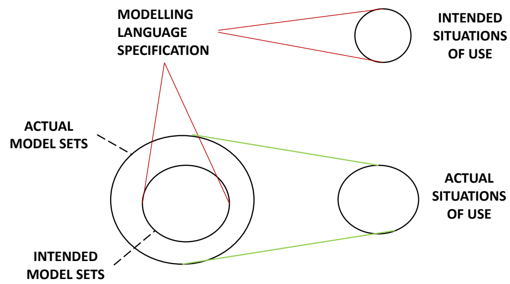


Figure 1: Intended and actual use of the modelling language.

language allows to express. This also limits the sets of *intended* modelling situations in which, by using a particular fixed language, a satisfactory model of the domain can be produced. Therefore, when *freezing languages* (Hoppenbrouwers, 2003), the designers of the language, implicitly or explicitly, restrict the intended use of a language. The more models a language can 'produce', the more *expressive* the language is. However, the actual *suitability* of the language is dependent on the particular modelling situation in which language is *used*. It refers to whether the modelling language allows to create the model of the domain such that it *satisfies* the needs of actual modelling situation (e.g. the level of detail in domain, coverage of specific aspects, specific form etc.). This is the area of *modelling pragmatics*. According to (Thalheim, 2012), it studies the use of languages in a particular modelling situation depending on the purposes and goals of models within a community of practice.

In most cases, graphical modelling languages are defined semi-formally, i.e. with explicit (and more or less formal) definitions of the (abstract and concrete) syntax. The conceptual foundation of the language may be defined at different levels of genericity (i.e. involving more or less generic concepts). This influences the intended model sets supported by the language. Also, different syntactic restrictions may be included in the language definition, further restricting intended model sets. The semantics is, however, usually given in an informal manner in the language specification, i.e. using natural language. The latter does not lend itself that easily to machine interpretation. Formally defined semantics is required for making the language specification (fully) machine readable. The standard and precise definition of modelling language syntax and semantics is indeed pre-requisite for automation. This makes possible e.g. model transformations, interoperability, computer-aided analysis techniques, simulation, and (developing tools for) various other manipulations of models. These are some clear benefits of fixed languages, and one possible strategy

[4]This way of addressing meaning is inherent to the functional perspective or action tradition on language. We will elaborate this in the Section 6

to ensure a return on modelling effort.

The predominant factor for *a priori* fixing the modelling language is therefore the technology (Hoppenbrouwers, 2003), i.e. the fact that mechanical manipulation of models requires fixed representations. It is also assumed that by having standard (and precise) definition of the modelling language, all the meta-discussions on concepts can be avoided. This would contribute to the certainty and efficiency of communication (Hoppenbrouwers, 2003), and shared understanding of models would be easier to reach.

However, as the next two Sections will discuss, standardisation comes at a cost, in particular for pragmatics, which ends up being swept under the carpet.

3 USE OF FIXED LANGUAGES

A key problem in the use of fixed languages in modelling is rooted in the lack of *suitability* of a language for an actual modelling situation. It is indeed often the case that the choice of the modelling language is imposed from the "outside" onto the modelling situation. Typically, the existing modelling infrastructures within the enterprise, the expertise of the modelling team, etc. constrain the choice of the language.

As reported in several surveys on the practice of modelling, see e.g. (Davies et al., 2006) (Anaby-Tavor et al., 2010) (Malavolta et al., 2012), general-purpose modelling languages are the most widely used modelling languages. Nonetheless, these surveys also indicate that 'variants' of these languages are in place. For instance, several experience reports of the use of i* in specific situations (Briand et al., 1995) (Elahi et al., 2008) explain in detail why and how the language was extended to be able to make models that satisfy the needs of the given modelling situation. In the case of ArchiMate, this has e.g. resulted in the suggestion to distinguish between a 'sketching' and a 'designing' (Lankhorst et al., 2005) variation of the notation (using more sketchy lines and more informal looking fonts). This variation can even be combined in one model to differentiate between the status of different parts of the model. On the same line, (Malavolta et al., 2012) indicate the need for informal 'variants' of (software and enterprise) architecture models for their communication to different stakeholders.

In our view, these '*variants*' are essentially purpose-specific variations of the same original generic modelling language, differing only in their *syntactic and semantic restrictions*, i.e. purpose-specific modelling languages (Bjeković et al., 2012). They emerge from the need to make the language suitable for the communicative task in the actual modelling situation at hand. When the modelling language used is not suitable enough, variations will emerge to compensate for this lack of suitability. Pragmatics re-emerging from under the carpet.

An extreme case of adapting the language to the actual modelling situation is the use of 'home-grown' notations (Anaby-Tavor et al., 2010) and/or emergent modelling languages, i.e. the languages that are being constructed along the modelling process. For instance, in (Anaby-Tavor et al., 2010) business architects express a clear preference for home-grown, semi-structured models, since they offer flexibility in terms of re-factoring, delayed commitment to syntax, and closer fit to the inherent way of thinking in these phases. These semi-structured models emerged through the repeated use in similar modelling situations, whereby the sets of concepts and their meanings, and (right level of) restrictions gradually yielded a new language.

Collaborative modelling situations also demonstrate the challenge of adequate modelling support. For instance, in situations whose primary goal is collective knowledge creation, e.g. *developing vision and strategy, scoping the problem, and high-level business design*, the need for simple and intuitive modelling notations, as well as unconstrained medium (e.g. plastic walls, whiteboards) prevails (Bubenko et al., 2010). As most stakeholders do not have modelling expertise, the language and tools have to accommodate this, and thus are required to be simple, intuitive, and corresponding to the natural interaction that occurs in such situations (Barjis, 2009).

Depending on the nature of a modelling situation, the modelling language is, to a greater or lesser degree, able to support the formulation of the desired models. We have discussed a number of different strategies used in practice to compensate for the lack of language suitability. These strategies in one way or the other act on the language specification, aiming to 'extend' the actual sets of models which a given language can express. In doing so, there is inevitably the risk of violating the intended pragmatics of the fixed language. However, such a practice may well be an indication that the pragmatic richness of modelling situations to be supported by the language has not been adequately taken into account when the language was designed. In our view, answering this dilemma requires a more fundamental understanding of the role of language in modelling.

4 MODEL INTEGRATION

Since enterprises are modelled using different models/views, it is desirable to maintain their coherence. The use of a wide range of models and modelling languages can easily lead to a fragmentation of the modelling landscape; i.e. a break up of coherence. To avoid or deal with such a situation, different strategies are employed, e.g. ((Frank, 2002), (Iacob et al., 2012), (Anaya et al., 2010)). We classify them into *language unification* and *language federation* strategy. They both address the integration challenge at the level of fixed language definition. In this section, we discuss some of the challenges that such an approach raises.

The strategy of *unification* of enterprise modelling language(s) has the ambition to define a standard set of constructs in which all the models (i.e. perspectives) can be expressed. The unified language is intended to be used *instead of* the different languages that partially cover the domain of interest. We can observe this logic in the definition of UML, ArchiMate, as well as language of EKD method (Stirna and Persson, 2012). This approach boils down to preventing the fragmentation from occurring in the first place.

The unified language offers a fixed, but integrated, view on some domain of interest. Besides a priori fixing the set of constructs, the standardising effect of the unifying language lies also in the fact that it *a priori* fixes the perspectives for modelling some domain. The integration between the different perspectives, i.e. models, is easier to ensure, given that consistency and coherence rules can be embedded in the language definition. The CASE tool can then *automatically check* these properties.

Regretfully, however, it is nearly impossible to a priori identify which perspectives should be part of an integrated language. The challenge lies in the fact that the relevance of different perspectives (and its related modelling concepts) is highly context-dependent. For instance, different perspectives may be relevant for different (types of) enterprises, or even in different transformation projects of the same enterprise. Moreover, over time, new perspectives may become relevant for a particular enterprise.

To cater for context-dependency, standard modelling languages like UML and ArchiMate offer means for their extension. For instance, UML has the well-known stereotyping mechanism (whose problems are also well-known). In the case of ArchiMate, the very design of the language (Lankhorst et al., 2010) provides different possibilities for extension. However, these mechanisms are of limited scope, they mostly allow for refinement of concepts, not for adding the entire domains which were not envisaged by the original language definition.

At the same time, one can observe how there is a drive for the ArchiMate language as a *standard* to be extended with additional domains. The move from the ArchiMate 1.0 standard to the ArchiMate 2.0 standard included two additional domains, namely motivation and migration. Further integration between TOGAF and ArchiMate is likely to lead to even more extensions. Moreover, the extensions with e.g. business policies and rules, are also considered (Iacob et al., 2012). *Where will it stop?*

The fact that such unified languages are typically very generic is already an indicator that the perspectives and concepts that are specific to different contexts cannot be covered. Therefore, in the actual use of a unified modelling language in the specific context, the need to *extend* the language with 'missing domains' is likely to emerge. On the other hand, defining the comprehensive and overly detailed language covering all the potentially relevant perspectives and related concepts would most likely result in the overly complex language that would be costly to use.

According to (Egyedi, 2007), this tension is inherent to any standard definition process, including modelling standards. The authors argue that defining the context-independent standards (e.g. enterprise- or application-independent, etc.) typically leads to very comprehensive and/or generic standards, therefore also difficult or too expensive to use. Even more, this tension is recognised as a fundamental dilemma in developing standards, which is very difficult to resolve (Egyedi, 2007).

Moreover there is another issue with a unified language approach. A common belief is that by means of *a priori* imposing standardised vocabulary, frequent meta-discussions could be avoided, and the knowledge transfer could be facilitated. However, as well as with perspectives, the languages used in enterprises are also context-dependent, depending on e.g. professional background and education of stakeholders. These factors exert an influence on the default way different people conceptualise (Linden et al., 2012). In that sense, imposing another language, which is 'outside' their area of practice, straight from the beginning is likely to cause, and not resolve, conceptual misunderstandings.

A language such as ArchiMate is designed to deal with this issue by enabling users to define their own *viewpoints*, i.e. to have their model (the *view*) derived from the integrated model. However, this viewpoint is still to be defined from the unique fixed 'footprint', i.e. unified metamodel.

Conversely to the key drive of *language unification*, the rationale of the strategy of *federating languages* is not to prevent, but rather to allow and manage the variety of modelling languages within the modelling landscapes. To avoid fragmentation, it aims to ensure the links between the modelling languages. A number of different approaches exists to establish these links. We can classify them on several dimensions, for the goal of this discussion.

One classification dimension regards the way in which bridges are constructed between the languages. One approach is to directly connect the languages on a point-to-point basis, e.g. (Bézivin et al., 2006) (Zivkovic et al., 2007) (Fabro and Valduriez, 2009). The links can also be established with the help of a 'mediator', as in Unified Enterprise Modelling Language (UEML) (Anaya et al., 2010). The mediator's role is to serve as the basis to which all the modelling languages are mapped, and to ensure the consistency between the models. In UEML, the unified 'ontology' (Opdahl et al., 2012) plays this role. However, it requires the languages to be (re)defined in accordance with the specific grammar, as well as in full formal precision, given that the approach targets the semantic interoperability of tools and associated languages.

We may also distinguish the approaches based on the moment (in the language's lifetime) when the links are established, i.e. based on the *temporality*. In the case of point-to-point bridges, the links are established between already existing languages. However, the links may also be defined at the moment of language design. In that case, the new language is defined extending or specialising an already existing (more generic) language, the approach akin to the older work on the so-called meta-model hierarchies (Falkenberg and Oei, 1994) (Falkenberg et al., 1998). For instance, (Vallecillo, 2010) discuses in detail different techniques for combining (domain-specific) modelling languages based on this hierarchy logic.

Another dimension of classification may be whether the links are established based on syntactic correspondences only, or whether they also involve semantic correspondences. MDE approaches to model transformations and model weaving (Fabro and Valduriez, 2009) generally only consider the syntactical level. In turn, semantic integration is explored in order to enhance the quality and reduce the complexity of *a priori* establishing language bridges (Karagiannis and Höfferer, 2006). It is however questionable what *kind* of semantics of language constructs can be captured *a priori*. This is especially pronounced for generic languages such as i* (Yu and Mylopoulos, 1996) or UML, but also relevant for domain-specific languages, as discussed in (Frank, 2011). Without taking the context of language use, the abstract concepts (underlying generic languages) may allow for various interpretations. The precise, i.e. *contextualised* interpretation is, however, necessary for the bridges to be meaningful. For instance, a modelling language such as i* (Yu and Mylopoulos, 1996) can be used for modelling strategic goals of actors in relation to the system, but also to express information systems requirements. In each of these contexts, the *inherent semantics* of e.g. the modelling construct *actor* will vary: when modelling strategic goals of enterprise, *actor* can only be a human actor, while in the context of modelling software requirements, a machine may be an *actor* as well. Without taking this into account, the a priori mappings between i* and ArchiMate (based on language definition and not use) might be meaningless in some of the contexts of i* use.

This context-dependent semantic variation can only be determined by taking into account the context of language use, not *a priori*. Indeed, in discussing the meaning of models, (Thalheim, 2012) distinguishes its two complementary aspects: referential and functional meaning. While the referential meaning is well investigated, the functional meaning relates model elements with the context in which they are used. The key to understanding the functional meaning is in modelling pragmatics.

In addition, as we have seen in the reports of the use of modelling languages, there is an indication that, in various usage contexts, the syntactical variation with respect to the original language specification also takes place. Therefore, the value of building the language bridges *a priori* and based on a priori fixed languages, i.e. out of the context of its actual use, might be questioned.

5 WHAT IS MODELLING?

The remainder of the paper discusses our fundamental understanding of the driving forces and challenges related to modelling and linguistic variety within enterprise modelling landscapes. In our ongoing research, we develop a theory to explain why and how enterprise modelling landscapes emerge and evolve, where we focus on the *use of modelling languages*. Based on such a theory, our ambition is to revisit the integration strategies and propose their realistic variants that better caters for the pragmatics of models/languages. Our view on models and modelling is rooted in semiotics, linguistics and cognitive science. This view is inspired by different related research tackling the fundamental modelling aspects such as ((Stachowiak, 1973), (Rothenberg, 1989), (Falkenberg et al., 1998), (Hoppenbrouwers et al., 2005), (Proper et al., 2005), (Thalheim, 2012)). We look at the models as being essentially a means of communication about some domain of interest, and the process of modelling as a communication-driven process led by a pragmatic focus (Hoppenbrouwers and Wilmont, 2010).

Though different views on models and modelling exist, as well as many different definitions, we will elaborate the reasons for which we propose the following (general) definition of model (based on ((Stachowiak, 1973), (Rothenberg, 1989), (Falkenberg et al., 1998), (Thalheim, 2011))):

> *A **model** is an artefact acknowledged by an observer as representing some domain for a particular purpose.*

By stating that a model is an **artefact**, we exclude from the definition the conceptions (Falkenberg et al., 1998) or so-called "mental models" (mental spaces in (Fauconnier, 2010)). Nonetheless, we do consider conceptions as important within the modelling process. Later in this section, we elaborate on their role and importance, especially in the case of collaborative modelling.

With **observer** we refer to the group of people creating (i.e. model creators) and using the model (i.e. model audience). On one extreme, it can refer to the entire society, on the other extreme, to the individual. Though it may not be the general rule, in enterprise modelling context, it is very often the case that model creators are at the same time its audience. The observer is the key element in modelling, as it is only through the appreciation of the observer that some artefact is acknowledged as being the model.

Similarly to (Falkenberg et al., 1998), we define **domain** as any part "part" or "aspect" of the "world" *considered relevant by the observer* in the given modelling context. The "world" here may refer not only to the "real" world, but also to hypothetical or imagined worlds. Even more, the domain of a model can be another model as well.

A model always has a purpose. This *purposefulness* dimension is explicitly present in most of the model definitions, e.g. ((Stachowiak, 1973), (Rothenberg, 1989), (Thalheim, 2011)). Although acknowledged as an essential dimension of models, the concept of purpose is rarely defined and its role in the entire modelling process is scantly discussed.

We see the **purpose** of the model as a combination of the following dimensions: (1) the *domain* which the model should pertain to and (2) the intended *usage* of the model (e.g. analysis, sketching, contracting, execution, etc.) by its intended *audience*. In line with (Rothenberg, 1989), (Thalheim, 2011), we argue that, although usually implicitly present in modelling, the purpose should be explicit within the modelling process; i.e. the model creator should be aware of the intended usage and audience of the model. This is quite important, since the fitness-for-purpose directly determines/influences the degree to which the models satisfies the conception of the domain within the actual modelling situation. This subsequently influences the value of the model for its intended usage.

As illustrated in Figure 2, when modelling, the observer **O** decides[5] what the relevant aspects of the "world" under consideration are in the given modelling situation. This results in the *conception of* the domain, $\mathbf{c}_d$. This process of abstracting away from certain aspects of the "world" which are not relevant should be driven by the purpose **p** of the model **m** ((Rothenberg, 1989), (Thalheim, 2011)) (depicted as influence **1** of the purpose **p** on the relation *conception of*, see Figure 2).

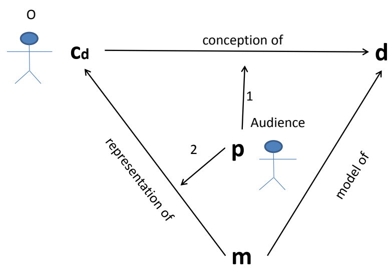


Figure 2: The process of modelling.

The observer subsequently tries to shape an artefact (i.e. the model-to-be) in such a way that it adequately represents, for the purpose **p**, his/her conception of the domain $\mathbf{c}_d$. The purpose **p** is a conception as well, i.e. the conception of the purpose of the model-to-be $\mathbf{c}_p$. Even more, the observer **O** also has the conception of the model-to-be, $\mathbf{c}_m$. The modelling process actually consists in the observer's gradual alignment of these three conceptions (illustrated in Figure 3). This process usually takes place in parallel with the very shaping of the model artefact. When their mutual alignment is achieved, the artefact is acknowledged as the *representation of* the (conception of the) domain **d** for the purpose **p**. In other words, the observer **O** *acknowledges* that the artefact **m** is a *model of* the domain **d** for the purpose **p**.

[5]Obviously, the observer's judgement may be influenced by many different factors, e.g. observer's intentions, experience, previous knowledge, etc. We exclude from our consideration the potential political intentions of the observer.

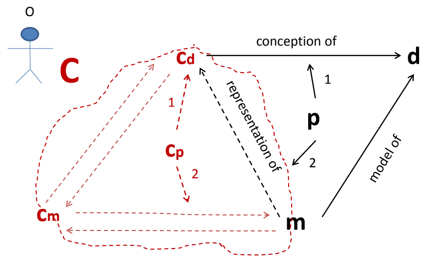


Figure 3: Aligning conceptions in modelling.

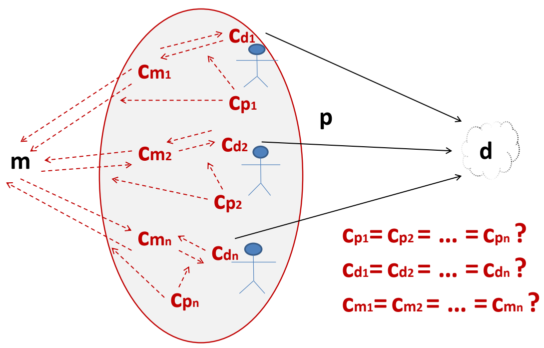


Figure 4: Co-aligning conceptions in modelling.

The previous explanation holds for a single observer modelling process. But what does happen in a collaborative modelling process? In such a situation, a group of **n** human actors is involved in the process of modelling, and is supposed to *jointly* observe some domain and come up with its model, for some purpose. The great challenge in the collaborative modelling consists in the fact that each participant has its own conception of the domain, of the model taking shape and of the purpose of that model. This is illustrated in Figure 4. In addition, these individual conceptions are influenced by the individual pre-conceptions (Proper et al., 2005), brought by the particular social, cultural, educational and professional background. So, in order to reach shared understanding and appreciation of the artefact **m** as a *common* model of the domain **d** for the purpose **p**, the co-alignment of the $\mathbf{n}\times\mathbf{3}$ conceptions has to take place. Indeed, this co-alignment of conceptions is generally considered as a critical step in the collaborative modelling, where all the discussions, negotiations and agreement about the model has to take place.

We have seen how, in the modelling process, models gradually come into being by (co-)aligning the different conceptions in the observer's mind(s), and by their subsequent externalisation. To externalise the (aligned) conception of the domain $\mathbf{c}_d$ into **m**, the observer **O** uses some system of symbols/signs. Essentially, the observer establishes the mapping between the conception into some system of signs, whereby signs used come to *represent* the observer's conception of the domain $\mathbf{c}_d$. The models externalised using some system of symbols are usually referred to as *symbolic models*.

We will now look more closely at the role of the modelling language, as well as the factors contributing to the modelling and linguistic variety in the modelling landscapes.

6 MODELLING LANGUAGES

6.1 Role of Modelling Languages

A language *in use* may be regarded as a *medium system* (Hoppenbrouwers, 2003), involving both a *language* and a *medium*. The medium refers to the physical means to achieve communication (Hoppenbrouwers, 2003), e.g. audio, video, writing, etc. We entertain this view as it allows us to discuss, from a fundamental perspective, different and often conflicting requirements put on modelling languages. We thus propose to consider that a modelling language has two primary roles, *the role as a language* to be used by humans, and *the role as a medium*, i.e. carrier of sets of models aimed at mechanical manipulation. Fundamentally, we regard language as an instrument of human activity, primary in support of reflection and communication. This is in line with a functional perspective (Cruse, 2011) and the action tradition on language (Clark, 1993). We are thus primarily interested in how fixed modelling languages play that role.

In its *role as a language*, it should serve as a support of activities taking place in the modelling process. The central issue therefore is to which extent an a priori fixed language can act as an effective means of human communication and knowledge creation in the actual modelling situation. Let us look closely at its adequacy for creating conceptions. As shown in Figure 5, if model **m** is expressed in a modelling language $\mathbf{L}_m$, what is the language $\mathbf{L}_c$ in which the conception $\mathbf{c}_m$ is constructed in the observer's mind: is it the modelling language $\mathbf{L}_m$, or some other language, e.g. the observer's native language?

Obviously, the definitive answer to this question is not easy to provide. Nevertheless, we believe that it is important to create awareness of the potential gap between $\mathbf{L}_m$ and $\mathbf{L}_c$ and its consequences. As previously discussed, a fixed modelling language comes with an a priori embedded filter on the 'world'. In the modelling process, it thus tends to constrain, or at least influence, the conception of a domain. Depending on the actual modelling situation, this pre-conceived fil-

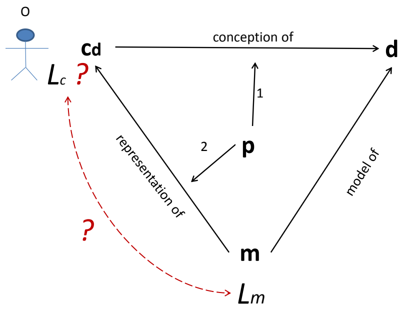


Figure 5: The role of language in the modelling process.

ter may prove to be inadequate for the particular observer and for the particular modelling problem.

For instance, research from cognitive linguistics shows that the entire social, cultural, educational and professional background of an observer plays a role in shaping their 'linguistic personality' (Novodranova, 2009). This includes their 'world view'; i.e. their natural way of conceptualising phenomena in the 'world' (Schmid, 2010). Likewise, (Linden et al., 2012) shows that different people have different default interpretations of the abstract concepts underlying enterprise modelling languages. It is reasonable to assume that, in a particular modelling situation, (at least) non-modelling-experts form conceptions in a language significantly different from $\mathbf{L}_m$. This is likely to increase the $\mathbf{L}_c - \mathbf{L}_m$ gap and negatively impact the general suitability of $\mathbf{L}_m$.

Another factor potentially effecting the $\mathbf{L}_c - \mathbf{L}_m$ gap involves the nature of the modelling task in the particular modelling situation. While the relation between the nature of the modelling problem and the suitability of the modelling language needs further study, the empirical data indicates rather negative influence of overly restrictive (in terms of syntactic-semantic restrictions embedded in) fixed language on the creativity in modelling. For instance, (Anaby-Tavor et al., 2010) observe the inadequacy of fixed modelling languages to support the exploration phases where things are unclear and ambiguous, and models are used to organise information, gain insight, envision alternative possible futures etc. (Anaby-Tavor et al., 2010). Similarly, (Bubenko et al., 2010) observe that in highly creative and collaborative situations such as vision and strategy development, rather informal and intuitive notations (and mediums) seem to be of better support.

In its role as a *medium*, a modelling language should accommodate the formulation of models, while allowing their mechanical manipulation. The potential added value of the modelling language, from this perspective, lies primarily in its re-usability across different modelling problems, and the extent to which the language specification is machine readable.

As discussed in Section 2, the reusability of a language relates to its expressiveness. Obviously, this makes sense for the development of tools and automated model management. However, while general-purpose modelling languages are usually more expressive, they are less suitable than domain or purpose-specific languages for specific problem domains and modelling situations. These languages incorporate in their definition concepts that are tuned to the modelling of particular domains. The overall aim is to foster modelling productivity, facilitate the understanding of the models by the domain stakeholders and increase the overall quality of resulting models, in particular semantic and pragmatic quality (Krogstie et al., 2006). While domain and purpose-specific languages seem to correspond to the natural, i.e. human, need for suitability of the modelling *language*, they are not easy to reuse across different situations. General-purpose modelling languages, on the other hand, are easier to reuse for modelling different domains. Nevertheless, the interpretation challenge of the models expressed in them is more pronounced.

It certainly does not make sense to have situation-specific modelling language emerge from scratch in each new modelling situation. However, it does make sense to embed in the (generic) meta-model the elements that are repeatedly discussed in similar modelling contexts. Therefore, there is a need to carefully balance potential sets of models one would like to express in a language, and potential sets of modelling situations one would like to support.

The second added value of a language as a medium is machine readability. This is driven by the need for automated manipulation of models. This is achieved by formal, i.e. precise and unambiguous, definition of both the syntax and semantics of the language, usually in a mathematical language. This essentially boils down to expressing the semantics of the model/language in terms of another syntactic representation, i.e. expressing the *syntactic semantics*. Though necessary for the machine's correct interpretation of the model, this kind of a priori fixed semantics does not tell anything about what the model means to the observer. In particular, it allows by no means to a priori precisely capture the meaning of a model as it occurs in the actual modelling situation, i.e. the model's *pragmatic semantics*.

6.2 Language Variety

We are now able to suggest two primary drivers for the variety within enterprise modelling landscapes. In our view, these drivers relate to:

Abstraction variety – Abstraction is at the heart of modelling. As we have seen, it boils down to *purposefully* neglecting irrelevant details of the observed "world". The need for differing levels of abstraction in dealing with an enterprise is related to its complexity/multifaceted-ness. The abstraction variety thus leads to the increase in number of needed perspectives, i.e. models, in modelling.
Manifestation variety – The manifestation here refers to the way the model is *represented* on the *medium*. Fundamentally, the need for this kind of variety is rooted in the complexity and heterogeneity of social structures underlying modelling. The more heterogeneity in stakeholder groups, the more likely is the need for different manifestations of arguably the "same" model. This invites an increase in linguistic variety on top of the abstraction variety.
These types of variety are illustrated in Figure 6. Throughout the paper, we have seen that these factors are to a large extent situational, i.e. they depend on the particular enterprise and enterprise modelling effort, involved stakeholders (observer), the purpose for which models are created in this effort, etc. This leads us to the conclusion that modelling landscapes should be *situated*. Indeed, the 'standard eroding' effects discussed in Sections 3 and 4, might be seen as the manifestation of the need to make these landscapes better situated, driven by the *pragmatic needs* of the wider organisational context that the landscapes cover.

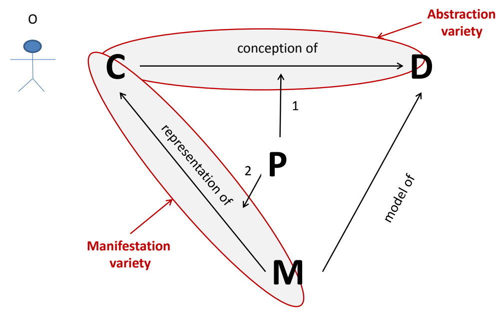


Figure 6: Sources of variety in the modelling landscapes.

Evidently, both of these drivers stem from the *language role* of the modelling language. They also fundamentally conflict with the drivers for the a priori standardisation of the language, which stem from its *medium role*. We believe that this natural polarity deserves careful management, rather than denial. It should by no means be swept under the carpet.

7 CONCLUSION & OUTLOOK

Based on the discussions so far, we posit that, at the heart of the challenge of creating integrated modelling landscapes, lies the question: *What can be a priori fixed in a modelling language?* To create more balanced strategies that cater for the pragmatic needs of modelling landscapes, it is necessary to carefully examine which aspects may be feasible to a priori fix in the modelling language. In this final section, and relying on the presented fundamental view on modelling and language, we draw some initial conclusions as a tentative (though partial) answer to this question.

First of all, the semantics. Given that (the conception of) the *domain* actually does not 'exist'[6] a priori but emerges in the very process of its modelling, the semantics, in the *pragmatic semantics* sense, also cannot be captured a priori. One would expect to be able to at least fix the grammar (abstract syntax) and symbols used for its representation on the medium (concrete syntax). To the extent to which the grammar is tuned to the needs of the intended set of modelling situations, it can be *a priori* fixed. One can also start by only a priori fixing the core grammar, and allowing its further refinement *a posteriori* during the use of the language. For instance, it can be possible to start the modelling process with lightly constrained vocabulary adapted to the domain/purpose (e.g. based on some historical heuristics), and then gradually include more formalisation, to the extent necessary for the intended usage and audience of the model. Needless to say, this would necessitate modelling infrastructures to be more flexible. A growing interest in this subject can indeed be observed, e.g. ((Cho et al., 2011), (Ossher et al., 2009)).

The point we aim to make is that, although having an a priori fixed representation to a large extent facilitates the development of tools and automation of model manipulations, fixing the language that is to be used in human communication may seriously damage its capacity to adequately express thoughts, i.e. conceptions of domains in the given modelling situation. Even if carefully defined, the standardised enterprise modelling language will inevitably demonstrate the need to evolve, to adapt to the dynamically changing

[6]The term *exist* is used here in the sense of Heidegger's notion of *breaking down*, discussed in (Winograd and Flores, 1986). Indeed, "*Heidegger insists that it is meaningless to talk about the existence of objects and their properties in the absence of concernful activity, with its potential for breaking down. What really is is not defined by an objective omniscient observer, nor is it defined by an individual – the writer or computer designer – but rather by a space of potential for human concern and action*" (Winograd and Flores, 1986, p.37).

'reality' of enterprises and their environments, and thus to the human sense-making of that 'reality'.

As our next step, we aim to further explore the theoretical and practical considerations presented in this paper. We aim to start by analysing the available instruments for modelling language design, adaptation and combination, and the potential of their improving or combining in order to support purpose-specific language adaptations. In addition, we aim to extend these instruments to allow for explicit modelling of the modelling pragmatics. For instance, megamodel (Favre and Nguyen, 2004), viewpoint (ISO, 2011), metamodel hierarchies (Falkenberg et al., 1998), metamodel inference (Ossher et al., 2009) etc. are some of the instruments of our particular interest.

ACKNOWLEDGEMENTS

This work has been partially sponsored by the *Fonds National de la Recherche Luxembourg* (www.fnr.lu), via the PEARL programme.

REFERENCES

Anaby-Tavor, A., Amid, D., Fisher, A., Bercovici, A., Ossher, H., Callery, M., Desmond, M., Krasikov, S., and Simmonds, I. (2010). Insights into enterprise conceptual modeling. *Data Knowl. Eng.*, 69(12):1302–1318.

Anaya, V., Berio, G., Harzallah, M., Heymans, P., Matulevicius, R., Opdahl, A. L., Panetto, H., and Verdecho, M. (2010). The Unified Enterprise Modelling Language - Overview and Further Work. *Computers in Industry*, 61:99–111.

Barjis, J. (2009). Collaborative, Participative and Interactive Enterprise Modeling. In *ICEIS*, pages 651–662.

Bézivin, J., Bouzitouna, S., Fabro, M. D. D., Gervais, M.-P., Jouault, F., Kolovos, D. S., Kurtev, I., and Paige, R. F. (2006). A canonical scheme for model composition. In *ECMDA-FA*, pages 346–360.

Bjeković, M., Proper, H. A., and Sottet, J.-S. (2012). Towards a coherent enterprise modelling landscape. In *PoEM (Short Papers)*. CEUR-WS.org.

Briand, L. C., Melo, W. L., Seaman, C. B., and Basili, V. R. (1995). Characterizing and Assessing a Large-Scale Software Maintenance Organization. In *ICSE*, pages 133–143.

Bubenko, J. A. j., Persson, A., and Stirna, J. (2010). An Intentional Perspective on Enterprise Modeling. In Salinesi, C., Nurcan, S., Souveyet, C., and Ralyté, J., editors, *Intentional Perspectives on Information Systems Engineering*, pages 215–237. Springer.

Cho, H., Sun, Y., Gray, J., and White, J. (2011). Key challenges for modeling language creation by demonstration. In *ICSE 2011 Workshop on Flexible Modeling Tools, FLEXITOOLS*.

Clark, H. (1993). *Arenas of Language Use*. University of Chicago Press.

Cruse, A. (2011). *Meaning in Language: An Introduction to Semantics and Pragmatics*. Oxford Textbooks in Linguistics. Oxford University Press.

Davies, I., Green, P. F., Rosemann, M., Indulska, M., and Gallo, S. (2006). How do practitioners use conceptual modeling in practice? *Data Knowl. Eng.*, 58(3):358–380.

Delen, D., Dalal, N. P., and Benjamin, P. C. (2005). Integrated modeling: the key to holistic understanding of the enterprise. *Commun. ACM*, 48(4):107–112.

Egyedi, T. M. (2007). Standard-compliant, but incompatible?! *Computer Standards & Interfaces*, 29(6):605–613.

Elahi, G., Yu, E. S. K., and Annosi, M. C. (2008). Modeling Knowledge Transfer in a Software Maintenance Organization - An Experience Report and Critical Analysis. In *PoEM 2008*, pages 15–29.

Fabro, M. D. D. and Valduriez, P. (2009). Towards the efficient development of model transformations using model weaving and matching transformations. *Software and System Modeling*, 8(3):305–324.

Falkenberg, E. and Oei, J. (1994). Meta Model Hierarchies from an Object–Role Modelling Perspective. In *Proceedings of the First International Conference on Object–Role Modelling (ORM–1)*, pages 218–227.

Falkenberg, E. D., Hesse, W., Lindgreen, P., Nilsson, B. E., Oei, J., Rolland, C., Stamper, R. K., Assche, F. J. V., Verrijn-Stuart, A. A., and Voss, K. (1998). FRISCO - A Framework of Information System Concepts - The FRISCO Report. Technical report, IFIP WG 8.1 Task Group FRISCO.

Fauconnier, G. (2010). Mental Spaces. In Geeraerts, D. and Cuyckens, H., editor, *The Oxford Handbook of Cognitive Linguisics*, pages 351–376. Ofxord University Press.

Favre, J. and Nguyen, T. (2004). Towards a megamodel to model software evolution through transformation. In *SETRA workshop*.

Frank, U. (2002). Multi-perspective Enterprise Modeling (MEMO) - Conceptual Framework and Modeling Languages. In *Proceedings of HICSS'02*, volume 3. IEEE.

Frank, U. (2011). Some Guidelines for the Conception of Domain-Specific Modelling Languages. In *EMISA*, pages 93–106.

Gordijn, J. and Akkermans, H. (2003). Value based requirements engineering: Exploring innovative e-commerce ideas. *Requirements Engineering Journal*, 8(2):114–134.

Greefhorst, D., Koning, H., and Van Vliet, H. (2006). The many faces of architectural descriptions. *Information Systems Frontiers*, 8(2):103–113.

Harel, D. and Rumpe, B. (2004). Meaningful Modeling: What's the Semantics of "Semantics"? *IEEE Computer*, 37(10):64–72.

Hoppenbrouwers, S. (2003). *Freezing Language; Conceptualisation processes in ICT supported organisations*.

PhD thesis, University of Nijmegen, Nijmegen, The Netherlands.

Hoppenbrouwers, S., Proper, H., and Weide, T. v. d. (2005). A Fundamental View on the Process of Conceptual Modeling. In *Conceptual Modeling - ER 2005*, volume 3716, pages 128–143. Springer.

Hoppenbrouwers, S. J. B. A. and Wilmont, I. (2010). Focused Conceptualisation: Framing Questioning and Answering in Model-Oriented Dialogue Games. In *PoEM 2010*, pages 190–204.

Iacob, M.-E., Jonkers, H., Lankhorst, M., Proper, H., and Quartel, D. (2012). *ArchiMate 2.0 Specification*. The Open Group.

ISO (2011). *ISO/IEC/IEEE 40210:2011 Systems and software engineering – Architecture description*.

Kaidalova, J., Seigerroth, U., Kaczmarek, T., and Shilov, N. (2012). Practical Challenges of Enterprise Modeling in the Light of Business and IT Alignment. In *PoEM 2012*, pages 31–45.

Karagiannis, D. and Höfferer, P. (2006). Metamodels in Action: An overview. In *ICSOFT (1)*. INSTICC Press.

Karlsen, A. (2011). Enterprise Modeling Practice in ICT-Enabled Process Change. In *PoEM 2011*, pages 208–222.

Krogstie, J., Sindre, G., and Jørgensen, H. D. (2006). Process models representing knowledge for action: a revised quality framework. *EJIS*, 15(1):91–102.

Lankhorst, M., editor (2005). *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer, Berlin, Germany.

Lankhorst, M., Torre, L. v. d., Proper, H., Arbab, F., and Steen, M. (2005). Viewpoints and Visualisation. In (Lankhorst, 2005), pages 147–190.

Lankhorst, M. M., Proper, H. A., and Jonkers, H. (2010). The Anatomy of the ArchiMate Language. *IJISMD*, 1(1):1–32.

Linden, D. J. T. v. d., Hoppenbrouwers, S., Lartseva, A., and Molnar, W. (2012). Using Psychometrics to Explicate Personal Ontologies in Enterprise Modeling. *Applied Ontology*, 7:1–15.

Malavolta, I., Lago, P., Muccini, H., Pellicone, P., and Tang, A. (2012). What industry needs from architectural languages: an industrial study. Technical report, University of L'Aquilla.

Moody, D. (2009). The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. *IEEE Transactions on Software Engineering*, 35(6):756–779.

Novodranova, V. F. (2009). Representation of the Language Personality in LSP. In *Reconceptualizing LSP , Online proceedings of the XVII European LSP Symposium*.

OMG (2003). UML 2.0 Superstructure Specification – Final Adopted Specification. Technical Report ptc/03–08–02.

OMG (2006). Semantics of Business Vocabulary and Rules (SBVR). Technical Report dtc/06–03–02, Object Management Group, Needham, Massachusetts.

OMG (2008). Business Process Modeling Notation, V1.1. OMG Available Specification OMG Document Number: formal/2008-01-17.

Opdahl, A., Berio, G., Harzallah, M., and Matulevicius, R. (2012). Ontology for Enterprise and Information Systems Modelling. *Applied Ontology*, 7(1):49–92.

Ossher, H., Bellamy, R. K. E., Amid, D., Anaby-Tavor, A., Callery, M., Desmond, M., de Vries, J., Fisher, A., Frauenhofer, T., Krasikov, S., Simmonds, I., and Swart, C. (2009). Business insight toolkit: Flexible pre-requirements modeling. In *ICSE Companion*, pages 423–424. IEEE.

Proper, H. A., Verrijn-Stuart, A. A., and Hoppenbrouwers, S. (2005). On Utility-based Selection of Architecture-Modelling Concepts. In *APCCM 2005*, pages 25–34.

Rothenberg, J. (1989). The Nature of Modeling. In *Artificial intelligence, simulation & modeling*, pages 75–92. John Wiley & Sons, Inc., USA.

Schmid, H.-J. (2010). Entrenchment, Salience, and Basic Levels. In Geeraerts, D. and Cuyckens, H., editor, *The Oxford Handbook of Cognitive Linguistics*, pages 117–138. Oxford University Press.

Stachowiak, H. (1973). *Allgemeine Modelltheorie*. Springer, Berlin, Germany.

Stirna, J. and Persson, A. (2012). Evolution of an Enterprise Modeling Method - Next Generation Improvements of EKD. In *PoEM 2012*, pages 1–15.

Thalheim, B. (2011). The Theory of Conceptual Models, the Theory of Conceptual Modelling and Foundations of Conceptual Modelling. In *Handbook of Conceptual Modeling*, pages 543–577. Springer.

Thalheim, B. (2012). Syntax, Semantics and Pragmatics of Conceptual Modelling. In *NLDB*, pages 1–10. Springer.

Vallecillo, A. (2010). On the Combination of Domain Specific Modeling Languages. In *ECMFA*, pages 305–320.

Vernadat, F. (2002). UEML: Towards a Unified Enterprise Modelling Language. *International Journal of Production Research*, 40(17):4309–4321.

Wagter, R., Proper, H., and Witte, D. (2012). A Practice-Based Framework for Enterprise Coherence. In *Proceedings of PRET 2012*, volume 120 of *LNBIP*. Springer. To appear.

Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition - A New Foundation for Design*. Ablex Publishing Corporation.

Winter, R. and Fischer, R. (2007). Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. *Journal Of Enterprise Architecture*, 3(2):7–18.

Wood–Harper, A., Antill, L., and Avison, D. (1985). *Information Systems Definition: The Multiview Approach*. Blackwell, Oxford, United Kingdom.

Yu, E. and Mylopoulos, J. (1996). Using goals, rules, and methods to support reasoning in business process reengineering. *International Journal of Intelligent Systems in Accounting, Finance and Management*, 5(1):1—13.

Zachman, J. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3).

Zivkovic, S., Kühn, H., and Karagiannis, D. (2007). Facilitate Modelling Using Method Integration: An Approach Using Mappings and Integration Rules. In *ECIS*, pages 2038–2049.

Analysis and Evaluation of Business Process Modeling Adoption in Collaborative Networks

Hodjat Soleimani Malekan and Hamideh Afsarmanesh
Informatics Institute, University of Amsterdam, Post bus 94323, 1090 GH Amsterdam, The Netherlands
{h.soleimanimalekan, h.afsarmanesh}@uva.nl

Keywords: Business Processes, Collaborative Network, Business Process Modeling Languages, Virtual Organizations (VO), VO Breeding Environments, Service-Oriented Architecture.

Abstract: Besides, knowledge and information enterprises can share Business Processes (BPs) within Collaborative Networks (CNs). Each enterprise has a set of BPs that it can perform, and through developing integrated BPs in the CN they deploy their capacities and capabilities. Selecting and adopting the appropriate BP modelling languages (BPML) for the purpose of formalizing BPs are challenging, because of the variety of existing methods, tools, and standards with different strengths and weaknesses. In surveys published so far on BP modeling mostly, a set of general features of the main BP languages and standards are compared. However, they have not paid attention to the level of different categories of BPMLs. Furthermore, there are no surveys analysing and evaluating the prerequisites to fulfil CN's requirements. This paper first proposes a set of categories for the main BP languages and standards. Then a novel BP evaluation approach, in CN context is introduced. Finally, different categories are discussed and analysed by addressing their suitability to support CNs.

1 INTRODUCTION

Adopting Business Process Modeling Languages (BPMLs), including the introduced standards, tools, and techniques, have greatly influenced enterprises toward capturing opportunities, reducing costs, and increasing productivity.

The BP technologies themselves however have also been affected by high demand of market, as well as the step-wise maturity of Business Process Management (BPM) theories. This has caused rapid changes in the last decade developed BPML tools and standards, while creating challenges for the BP modeling selection and adoption in networked enterprise. (Camarinha-Matos and Afsarmanesh, 2008).

Most of these aproaches are founded on Service Oriented Architecture (SOA), apply the formalized BPs, performed at every enterprise, to facilitate service interoperation and enterprise collaboration (Papazoglou and Heuvel, 2006). formalized BP are therefore important for effective cooperation among different enterprises within the Collaborative Networks (CNs), and without formalized representation of their BPs, enterprises cannot effectively share their competencies and capabilities.

The BPMLs differ from each other in their modeling approaches for design, analyzing, and enacting of BPs. Focusing on the purpose of supporting enterprises, with their collaboration within the CNs, the selection and adoption of a suitable BPML is critical, while challenging. Published surveys on BPMLs e.g. Roser and Bauer (2005), LU and Sadiq (2007), also Ko, S.Lee and E.Lee (2009) have already tackled the comparison between a certain features of the main BPMLs tools and standards.

Most contemporary surveys focus on comparing a set of two or more BPML standards and tools. So, there is a lack of emphsis on comparing different categories of BPMLs, to which these standards or tools may belong. For instance the distinct features aimed by their design, such as to evaluate and emphasize their graphical, ontological, executional, etc. aspects of the BP modeling, is not assessed for this purpose.

Moreover, demonestrating a set of categories for BPMLs classification, in order to perform analysis and evaluation of BPMLs for their adoption in support of CNs, a novel analysis method is introduced to manifest CN's characteristics and to assess different BPML categories against them.

Hence, we first review the main concepts of CNs and BPs, identifying the role of formalized BPs in CN (in Section 2). Then, based on a systematic reviewing approach, and considering the existing categorizations of the main BPMLs, we introduce our BPML categorization (in Section 3). In Section 4, founded on collaboration purposes, we specify a number of most relevant criteria for comparing the introduced BPML categories, and analyzing them for the aim of supporting enterprise collaborations. Finally, our analysis and evaluation approach is discussed (in Section 5), and our conclusions are presented (in Section 6).

2 ROLE OF BPs IN CNs

Within the collaborative-networked environment the enterprises have the opportunities to share their resources through collaboration, including knowledge, information. This can be best achieved, by means of formalized BPs (Camarinha-Matos and Afsarmanesh, 2008). Collaborative BP integration is aimed by enterprises to accomplish value-added business services, beyond the capabilities of their individual organizations.

Besides integration, in most approaches for instance presented by Papazoglou and Heuvel (2006), BPs constitute the building blocks for establishment of SOA, through BPs implementation as web services. In this section, after reviewing related definitions for CNs and BPs, we present an analysis of the BPMLs from the CN requirements point of view.

2.1 Principal Definitions

A general definition of Collaborative Network is presented by Camarinha-Matos and Afsarmanesh (2008) as: "*an alliance constituting a variety of entities that are autonomous, geographically distributed, and heterogeneous in terms of their operating environment, decision making, culture and social capital, that cooperate/collaborate to better achieve common/compatible goals, and their interactions are supported by the computer networks.*"

The two main forms of CNs are the: Virtual Organization (VO) and VO Breeding Environment (VBE). In a VO, partners choose co-working and sharing their BPs and other resources to accomplish their common goals. The motivations for this coalition are commonly formed around specific market targets or innovation purposes. VBEs, which establish long-term alliances of organizations, capture and save BPs of partners in their directories. The VO broker, who seizes the opportunity and chooses the participants for the VO in the VBE context, considers selecting and integrating BPs of different organizations to shape new VOs responding to achievable opportunities. (Afsarmanesh et al. 2011).

Related to our research, a set of standard definitions for the BP notions exist that is provided by Workflow Management Coalition (WFMC, 1999) and is addressed below.

A typical definition of BP is: a series of one or more linked procedures or activities, which collectively realize a business objective or policy-related goal. Workflow Management System (WFMS) can automate and control the execution of the BPs. The notion of BPM comprises concepts, methods, and techniques to support organizational aspect of processes, which are needed for the design, administration, configuration, enactment, and analysis of BPs (Weske, 2007). It also covers the "*diagnosis*" aspect of the BPs further to the WFMS lifecycle (Van der Aalst, 2003).

Havey (2009) outlines the focuses of BP modeling on design and execution aspects of the BPs. BP Modeling aims at representing an abstract but meaningful demonstration of the real business domains. This goal is achieved through provision of appropriate syntax and semantics in BPMLs, to meet the BP's requirements. (Lu and Sadiq, 2007).

2.2 Chronological View of BPMLs in Support of Collaboration

Here we address the evolution of BPMLs from collaboration point of view. In the 80s, the necessity of process-awareness was recognized, beyond the level which was required for development of Management Information Systems (MIS). Furthermore, besides understanding the flow of operations in MIS, organizations and business domain experts needed to also understand the information aspects of the BPs in MIS (Delvin and Murphy, 1988).

The WFMSs, which initially were intended to facilitate automatic transformation of electronic documents, was then introduced as the new tools to enable business analysts in designing and expressing BPs, at the beginning of 90s. For the purpose of depicting information exchange among systems, the behavioural concepts (i.e. the sequence and merge) were then used in BP modeling (Georgakopoulos, 1995).

Afterward in the 90s, applying the Business Process Re-engineering as well as embedding the best business practices in the market, vendors were able to integrate and aligned separate software modules, under the so-called Enterprise Resource Planning (ERP) systems. To support ERPs, the BPMLs have focused on dynamic aspects of the BPs. Nevertheless the interactions between the designed modules were not so easy to achieve within the ERPs (Van der Aalst, 2009).

Responding to the proliferation needs of the integrating legacy systems into customized applications and ERP modules, the Enterprise Application Integration (EAI) (Lee and Siau, 2003) have tried to remedy the problem of inefficient BPs' integration. So, interaction-enabling entities (e.g. messages) gained significance. This level of collaboration provided an infrastructure for cooperation of enterprises through resource sharing, while preserving their heterogeneity.

The more maturity in deployment of XML, in the late 90s, resulted in better integration of applications, and changed the co-working intensity of enterprises to an advanced level, called business to business (B2B) (Havey, 2009).

Coordinating the BPs adopted by companies, concluded in integrating autonomous and independent applications, via loosely coupled mechanism of SOA (Zdun et al., 2006). SOA approach tries to establish orchestration and choreography of web services, to achieve their successful cooperation.

Nowadays, BP related topics e.g. the BP mining (Van der Aalst and Dustar, 2012) and diagnosis approaches that address BP monitoring and their continuous improvement constitute promising research lines.

3 CATEGORIZATION OF BPMLs

Aiming to cover various BP modeling tools and standards, which are introduced in the main related publications, we focus on a specific set of attributes and specifications of the BPMLs for their categorization. Our categories basically focus on recognizing the BPML's capabilities as well as the suitability features in each category, in support of criteria for collaboration. Therefore, we first study related scientific DB and conferences, then classify the existing categorization publications into two classes of: "General Review", and "Particular Evaluation".

In this section, first we review the results presented in published surveys, from the point of view of our two classes addressed above, and further classify a set of minimal relevant BPML categorization approaches. Finally, we introduce our more detailed categorization.

3.1 BPML Categorization Review

As mentioned before, we divide the contemporary reviews of BPMLs into two main classes of "*General Reviews*" and "*Particular Evaluation*". "General Reviews" are mostly focused on general uses, and on encompassing the main specifications of the BPML categories. For instance the work of Havey (2009) that focuses on *presenting good BP Modeling Architecture*, where it first addresses aspects of BP modeling applications (i.e. design, run, monitor, etc.), and then introduces the four categories of BPMLs, including: *notation* languages (e.g. BPMN), *execution* languages (e.g. BPEL), *choreography* languages (e.g. WS-CDL), and *process administration* languages (e.g. BPQI). Also, classification presented by Ko et al. (2009) and Mili et al. (2010) are instances of this category.

But, publications in the "Particular Evaluations" class focus on BPML categorization for specific purposes. The works of Roser and Bauer (2005), Lu and Sadiq (2007), and De Nicola et al. (2007) are instances in this category. For example, in (De Nicola et al., 2007) the categories are introduced around the subject of "introducing an ontological approach for BP modeling", including Descriptive (e.g. BPMN), Procedural (e.g. XPDL), Formal (e.g. PSL), and Ontology-based (e.g. OWL-s).

3.2 Introduced BPML Categories

Using the "general review" and "particular evaluation" criteria as the base, we introduce a more comprehensive framework including six classes: "graphical", "formal", "executional, "ontological", "inter-operational", and "monitorial", that together capture all kinds of addressed BPMLs.

The main characteristic of each of these six categories, and their main representative example BPMLs are briefly (due to space limitation) described in the following subsections. Also, a set of popular BP languages and standards are named below as the example of each category.

Although it is possible for a BPML to be categorized in more than one category, but here we have placed each BPML in its most representative category only. They could be adopted and utilized by CN members based on their category's

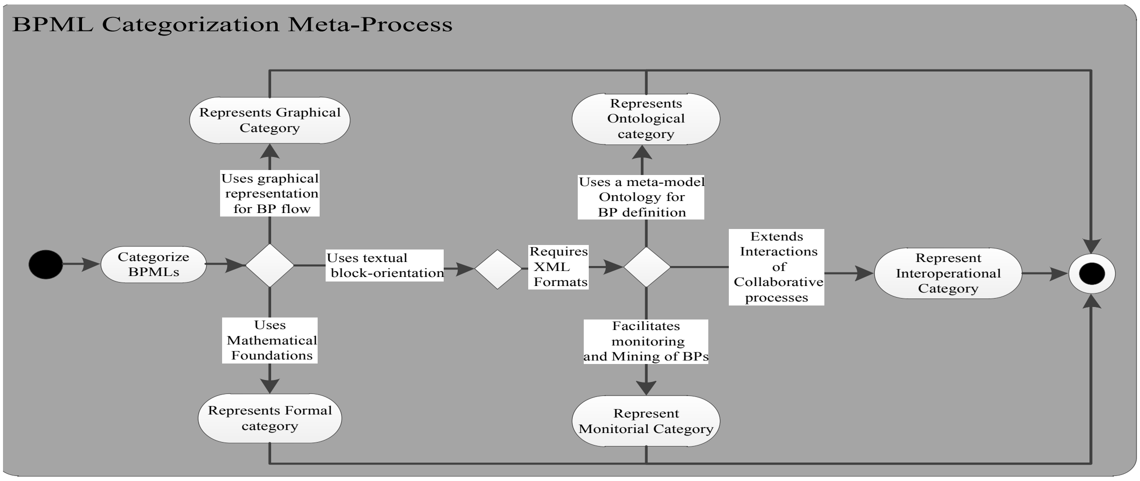


Figure 1: The BPML categorization meta-process method.

characteristics.

The meta-process adopted by our categorization method and how we reached the six specific classes is briefly depicted in figure1.

3.2.1 Graphical BP Languages

Rooted in graphical picturesque format, this classical generation has appeared. BP modeling languages in this category mostly emphasize illustrating the system behaviour and its abstraction. These languages are not typically formal. (e.g. IDEF, EPC, UML 2.0, BPMN).

3.2.2 Formal BP Languages

Formalization in this category is founded upon mathematical principles. Although, adoption of graphical symbols is possible in some of these languages, but difficulties in user's understanding hold them mostly at theoretical and mainly academic utilizations. (e.g. Petri-Net, Pi-calculus, PSL, Reo).

3.2.3 Executional BP Languages

The idea of automatic execution of BPs by software engines, support the formation of this category. The XML structure plays an important role in deployment of this category, and clarifies BPs by their computerized semantics. Besides, the popularity of BP modeling and service invocation in industries are other important issues in the category. (e.g. BPEL, WS-CDL, XPDL, YAWL).

3.2.4 Ontological BP Languages

Likewise the ontology approach, which studies the things that exist and tries to describe them, this category addresses semantic capture and tries to constitute the base for an increasing number of BP modeling languages, through proposing different meta-models. The ontological layer in these languages clarifies the roles, entities, and interactions. This category has also the advantages of using XML formats. (OWL-s, WSMO, BPDM).

3.2.5 Interoperational BP Languages

Rooted in business-to-business interaction, this category focuses on modeling public sharable processes of partners, among many business partners. To accomplish this key concern in inter-operational category, XML standards are elaborated as the main enablers. (e.g. RossettaNet, eb-XML/BPSS).

3.2.6 Monitorial BP Language

As we discussed previously (in section 2), modern business process modeling trends to address the diagnosis iteration of the BP Lifecycle. Focusing on the Business Activity Monitoring (BAM) point of view, the emphasis is on monitoring and resolving the deadlocks or problems in flow of BPs. Furthermore, extract and unambiguous approach for recognizing BP modeling based on a dynamic logging of process behaviour, the so-called process mining is still promising (van der Aalst and Dustar, 2012), (e.g. BPRI and BPQI).

4 SUPPORTING CN - BPML EVALUATION

The evaluation framework should be concise and descriptive. Having emphasis on categories and not every BPML, multi-aspect evaluation of a phenomenon requires a methodology, to support maximal coverage of the target area. For the purpose of appraising BPMLs in supporting CNs requirements, we should consider both the BPs and the CNs aspects simultaneously. Therefore, our designed evaluation methods as well as our evaluation process are discussed in following sub-sections, respectively.

4.1 Proposed Evaluation Method

Several BP modeling goal-settings have been introduced based on different approaches. For instance a set of five generic software process modeling objectives have been specified in (Curtis et al., 1992) as follow: “to support process improvement”, “to facilitate human understanding and communication”, “having automated guidance in performing process”, “to support process management”, and “to automate execution support”. Also, for the non-functional BP modeling requirement (Chung and Do Prado, 2009) has presented a series of objectives (e.g. the support for discovering of dependencies of processes, the support for changing management, etc.). These context-aware objectives still hold today. In our point of view for supporting more effective BP collaboration in a CN, we can further add, “to support enterprise collaboration” into this context.

Rooted in the debate in Section 2, our primary aim is to focus on supporting collaboration through formalized BPs and evaluating BPML categories for this purpose. Therefore, we first follow a goal-based approach (also known as the objective-based approach) explained in (Goldkuhl and Lagsten, 2012) to extract the collaborative intention issues within CN’s concept.

Our goal-based approach has focused on a number of qualitative criteria and indicators, related to set goals, systematically. As the evaluation method, we adopt Critical Success Factors (CSFs) method. CSF is a classical flexible method to maximize goal achievement, through selecting, working, and monitoring a few certain factors, which are vital for success. So, we follow the requirements of achieving established objectives, by running a Critical Success Factors Analysis described and explained in (OASIS, 2008) and partially in (Trkman, 2010) and (Sudhakar, 2012). After CSF identification, a set of requirement indicators for monitoring them is provided by CFA.

The CFA constitutes following elements:

- **Objectives**: Those are directed by customers and are hard to measure.
- **CSFs**: including between three to six sub-goals, which without their direct support, achieving goals are unattainable.
- **Requirements Indicators**: represented key performance indicators, which are measurable and directly support CSFs.
- **CFA Diagram**: for better illustrating the measurable context, CFA diagram is used.

For supporting characteristics of CNs to achieve their goals and to better describe the particularities of the CN context, especially for VOs and VBEs, we use the Reference model for Collaborative Networks” (ARCON). The ARCON model explains aspects, approaches and elements of the CN’s environment (Camarinha-Matos and Afsarmanesh, 2008).

Eventually, we hold a CFA study to find out CN-compliance CSFs, and the vital requirements indicators for achieving our goal. These issues are provided based on systematic technical reviews and experts opinions. We then discuss, the BPML’s categories versus the recognized requirements indicators, and represent the conclusions.

4.2 Evaluation Process

Regarding the (ARCON) model introduced by Camarinha-Matos and Afsarmanesh (2008), and our discussion in section 2, VO/VBE need to manipulate formalized BPs.

Also, regarding CN’s definition (in 2.1) the following aspects indicate the main constitutional objective themes in the CN discipline:

- **Goal-orientation** [focusing on goals through business interactions]
- **Infrastructure for Commonality** [supporting co-working and coordination toward goals]
- **Node Heterogeneity** [non-uniformity in different properties, i.e. operational processes]
- **Network enabling** [support by computer networks]

The four above-mentioned objective themes are the main CN realization’s objectives, extracted from its standard definition. To attain these objectives, defining and aligning a set of CSFs are inevitable.

The supporting CSFs for CN are as follows: first, to enable successful collaboration, BP modeling tool should provide enough "comprehensibility" for partners (BP Analysts, IT experts, etc.). The "ease of use" is another issue, which supports convenient interoperation through CNs. "expressiveness for behaviour", is another challenging issue for enactment of BPs in CNs. For cost-effective achievement of goal in CNs, "accessibility" of BP documents and standards has to be considered. The coverage of CN's objective and the introduced CSFs are illustrated in table 1. The "C" in the box at the intersection of rows and columns represent the minimal *coverage* between our CSF and CN's constitutional objective elements.

Table 1: Intersection of CN's objectives and CSFs.

CSFs \ CN's Objectives	Goal-Orientation	Commonality	Heterogeneity	Network enabling
Comprehensibility		C	C	
Ease of Use		C	C	
Expressiveness for behaviour			C	C
Accessibility	C			

The last step of our evaluation, is finding a series of generic required indicators from BP modeling context to appraise the suitability of BPML's categories for CN's. They have been extracted from the literature and the standards (ISO, 2010E):

Understandability: is the ease of interpretation and capture by which under specified circumstances, the user can interpret an instance, model, analyse, and develop the BP model (Mendling et al., 2007).

Expressability: explains the capability to represent the process model's attributes like: control, resources, flow structures, in an unambiguous way (Kiepuszewski, 2002).

Flexibility: is defined as the ease with which in BP modeling the modifications are possible in types and instances, based on incomplete level of abstraction (Lu and Sadiq, 2007). So, partial effects of changes, does not necessarily imply completely replacement of BP models (Schonenberg et al., 2008).

Availability: comprises amount and degree to which business process modeling documents in specific formats, and standards are accessible and adoptable. They are ready-to-use for desired collaboration by the organizations (Milanivic, 2008).

Enactability: is defined as the ambition of acquiring capability to completely execute of the BP model directly and without exploiting extra tools and information (Russell, 2007).

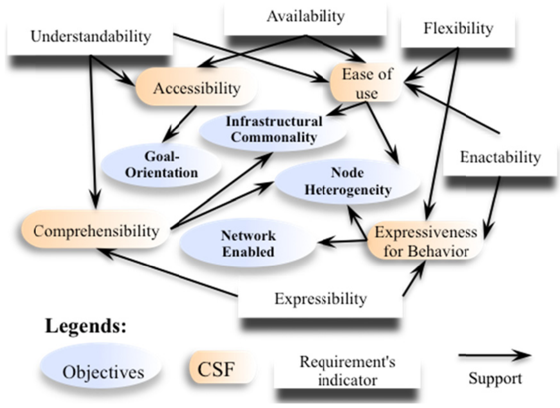


Figure 2: Interconnections in CFA diagram.

Figure2 represents CFA analysis as the interactions of different CN's objectives, supporting CSFs and monitoring requirements indicators and the types of effect between different entities in CNs (which is represented in the figure by the support arrow). In the next section we address and evaluate each of BPMLs categorization against the five standard criteria.

5 DISCUSSION

Our evaluation comprises a two-dimensional descriptive evaluation. The first dimension consists of BPML categories. Six comprehensive categories are introduced and defined. Please note that we focus on novel BPML categories – as the aimed origins – instead of the languages, so there could be a number of choices from BPML's members in each categories for CNs.

5.1 Availability

Availability has its roots in reliability notion, which implies, ratio of the time that users have received the service according to prior level of agreements (ISO, 2010E). Unavailability of a modeling BP language happens when we don't have "steady-state", "intervals", and "user-perceived" availability (Milanivic et al., 2008). For our evaluation, we assume availability as the existence of BPML documents within the context of CNs.

There is an annual research of BPM Market since 2005, by Wolf and Harmon (2012), thoroughly surveying the BPM trends. From their 2012 report, they state that the rate of availability for graphical BPMLs is at the highest level. For example the BPMN is used by more than 60% of all organizations. Meanwhile, there is less availability for ontological BPMLs (e.g. BPDM). Although, the debate on the timely development of trends is not the focus of this paper, but decrease in attraction level of BPEL during recent years is noticeable. Even interest and availability of UML and EPC slightly decrease. Also according to that survey the pervasiveness of the rest of BPML categories (e.g. interoperational, formal) are the lowest in usage ranking. So, it is expected that organizations initiate collaboration in CNs applying graphical BPMLs, and especially BPMN.

5.2 Enactability

The enactability is an important phase in BPM life cycle. According to (van der Aalst et al., 2003), after accomplishing "process design" and "system configuration" at the third step of the BPM's lifecycle "process enactment" is located right before the "diagnosis" step. The more independent is the BPML from the technology and vendor executable environments, the better enactability has in CNs.

Using the formal semantics for more effective enactment (ter Hofstede et al., 2010) supports – and does not contradict the increase in understandability in support of the requirements in CNs. Executional BPMLs enable the enactments of BPs, for sharing BP's information and automatically executing them through block-based and machine understandable structures. But, despite their common executional capabilities, they have their particularities.

Within Executional category, BPEL describes behaviour of BPs within interaction between process and its partner, and efficiently supports orchestration. WS-CDL executional aspect consists of peer to peer collaboration of partners from a global point of view for supporting choreography.

But, in this category some of the languages such as BPEL have restrictive syntax (Recker and Mendling, 2006), which is a limitation for this popular language, and some (e.g. YAWL) have exact executional syntax (ter Hoftstede 2010).

The formal category languages - except embedded notions like (pi-calculus in WSCDL) provide graphical enactability interface, e.g. in reo and Petri net. Ontological BPMLs, because of their XML supporting structures have convenient level of enactability.

Executional issues in interoperational BPML category, where XML enactability is embedded, have some difficulties e.g. naming and XML reusability in RosettaNet (Damodaran, 2004) or deficiencies in event handling during interactions (Green et al., 2007). Ontological BPML category focuses on semantic aspects (e.g. OWL-s), and runs enactment in an abstract level.

5.3 Expressability

The importance of expressability in CNs arises from the way we wish to express the BPs, so that they can be shared among partners. This expressive power of modeling language represents the possibility of expressing constructs in direct or indirect manner (Kiepuszewski et al., 2002). These constructs comprise: control, resources, data, organization, execution, and behaviour of a business models. Expressability encompasses the notion of suitability, which focuses on modeling and implies conformance of the BPML with for instance 43 workflow patterns introduced in (Russell et al., 2006). Although, the evaluated domain in that paper does not focus on BPML categories, but provides a general inception for comparison of BPML categories.

As we map BPMLs' evaluation in Russell, ter Hofstedeh and van der Aalst, (2006) to our proposed categories, a number of these patterns e.g. "discrimination", "milestone", "partially join", etc. are the kind of patterns which languages and standards have difficulties in expressing them.

We could state that, commonly, the graphical BPML category has better compatibility, while in executional category- except for YAWL- languages have some deficiencies, for example for supporting "Arbitrary Cycle", because of their rigidity in capturing real-world abstraction.

Based on evaluation of Russell et al., (2006) the of formal languages category members have good capability of expressiveness, because of their mathematical foundation, e.g. Petri-Net; expressive power (van der Aalst et al., 2003) used in workflow pattern design, or constraint Automata is used in Reo. Ontological languages use logical basis for instance in OWL-s for representing better expressiveness (W3C-OWL-s, 2004).

5.4 Flexibility

Supporting the dynamicity of CNs, the flexibility issues in BPMLs for describing BP's interaction is necessary. BPMLs focus to sustain their dynamicity in coping with expected and unexpected changes, through adopting flexibility. In (Schonenberg et al., 2008) four types of flexibility are presented as:*" design", "deviation", "underspecification"* and *"change"*.

The flexibility support, mostly in two first above-mentioned types, BPMLs rely on their pre-design notations and are abstract from flexibility concerns. On the other hands, the block-based (rule-based) BPMLs could manage the flexibility in higher level (e.g. deviation or underspesification) (Lu and Sadiq, 2007).

The flexibility in the graphical BPML category, within different languages and standards is considered in different ways. In BPMN, by predicting three types of diagram for collaboration, and for the concepts of Pool and lane, the decomposition for changes is possible. The Frame and Frame Heading techniques in UML 2.0 Activity Diagram let the elements of the languages to be defined and described in a modular and flexible structure. So, "design and deviation" are supported.

Likewise in formal category, mathematical concepts help to retain model identity; for instance the structure of Atomic and Complex activities in PSL, besides graphical representation in Petri-Net and Reo simplifies the modification flexibilities. So, "design and deviation" are supported.

Based on XML structures, which usually support flexibility in design and changes, and even underspesification, to certain extent (Schonenberg 2008). YAWL, BEPL (inter-relations), and WSCDL (choreography) support various types of flexibility. Even RosettaNet PIP techniques, channelizes the modifications. This benefit supports within block-based structure. Ontological BPML category considers flexibility at convenient level, which let modification to be based on primary definition of BPs (e.g. process model definition in OWL-s).

5.5 Understandability

The understandability shall facilitate the BP acquisitions and interactions among CN's stakeholders. This notion has been reviewed and analysed during several works especially verses the complexity as the other extreme. Generally, understandability comprises the following two aspects mentioned in (Mendling et al., 2007):

- Model-related factors, which affect the understandability, e.g. unambiguity, simplicity.
- Person-related factors, which have close relations to knowledge and experience of participants

Although, the understandability has been reviewed several times, and there is a number of guidelines e.g. the smaller size of the model makes it better for understanding; or the higher degree of input and output to one element causes the more complexity of understandability, etc. But, the ease of "comprehension of a model", "presenting without error", and "labelling less ambiguous" (Mendling et al., 2010) constitute main understandability's principles in BPMLs.

Generally, the graph-based languages are more understandable than rule-based ones (Lu and Sadiq, 2007). That is also the reason why they become more popular at enterprises. However, within graphical standards, BPMN is more complex for understanding compared to UML and EPC (Indulska et al., 2009).

On the other hand, Executional and ontological BPML categories because of having less cooperation with human side, their understandability is under criticism. Also, the interoperational standards (e.g. the PIP technics knowledge in Rosettanet) are at a more abstract level of understandability ((Damodaran, 2004), (Green et al., 2007)).

5.6 Comparing Results

Through the discussion, we analysed the adoption of BPML in regards to the set of requirement indicators, which represented for evaluating BPML categories at the second level of our evaluation. Grounded in our goal-based approach and by using a CFA method, we identified six requirement criteria that helped us, to measure the collaboration-aware adoptability of BPMLs. The result of our extensive evaluation in previous sections is summarized in Table 2.

Table 2: Summary of evaluation.

Requirements indicators in Support Of CN / BP Modeling Languages Categories	Understandability	Expressibility	Enactability	Availability	Flexibility
Graphical	S	A	A	S	A
Formal	M	S	A	M	A
Executional	A	A	S	M	A
Ontological	A	S	A	A	A
Interoperational	A	A	S	A	A
Monitorial	N	N	N	N	N

Comments:
S: Strong support A: Advanced support
M: Moderate support N: Not Addressed

As showed in that table, we use four levels of supports as: *Strong*, *Advanced*, *Moderate* and *Not addressed* levels, from CN's member relative points of view. Because of analytical theme of paper, we opt qualitative survey method.

Regarding discussions in our previous sections, the graphical category has advantages of understandability and availability. Executional category is strong in enactability, also flexibility of BPs, besides the importance of less ambiguity in modeling real world should not be disregarded, although lacks of interactive graphical depiction needed for less technical users is serious criticism yet.

Due to complexity of their user interaction, the formal languages are not pervasive, but should be considered as the supporting layer for soundness for graphical modeling languages. Ontological languages, because of their well-defined semantics, and their focus on graphical and executional aspects, are desirable but not yet sufficiently mature and popular.

BP adopting in interoperational BPML category, which isjust used for support of collaboration, mostly emphasize on interactions instead of abstract BP modeling from real world, also their flexibility level and understandability problems for users are serious concerns. Monitorial BP Languages are not practically fitting in this context to evaluate, but promising.

6 CONCLUSIONS

In our paper, we presented an analysis and evaluation of the Business Process Modeling Languages categories in support of Collaborative Networks. We review their suitability for supporting collaboration among enterprises.

To ensure a systematic and methodological approach in our review process, we have reviewed publications addressing categories of business process modeling in the context of BPMLs. Then, we have discussed different BP languages, and from a language-independent perspective, we have introduced our six categories of BPMLs.

Additionally, we have identified a set of criteria required for adopting BPs among enterprises in CNs. Based on these defined set of criteria, the six BPML categories are further analysed, regarding how they fulfil the collaboration requirements for CNs.

As we have employed a partially qualitative analysis approach, our analysis is not fully objective. Although, based on the results showed in table 2, the most suitable categories of BPMLs, especially for adoption in Virtual Organizations and VO Breeding Environments, are represented.

The elaborated results achieved through our evaluation of BPMLs in the context of CNs, indicate that depending on the requirements, the domain experts may preferably select BPMN or OWL-s for the purpose of their BP integration.

ACKNOWLEDGEMENTS

This work is partially supported by the FP7 project GLONET, funded by the European Commission.

REFERENCES

Afsarmanesh, H., Camarinha-Matos, L.M., Msanjila, S.S., 2011. Models, Methodologies, and Tools Supporting Establishment and Management of Second-Generation VBEs. *Systems, Man, and Cybernetics, IEEE Transactions on*, *41*(5), pp. 692-710.

Camarinha-Matos, L.M., Afsarmanesh, H., 2008. On Reference Models for Collaborative Networked Organizations. *International Journal of Production Research*, *46*(9), pp.2453-2469.

Chung, L., do Prado Leite, J., 2009. On non-functional requirements in software engineering. *Conceptual modeling: Foundations and applications*, pp.363-379.

Curtis, B., Kellner, M.I., Over, J.1992. Process modeling. *Communications of the ACM*, *35*(9), pp.75-90.

Damodaran, S., 2004. B2B integration over the Internet with XML: RosettaNet successes and challenges. In *Proceedings of the 13th international World Wide Web conference,* pp. 188-195.

Devlin, B. A., & Murphy, P. T., 1988. An architecture for a business and information system. *IBM systems Journal*, *27*(1), pp.60-80.

De Nicola, A., Lezoche, M., Missikoff, M., 2007. An Ontological Approach to Business Process Modeling. In *Proceedings of IICAI-07.*

Georgakopoulos, D., Hornick, M., Sheth, A., 1995. An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure. In *Distributed and Parallel Databases*, *3*(2), pp. 119-153.

Goldkuhl, G., & Lagsten, J., 2012. Different roles of evaluation in information systems research. In IT Artefact Design and Workpractice Intervention.

Green, P., Rosemann, M., Indulska, M., Manning, C., 2007. "Candidate Interoperability Standards: An Ontological Overlap Analysis", Data & Knowledge Engineering, Vol.62, No.2, pp. 274-291.

Havey, M., 2009. Essential business process modeling. O'Reilly Media,.

Indulska, M., zur Muehlen, M., and Recker, J., 2009."Measuring Method Complexity: The case of

Business Process Modeling Notation", Report 09-03:viewed on December 2012 in www.BPMcenter.org.
ISO, International Organization for Standardization, 2010E. ISO/IEC/IEEE24765, Systems and Software Engineering Vocabulary.
Kiepuszewski, B., ter Hofstede, A.H.M, van der Aalst, W.M.P., 2002. "Fundamentals of Control Flow in Workflows" (Revised version). ‚QUT Technical report, FIT-TR-2002-03.
Ko, R.K., Lee, S.S., Lee, E.W., 2009. Business Process Management (BPM) Standards: a Survey. *Business Process Management Journal*, *15*(5), pp.744-791.
Lee, J., Siau, K., Hong, S., 2003. Enterprise Integration with ERP and EAI. In *Communications of the ACM*, *46*(2), pp.54-60
Lu, R., Sadiq, S., 2007. A Survey of Comparative Business Process Modeling Approaches. In *Business Information Systems*. Springer Berlin, pp. 82-94.
Mendling, J., Reijers, H., Cardoso, J., 2007. What makes process models understandable?. 5th international Conference *BPM*, Vol. 4714 Springer, pp.48-63.
Mendling, J., Reijers, H.A., van der Aalst, W.M., 2010. Seven Process Modeling Guidelines (7PMG). *Information and Software Technology*, *52*(2), pp.127-136.
Milanivic, N., Milic, B., Malek, M., 2008. Modeling Business Process Availability. In Proc. IEEE congress on services (SERVICES), pp.315-321.
Mili, H., Tremblay, G., Jaoude, G.B., Lefebvre, E., Elabed, L., Boussaidi, G.E., 2010. Business Process Modeling Languages: Sorting Through The Alphabet Soup. *ACM Computing Surveys (CSUR)*, *43*(1), 4.
OASIS, 2008. "Reference Architecture for service Oriented Architecture version 1.0", viewed on December 2012 available in: http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf.
Papazoglou, M.P., van den Heuvel, W.J, 2006. Service-Oriented Design and Development Methodology. International Journal of Web Engineering and Technology, Vol. 2, No.4.
Recker, J., & Mendling, J., 2006. " On The Translation between BPMN and BPEL: Conceptual Mismatch between Process Modeling Languages", In Proceedings 18th International Conference of Advanced Information Systems Engineering, pp. 521-532.
Roser, S., Bauer, B., 2005. A Categorization of Collaborative Business Process Modeling Techniques. *E-Commerce Technology Workshops, Seventh IEEE International Conference* pp. 43-51.
Russell, N., ter Hofstede, A.H.M., van der Aalst, W.M.P., Mulyar, N., 2006. Workflow Control-flow Patterns: A Revised View", BPM center report, BPM-06-29, viewed in Dec, 2012 at www.BPM-Center.org
Russell, N.C., 2007. Foundations of Process-Aware Information Systems, Ph.D. Dissertation, Queensland University of Technology.
Schonenberg, H., Mans, R.S., Russell, N.C., Mulyar, N.A., van der Aalst, W.M.P., 2008. "Process flexibility: a survey of contemporary approaches", Advances in Enterprise Engineering I, Lecture Notes in Business Information Processing, vol. 10, Springer, pp. 16–30.
Sudhakar, G. P., 2012. A model of critical success factors for software projects. In *Journal of Enterprise Information Management*, *25*(6), pp.537-558.
Ter Hofstede, A.H.M., van der Aalst, W.M.P., Adams, M., Russell, N., 2010."Modern Business Process Automation: YAWL and Its Support Environment", Springer.
Trkman, P., 2010. The Critical Success Factors of Business Process Management. *International Journal of Information Management*, *30*(2), pp.125-134.
Van der Aalst, W.M, ter Hofstede, A., Weske, M., 2003. Business process Management: A survey. *Business Process Management*, Springer, LNCS 2678, pp.1-12.
Van der Aalst, W.M., 2009. Process-Aware Information Systems: Design, Enactment, and Analysis. *Wiley Encyclopedia of Computer Science and Engineering.*
Van der Aalst, W.M., Dustdar, S., 2012. Process Mining Put into Context. In *Internet Computing, IEEE*, *16*(1), pp.82-86
W3C-OWL-S, 2004. "Semantic Markup for Web Services", in http://www.w3.org/Submission/OWL-S viewed in December 2012
Weske, M., 2007. Business Process Management: Concepts, Languages, and Architectures. *Leipzig, Alemania: Springer-Verlag Berlin*, *pp.5-25*.
Workflow Management Coalition (wfmc), 1999. erminology & Glossary (WFMC-TC-1011) issue 3.0, Dec.2012, in standards of http://www.wfmc.org.
Wolf, C., Harmon, P., 2012, " State of Business Process Management", December 2012: www.bptrends.com
Zdun, U., Hentrich, C., Van Der Aalst, W.M., 2006. A Survey of Patterns for Service-Oriented Architectures. In *International Journal of Internet Protocol Technology*, *1*(3), pp. 132-143.

A Business Case Method for Business Models

L. O. Meertens, E. Starreveld, M. E. Iacob and L. J. M. Nieuwenhuis
Department of Industrial Engineering and Business Information Systems, University of Twente, Enschede, The Netherlands
lucas@lmeertens.nl, eelcostarreveld@gmail.com, {m.e.iacob, l.j.m.nieuwenhuis}@utwente.nl

Keywords: Business Case, Business Model, Method.

Abstract: Intuitively, business cases and business models are closely connected. However, a thorough literature review revealed no research on the combination of them. Besides that, little is written on the evaluation of business models at all. This makes it difficult to compare different business model alternatives and choose the best one. In this article, we develop a business case method to objectively compare business models. It is an eight-step method, starting with business drivers and ending with an implementation plan. We demonstrate the method with a case study for innovations at housing associations. The designed business case method can be used to compare and select the best business model successfully. In doing so, the business case method increases the quality of the decision making process when choosing from possible business models.

1 INTRODUCTION

Due to shortening product lives, intense global competition, a disruptive and agile environment, business models need to be renewed more rapidly and more frequently (Chesbrough 2007). In addition, the chosen course of action is of great importance for the future performance of organizations. With the renewal of business models, multiple possible directions can be defined. A recent example is seen in the automotive industry. Car manufactures need to choose if they want to produce cars running on alternative energy, and next, which type of energy. Hybrid, bio-fuel, electric, or hydrogen are all options. Making the choice is hard, for each of the alternatives require a business model change and the success of the produced car is unsure. This is an example of the need for a method to objectively compare alternative business models, and choose the best course of action.

A business case can be of help to form the answer to this question. A business case is a tool for identifying and comparing multiple alternatives for pursuing an opportunity and then proposing the one course of action that will create the most value (Harvard Business School Press 2011). Making a business case for the possible business model alternatives, gives the decision makers a solid and objective as possible basis, to make the best choice (Meertens et al. 2012).

Choosing one of the business model alternatives, should be well considered. Instead of a gut feeling, each of the alternative's consequences, impact, risks, and benefits for the organization, should be assessed as objectively as possible. This will result in a better choice, and better organizational performance.

However, the main problem is that it is unclear how alternative business models can be compared to choose the best course of action. A business case could be one of the solutions, for it compares alternatives in terms of costs, benefits and risks. Existing problems are that it is unclear how a business case should be made from a business model. Also, it is unclear what good business case components are, and which business model components are of relevance for the development of the business case.

2 METHODOLOGY

The research design is based on the design science research methodology (DSRM) by Peffers et al. (2007). This method is chosen because it creates an artefact as solution to a problem. In this research, the problem is the unstructured decision making of potential business models. The artefact designed is a business case method which enables objective comparison of business models. Further, the DSRM enables process iterations, so that it is possible to adjust previous phases to increase the quality of the artefact. However, because the review of academic literature is less emphasized, the method is adjusted to include the valuable academic literature in the process. For the literature study, the five-stage grounded theory method for rigorously reviewing literature by Wolfswinkel et al. (2013) is used. This method assures solidly legitimized, in-depth analyses of empirical facts and related insights, including the emergence of new themes, issues and opportunities (Wolfswinkel et al. 2013). Figure 1 shows the five sequential steps integrated with the DSRM method.

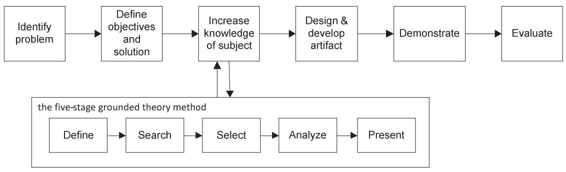


Figure 1: DSRM process of Peffers et al. (2007) with the grounded theory method from Wolfswinkel et al. (2013).

Starting with the first step of the DSRM of Peffers et al. (2007), the introduction to this article identifies the problem. Namely, the need to objectively compare business models. Following the DSRM, we identify the research objective: design a structural method to create a business case of business models, to be able to objectively compare the assessed business models, and choose the best alternative. We present the literature review of business cases and business models, which increases our knowledge on the subject, elsewhere.

3 THE BUSINESS CASE METHOD

This section creates a new artefact in the form of a business case method. The design of our business case method is based on the two approaches identified by literature review: Ward et al. (2008) and the Harvard Business School Press (2011). Both of them have a list of components. These lists partly overlap, yet each has distinct advantages and disadvantages. Based on the comparison of these two approaches, eight main components can be identified, which Table 1 lists.

In contrast to the business case method proposed by Ward et al. (2008), this method does take alternatives into account, similar to the model of Harvard Business School Press (2011). This is because in most cases more than one solution can be thought off and applied to reach the goal. Therefore, it would be bad to go with the first possible solution without putting some effort in the quest for other compelling solutions. Furthermore, the third point, alternatives, is different from the business case methods proposed in the reviewed literature, in which authors only look to the benefits that the proposal brings. Of course, the benefits are important for the business case. The possible negative effects, however, cannot be dismissed. Therefore, a good overview of not only the benefits but also the disadvantages should be presented in the business case as an overview of the caused effects of the proposed project. According to Ward et al. (2008), organizations who overstate the benefits to obtain funding are the least likely to review the outcome and less than 50% of their business case projects deliver the expected benefits resulting in unsatisfied senior management.

Table 1: Components of the business case method.

1.	Business driver	The cause, problem, or opportunity that needs to be addressed
2.	Business objectives	The goal of the business case stating which objectives are aimed for
3.	Alternatives	Representing the options to reach the objectives
4.	Effects	Positive and negative effects that come with the pursued alternative
5.	Risks	Risks that come with the pursued alternative
6.	Costs	Costs that come with the pursued alternative
7.	Alternative selection	Based on gathered data the best alternative is chosen
8.	Implementation plan	Plan which explains when and how the alternative is implemented

As the components are the main concepts of the proposed method, we clarify all eight components individually in this section.

3.1 Business Drivers

The meaning of the business drivers originates from the business case method by Ward et al. (2008) and has not changed. The business drivers stand for a statement of the current issues facing the organization that need to be addressed. These can either be problems or opportunities and ideas with enough potential to make it worth pursuing. Applied to business models, the business driver is most likely to originate from the need for business model innovation. Chesbrough (2007) argues that due to shortening product lives, even great technologies can be relied upon no longer to earn a satisfactory profit before they become commoditized. Practice has learned that even great business models do not last forever. Therefore, he argues, a company needs to think hard about how to sustain and innovate its business model. For future markets will be smaller, more highly targeted (and effective), and the new environment will require different processes to develop and launch products successfully.

3.2 Business Objectives

The business objectives are the goals of the innovation. Both methods discussed in the theoretical framework advice to set business objectives. They state which business drivers are addressed and how these are hoped to be achieved with the proposed project. This can be one or more specific aspects of the strategy that need to be improved or modified; one or more of the business model components that need improvement; or processes or products that need to become more efficient and better address the needs of customers.

3.3 Alternatives

The alternatives represent the available options to reach the objectives. At the start of this section, we describe the reasoning to include identification and assessment of alternative solutions in the method. Summarized, the argument is that it would be unwise to go with the first idea that comes along that addresses the business drivers, without investigating whether other, perhaps better, alternatives exist.

Sometimes, the benefits of a single specific opportunity or idea are assessed. In such cases, it might be hard to find a substitute or alternative to the opportunity. Thinking of alternatives and assessing them increases the chance of pursuing a better-balanced alternative, instead of the first that comes to mind. All alternatives need to be compared with the current situation.

Amongst others, identification of alternatives can be done by assigning a senior manager with the task to define and launch business model experiments (Chesbrough 2007). Harvard Business School Press (2011) proposes brainstorm sessions as a tool to identify alternatives. Both tools can be used to identify alternative business models. Next to those tools, market assessment tools or SWOT analysis may be suitable to come up with alternatives.

3.4 Effects

The effect component is the largest of all. This is because a variety of actions needs to be performed with the effects to create a consistent and structured overview of the effects on the organization per alternative. Effects are the positive (benefits) and negative (disadvantages) effects that an alternative causes. First, effects need to be identified. Second, it is important to come up with measures for each effect. Third, each effect must be connected to an owner. This increases involvement with the project within the organization, and stimulates owners of benefits to help establishing the alternative when it is approved. Fourth, each effect needs to be placed in the framework in Table 2 (Ward et al. 2008). For each effect, the framework determines the type of organizational change (do new things, do things better, or stop doing things) and the degree of value explicitness (from observable to financial).Fifth and final, a time frame is estimated per alternative. This time frame gives information of when the project starts, when it delivers results, and when it finishes. Each alternative goes through these five steps.

3.5 Risks

The fifth component is concerned with risk assessment of each alternative. Risk is defined as the probability that input variables and outcome results vary from the originally estimate (Remenyi 1999). How risks are assessed depends on the situation and needs further research per case. Amongst many others, the “best case/worst case scenario” method can be used to assess the risk of the alternatives. With this method, two scenarios are developed and the effects of each scenario on the organization are estimated. In the first scenario, the alternative will perfectly result in the expected benefits. In the second scenario, the worst reasonable possible situation will evolve caused by the alternative.

3.6 Costs

Costs are one of the most important aspects of a business case. The costs give an indication of the total expected investment costs, and expected profit over a specific time period. The investment costs represent the money needed to implement the business model change in the organization. Also, in the costs section, the expected payback time is calculated to indicate how long it will take for the break-even point is reached.

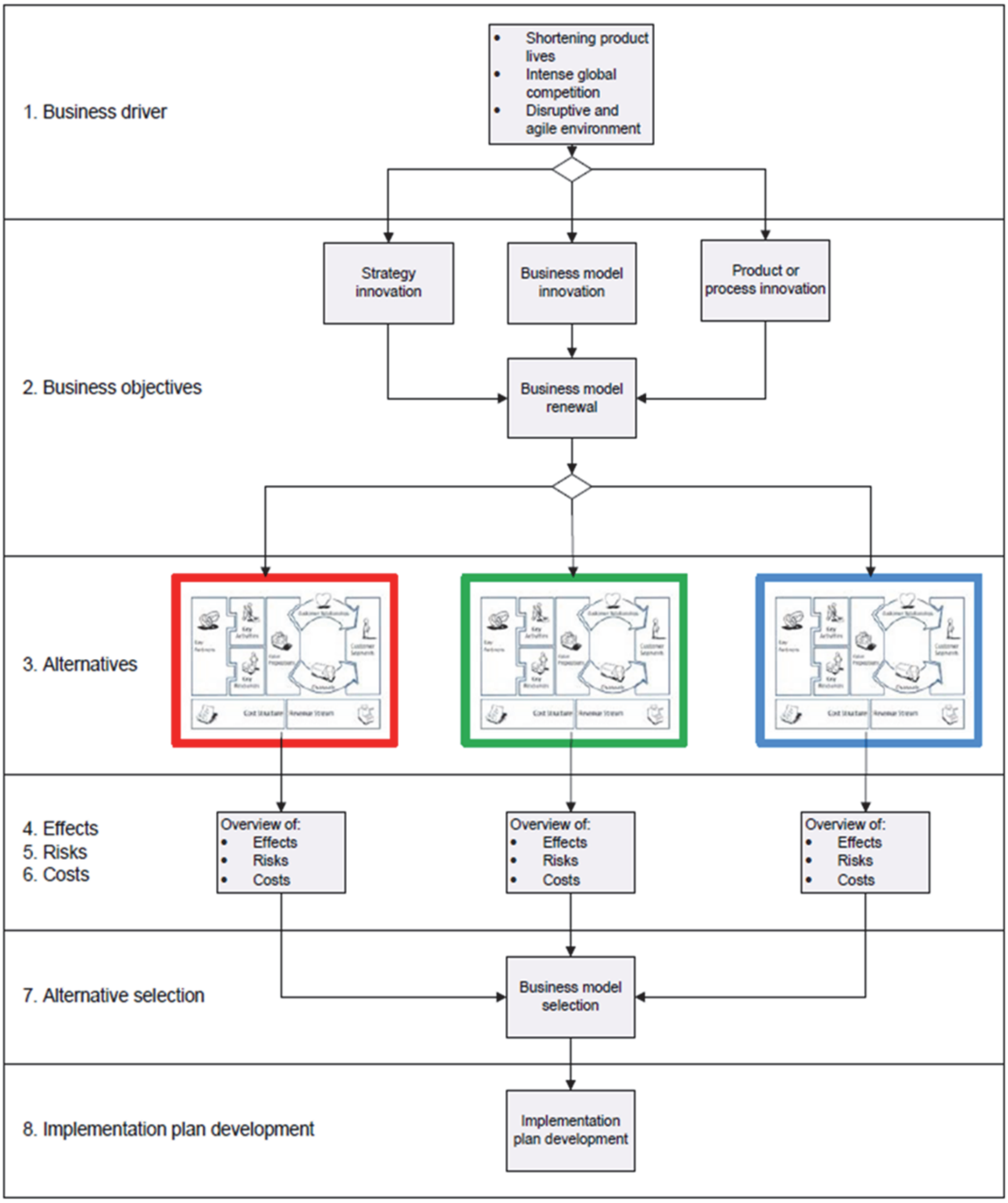


Figure 2: Business modelling connected to the business case method.

3.7 Alternative Selection

After gathering the data for all alternatives in the previous steps, the best option can be chosen by weighting the expected effects against the expected calculated costs. Harvard Business School Press (2011) suggests that the best alternative is partly chosen based on feelings. However, if the risks are translated into expected costs, this can be added to the costs-effect equation. Then the alternatives have to be compared based on the non-financial effects and the total expected costs/profit of the alternative. Many methods to do this exist, varying from complex to rather simple (e.g., the direct-rating method, point-allocation method, and analytical hierarchy process; Van Ittersum et al., 2004).

3.8 Implementation Plan

Now that the best alternative is selected, it is important to develop a plan of action. Tasks, roles, objectives, resources, dates, and responsibilities are parts of this implementation plan. The level of detail of an implementation plan varies depending on the case. The plan lays out how progress can be tracked and success measured when the proposed solution is put into action. Without this, actual success of a business case is hard to verify.

4 CONNECTING THE BUSINESS CASE METHOD TO BUSINESS MODELLING

In this section, the developed business case method is applied to the business model concept. Figure 2 visualizes the connection. The figure shows the business case steps on the left. The sources, types of information, or input for each of those steps are on the right.

The first step contains the business driver. Business drivers for business model innovation can come from different sources. In general, shortening product lives, intense global competition, and the disruptive and dynamic environment are the main sources (Chesbrough 2007). This can lead to one of the three causes for business model renewal. The business objective represents the goals that the business model change aims to achieve.

The next step is identification of alternatives. In this step, multiple business models can be developed with the focus on meeting the business objectives. Next, the effects, risks, and costs of each of the business model alternatives are assessed. The effects represent the positive and negative non-financial effects that alternatives cause. The effects can be represented with a framework for business case development (Ward et al. 2008).To assess the risks of the project, one of the risk assessment methods described in literature for project management can be used. The risk assessment part should at least cover the points of Remenyi (1999). The risk can be represented in a probability vs. impact matrix.

Often, the expected financial benefits, and the costs of the project, are the most important part for decision makers using business cases. In the costs section, changes in the business models costs and revenue component need to be assessed. The cost component of a business model must cover costs created in other components (Iacob et al., 2012), such as key activities. Next to the expected costs and profits, the payback period and return on investment should be presented.

Using a multi-criteria method, the most suitable business model can be selected in the seventh step. After that, an implementation plan can be developed. During step three till eight, alternative business models should be compared to the current business model to assess the changes and effects that it causes. For example, in the fourth step, only the effects that differ from the current business model are assessed. The reason for this is that the other effects remain the same for both alternatives, and thus only increases the size and complexity of the business case.

5 METHOD DEMONSTRATION AND EVALUATION: DEA LOGIC AND HOUSING ASSOCIATIONS

Having created the artefact (business case method), the next step is to demonstrate it. We use a case study of the company DEA Logic, which provides products and services for Dutch housing associations. The main two stakeholders in the case are the company DEA Logic and the Dutch housing associations. The innovation is developed by DEA Logic, and the target customers for this innovation are Dutch housing associations. The innovation will have an impact on the business model of the Dutch housing associations.

DEA Logic is an engineering company specialized in advanced electronics, security software, and consulting in information technology, information management, and building management. Over the last years, DEA Logic developed an access control system called C-Lock, which has a major position in their product portfolio currently. The C-Lock system can be extended with multiple solutions. This way, apartments can be better adjusted to the needs of the tenants. In this case, DEA Logic wants to discover whether their product is favourable for (Dutch) housing associations. A business case needs to be developed.

In the Netherlands, a housing association is a non-profit organization, which' mission is to build, manage, maintain, and rent houses and apartments. The responsibilities are defined and assigned by the Ministry of the Interior and Kingdom Relations. Each housing association is private, but can only operate within boundaries set by the Dutch government. Therefore, housing associations do not differ much. In addition, all housing associations have more demand than supply currently, which causes waiting lists. The houses they rent are favourable for citizens with a low income (an annual income of € 43.000 is the maximum). The associations are tasked to supply good housing possibilities for the relatively more vulnerable and poorer people in society. Similar constructions exist in other countries. For example, the United Kingdom has government-regulated housing associations with the same goal; to provide housing to people on a low income or people who need extra support.

Thanks to the public character of the housing associations, all needed information for this case is public and presented on websites of housing associations, the government, and the central fund for people housing. For the scope and purpose of this research, applying the DEA Logic case on Dutch housing associations in general is sufficient to demonstrate the designed method.

The data and numbers used in the business case are based on calculations by DEA Logic, and internet sources. For reasons of confidentiality, the numbers are not accurate. The business case gives an indication of the order of magnitude of the costs difference between the two discussed alternatives. If in the future, a housing association would like to realize the project, a new business case has to be made, to assess the effects of the innovation on their specific situation. For the purpose of demonstrating the business case method, the used numbers and accounted variables are sufficient.

DEA Logic develops technological and electronic innovations for real estate amongst others. The C Lock access control system is one of those products. The latest innovation for newly built or renovated apartment buildings is IP-infrastructure. In the current situation, each apartment in a building complex is supplied with public utilities and digital infrastructural connections. In the Netherlands, each apartment is provided with at least a telephone line, television cable, intercom system, and often fiberglass connection for internet. Each of these connections makes use of their own wires. The main idea of IP-Infrastructure is to supply each apartment with only one TCP-IP connection, combining telephone, television, intercom, and internet, as well as other possible data connections.

This infrastructure not only reduces infrastructural costs and materials of newly built or renovated apartments, but also increases the amount of possible functionalities. The currently developed functionalities are derived from the C-Lock access system, and can be connected to the receiver easily. Tenants can choose individually which solutions they need. The core of the innovation is to increase apartments' flexibility, functionality, and luxury, and to minimize the maintenance costs.

The C-Lock and IP-Infrastructure innovations by DEA Logic are suitable for Dutch housing associations, for they build, rent, manage, and maintain apartments for a diverse target group. The target group is diverse, as their customers are young as well as old people. In addition, families with children and people who need daily nursing support belong to the target customers. Introducing DEA Logic's innovations increases the suitable target group for each apartment, as it can be adjusted to the needs of the tenant more easily. Furthermore, the use of IP-infrastructure decreases maintenance costs.

The innovations affect the housing association's business model. Renting out C-Lock solutions and IP-infrastructure becomes a new key activity. DEA Logic becomes a new key partner, together with several service providers. Also the value proposition is extended, for apartments are more secure and luxury. The suitable customer segment for each apartment increases, as it can be adjusted to the needs of various tenants. Finally, a new revenue stream is added, for the IP-infrastructure is rented out, in combinations with C-Lock solutions, in addition to the traditional rent of apartments. Therefore, DEA Logic's innovation and Dutch housing associations form a good combination to test the business case development method.

The following eight paragraphs represent the eight steps of the business case method. We compare two scenarios. In both scenarios, the same apartment complex is built with one hundred apartments. The first scenario represents the current situation. In the second scenario, the IP-infrastructure is implemented together with C-Lock solutions.

5.1 Business Drivers

Based on the vision and strategy of the three largest housing corporations (CFV 2012), their mission is to build, manage, and maintain quality tenement housing for people with a low income and vulnerable groups in society. Therefore, it is preferable that building, managing, and maintenance costs of houses to be low. Housing corporations continuously seek possibilities to reduce costs and still deliver high quality, and affordable homes for a large and diverse target group. IP-infrastructure, in combination with the variety of possible C-Lock solutions provided by DEA Logic, is an innovation that contributes to the corporations' mission.

5.2 Business Objectives

In accordance with the business drivers, the pursued objectives of the IP-infrastructure presented in this business case are the following:

- Reduce maintenance costs
- Increase compatibility with target tenant group
- Increase quality of living environment
- Increase security of tenants
- Increase luxury

5.3 Alternatives

The yellow post-its in Figure 3 show the current business model of a Dutch housing association. The value proposition offers low-priced rental houses in a good living environment for people with low income belonging to vulnerable groups in society. Revenue is generated via monthly rent and subsidy from the government.

Figure 3: Business model of Dutch housing associations with IP-infrastructure and C-Lock solutions.

The blue post-its in Figure 3 are additions that show an alternative business model of a housing association with an apartment complex with IP infrastructure. In addition to the current key activities, renting out infrastructure and solutions form a new key activity. DEA Logic becomes a new key partner of the housing corporation, as they provide the solutions and maintain the system. Furthermore, the customer segments are extended with a target group including tenants who require special care. The fourth change is in the revenue stream building block. Next to the rent of houses and state subsidy, the housing corporations receive rent for the use of the IP-infrastructure by tenants.

Next to changes visible in the business model, many benefits of IP-infrastructure are within the tactical set of the current business model (Casadesus-Masanell & Ricart 2010). Therefore, they do not influence or change the business model. However, the resulting business case includes those effects as well.

5.4 Effects

Implementing IP-infrastructure in renovated or newly build apartment buildings affects the organization. Table 2 presents the effects of the new IP-Infrastructure compared to the current, classic infrastructure. The table structures them according to two factors. Horizontally, they are categorized according to the type of required organizational change. Vertically, they are categorized according to the degree of explicitness. Because the only difference between the two alternatives, in terms of business model, is the revenue model, other effects of both alternatives are equal. Therefore, they are represented in only one effects overview table.

Table 2: Effects of IP-infrastructure

Degree of explicitness	Do new things	Do things better	Stop doing things
Financial	Rent C-Lock solutions and IP-infrastructure		Reduce maintenance costs by not replacing door locks & nameplates
Quantifiable			
Measurable		Increased target group Increased security	
Observable	Dependable on non-standardized technology In line with mission and vision	Increase quality living environment	

5.5 Risks

As with each innovation, risks are involved. To assess the risks, we use a construction project risk assessment method (Tah & Carr 2000). This method is suitable, as renovating or building the apartment complex is a construction project. Most risks can be prevented, resulting in a very low overall project risk. However, some risks of the IP-infrastructure alternative remain, due to the following two points:

1. The technology is new. So far, it has been deployed in one apartment building only.

2. The technology is developed and built by one company. The current market does not provide any substitutes that work with the same infrastructure.

These two points are interconnected. A small change exists that the technology does not work as good as was hoped for, or the subcontractor stops supporting the technology. In that scenario, the costs to transform the infrastructure back to the current standard are high. Other risks for both alternatives can either be prevented, or do not have a negative influence on the organization. The total risk of IP-Infrastructure, before prevention, is one and a half times the risk of the classic approach. This is mostly because the classic infrastructure is used almost everywhere and has been improved over time.

5.6 Costs

The cost difference, between the current situation and the IP-Infrastructure alternative, depends on two variables. First, the number and type of C-Lock solutions affect the costs. The second variable is time. Time is important, as the housing association's objective is not only to build apartment complexes, but also to maintain them. Therefore, the cost overview also includes maintenance.

To compare the costs of both approaches, an indication of the costs for an apartment complex with 100 apartments is calculated. Only the costs for the infrastructure and the C-Lock solutions are covered. The other building costs are equal for both alternatives. Because the costs for construction and maintenance of the infrastructure and the C-Lock solutions vary from situation to situation, several assumptions and raw cost estimates are used.

Table 3 shows estimates of construction costs, yearly maintenance costs, and yearly profit, per function. Next, the maintenance costs and profits are extrapolated over five years to get more insight in the breakeven point of the alternatives. The initial costs for the IP-Infrastructure are higher compared to the current situation. However, the difference is not very big, and within three years, the IP-Infrastructure in combination with the access C-Lock solution is cheaper than the current alternative.

Table 3: Estimated costs of construction and maintenance, and estimated profit.

Function Costs (€)	Infrastructure		Access		Intercom		Care		Communication	
	Old	New	Old	New	Old	New	Old	New	Old	New
Construction (Initial)	13,000	26,000	30,000	30,000	52,000	50,000	800	400	-	-
Maintenance (Yearly)	500	1,000	11,250	6,950	16,500	7,000	3,600	1,800	750	0
Profit (Yearly)	-	-	-	-	-	-	-	300	-	-

5.7 Alternative Selection

The effects, risks, and costs of IP-infrastructure, compared to the classic infrastructure, are discussed in the previous sections. Based on this information, one of the alternatives needs to be selected. Looking at the effects, IP-infrastructure is the best choice as it increases the amount of target groups, quality of living, and security of tenants. Additionally, with the new technology, apartments become more luxury. The risks, however, are one and a half times higher than with classic infrastructure. Again, this can be reduced using available risk prevention options.

Initial costs of IP-infrastructure are higher, but within four years it becomes cheaper than the classic alternative. Depending on the functions, the estimated IP-infrastructure savings are around € 70.000 after five years. Initial costs are higher, yet maintenance costs are much lower.

IP-infrastructure offers new functionalities and increases security of tenants, quality of living, and target group. Risks are higher, but can be prevented. Initial costs are higher, but money is saved due to the low maintenance costs over time. Therefore, IP-infrastructure is the best alternative to choose.

5.8 Implementation Plan

After their board of directors approves this project, the housing association can implement the project. In this phase, however, it is too far stretched to determine an explicit implementation plan.

6 DISCUSSION

The objective for designing the business case development method to compare business models was to design a method to create a business case of business models, to objectively compare the assessed business models, and choose the best alternative. Because of the abstract descriptive nature of business models, it is often required to involve more tactical and operational details, only implicated by changes in the business model. Deciding which details are useful and which are not must be judged by the maker of the business case. This allows for a certain amount of subjectivity. Table 4 represents which method steps are objective and which are open for subjectivity.

During creation of the business case, one of the experienced difficulties was switching between abstraction levels. A business model is an abstract representation of an organization. Processes and products are on a more tactical or even operational organizational level. The outcome of comparing business models in the business case depends on choices made in organizationally lower abstraction levels, like the tactical and operational level. The distinction between a process or product business case, and a business model business case needs to be made. In the first case, focus is on cost and benefit comparison of the innovated process or product. In the second case, it is about choosing the best alternative way of how an innovated product or process affects the business model.

Table 4: Assessment of the objectivity of the business case method.

Method step	Objective / Subjective
Business driver	Objective
Business objectives	Objective
Identification of alternatives	Subjective
Effects	Subjective
Risks	Subjective
Costs	Objective
Alternative selection	Objective / Subjective
Implementation plan	Subjective

Furthermore, we found some empirical evidence supporting the “strategy – business model – tactical set” framework by Casadesus-Masanell & Ricart (2010). In hindsight, the case study is mostly a product innovation within the tactical set of the building association’s business model. Some minor changes were made in the business model. This made it hard to devote the business case to the business model, and forced us to include more operational aspects in the business case. This is not per se negative for the demonstration, the method, or the outcome of the business case, but the goal and focus of the designed method, is to objectively compare two business models, in contrast with assessing the costs and benefits of a product innovation.

A limitation of the research is due to an almost complete lack of academic literature about business cases. The concept is used often, but without a well-designed and widely accepted methodology. As well as for the business model concept, it would have been better if a general accepted business case development method would have existed in academic literature for the reliability thoroughness of the research.

Overall, the method does what it is designed for. It is a method to develop a business case, which allows different business models to be compared, and the best one to be chosen as objective as possible.

7 CONCLUSIONS

The designed business case method to objectively compare business models can be used to compare and choose the best business model successfully, as demonstrated by the case study. The goal of this research was to increase the quality of the decision making process between possible business models, by developing a method to objectively compare the alternatives. Based on literature research, the business case method was designed. This method contains the eight components that Table 1 lists.

The case study showed that the developed method can be used to compare business models and choose the best one. However, the output of the business case depends partially on the people making the business case. Steps 3, 4, 5, and 7 are relatively subjective steps, which gives freedom to decision makers. Further research is needed to establish the effects of this decision freedom on the quality of the outcome of the business case. Still, the method fulfils the defined goal of the research.

ACKNOWLEDGEMENTS

This work is part of the IOP GenCom U-CARE project, which the Dutch Ministry of Economic Affairs sponsors under contract IGC0816.

REFERENCES

Casadesus-Masanell, R. & Ricart, J.E., 2010. From Strategy to Business Models and onto Tactics. Long Range Planning, 43(2–3), pp.195–215.

CFV, 2012. Corporatie in perspectief 2012, Baarn: Centraal Fonds Volkshuisvesting.

Chesbrough, H., 2007. Business model innovation: it’s not just about technology anymore. Strategy & leadership, 35(6), pp.12–17.

Harvard Business School Press, 2011. Developing a business case : expert solutions to everyday challenges., Boston, Mass.: Harvard Business Review Press.

Iacob, M.E., Meertens, L.O., Jonkers, H,. Quartel, D.A.C., Nieuwenhuis, L.J.M. Sinderen, M.J. van, 2012, From Enterprise Architecture to Business Models and back, Software and Systems Modeling journal, December 2012, DOI10.1007/s10270-012-0304-6, Springer.

Ittersum, K. Van et al., 2004. A multidimensional approach to measuring attribute importance. Advances in consumer research, 31, pp.86–87.

Meertens, L.O., Iacob, M.E., Nieuwenhuis, L.J.M., 2012. A method for business models development, In B.

Shishkov (editor), Business Modelling and Software Design, Volume 109 of Lecture Notes in Business Information Processing, pages 113–129. Springer Berlin Heidelberg.

Peffers, K. et al., 2007. A design science research methodology for information systems research. Journal of Management Information Systems, 24(3), pp.45–77.

Remenyi, D., 1999. Stop It Project Failures through Risk Management, Newton, MA, USA: Butterworth-Heinemann.

Tah, J.H.M. & Carr, V., 2000. A proposal for construction project risk assessment using fuzzy logic. Construction Management and Economics, 18(4), pp.491–500.

Ward, J., Daniel, E. & Peppard, J., 2008. Building better business cases for IT investments. MIS quarterly executive, 7(1), pp.1–15.

Wolfswinkel, J.F., Furtmueller, E. & Wilderom, C.P.M., 2013. Using grounded theory as a method for rigorously reviewing literature. European Journal of Information Systems, 22, pp.45–55.

Business Processes, Process Logic and Information Architecture
A Tentative Case Study

Coen Suurmond
RBK Group, Keulenstraat 18, 7418 EP Deventer, The Netherlands
csuurmond@rbk.nl

Keywords: Process Logic, Information Architecture, Organisational Semiotics, Business Processes.

Abstract: Subject of this paper is modelling the process logic of the business processes of an enterprise, taking into account both the formal and the informal aspects of the organisation, but disregarding how the business processes have developed over time in operational practice. The aim is to arrive at a stable information architecture that has sufficient flexibility to absorb developments in the environment and in the enterprise itself in the presentation layer and business rules, without affecting the information architecture itself.

1 INTRODUCTION

Subject of this paper is the modelling of the process logic of the business processes of an enterprise (here named AYS), taking into account both the formal and the informal aspects of the organisation, but independent of how the business processes have developed over time in operational practice. The aim is to arrive at a stable information architecture that has sufficient flexibility to absorb developments in the environment and in the enterprise itself in the application software that will be built upon it. Parts of this approach have been described in earlier ICEIS and BMSD papers, in particular, papers presented at ICEIS 2010, BMSD 2011, and BMSD 2012. This paper will present the preliminary results of the application of this approach to a practical case.

After this introduction, the second part of the paper describes the problem and the backgrounds to the issue. After a short explanation about the theory of the firm and the role of semiotics in the case study the role of the analysis and modelling of business processes as foundation for the development of an enterprise information system is explored.

The third part describes the backgrounds of the enterprise where the study was carried out as well as the most significant characteristics of the enterprise itself. This is followed by a short description of how the study was carried out and in particular of the interaction between analyst and practitioners in such a project.

The last part of the paper is the evaluation of the results of the case study against several aspects. Especially the process logic for the internal structure of the business processes and the concept of lean IT for the efficiency of the business processes are important here.

2 MANUSCRIPT PREPARATION

2.1 Problem

The problem in this case study is the design of a solid foundation for a newly to be developed information system for the enterprise concerned. The main aims for the enterprise are (1) to be able to replace the current software package in the short term without loss of essential functionality, (2) to expand the new system in the slightly longer term to provide the desired support for the enterprise's business processes and (3) to be capable of supporting possible strategic scenarios (of which it cannot be determined in advance if and when they will occur) at some later date.

At first, the problem demands an information architecture based on both the actual processes and on the new processes envisioned in the strategic scenarios. Meanwhile the architecture must allow the implementation of just a number of key functions at first to allow full decommissioning of the current system. An essential feature of the information architecture must be the maintainability of the business rules in a number of areas because the rules imposed by external stake holders are subject to sudden and rapid change. Complying with these rules is required to operate in this line of business.

The main idea behind the case study is that the stability of the desired information architecture is determined by its autonomy from chance factors and passing circumstances. In other words, the main idea is that the essential and durable structure of the business processes should form the foundation of the information architecture. This introduces the question how this stability can best be found. This approach presumes that the characteristics of markets and products determine the essential structure of the business processes for an enterprise. To be active in a certain market, the enterprise has to follow a number of social conventions that are associated with the market and that place norms on the behaviour of the individual enterprise in the market. The same holds for the products of an enterprise, for both material and immaterial products. Of course, for material products a number of physical rules and constraints apply as well, such as food safety requirements in case of food products.

The idea is that hard statements can be made regarding the structure of the business processes and the associated information flows based on knowledge of the norms that apply for markets and products.

An additional motive to start the analysis of the structure of business processes with the markets and products is that this provides a better foundation for the collaboration between analyst and practitioners than the analysis of the current business processes of the enterprise. This will be explored further in a later paragraph.

2.2 Earlier Work

For the case study, we will rely on earlier theoretical work, as presented at ICEIS 2010 and also at two editions of BMSD, namely BMSD 2011 and BMSD 2012 (Suurmond, 2012; Suurmond 2013), and we will also rely on a long-term involvement with the Organisational Semiotics Community as well as on extensive experience in the design of information systems for the food processing industry. However, this case concerns an electro-technical reparation enterprise and thus presents an interesting case for the transfer of practical experience between two very different lines of business.

2.3 Theory

2.3.1 Theory of the Firm

An enterprise derives its existence from successfully delivering products to its markets. The two basic requirements for sustainable business are market demand and efficiency of production. Every successful enterprise also has a form of 'uniqueness' that distinguishes it from its competitors and that cannot be copied (Kay, 1993). This unique and idiosyncratic character of an enterprise determines its place on the market and can be found in partly intangible factors such as company culture, history and market trust or reputation. These factors can indirectly be found in the company culture and directly in the way in which individual employees are dealing with individual cases in the business processes. The latter is subject to acculturation processes, with conscious and unconscious, designed and historically grown mechanisms by which individuals learn "how things are done here".

This approach to the enterprise indicates that how an enterprise operates and the operations within an enterprise always have to be evaluated in light of its position in the market. This does not mean that the contribution to the market position is the only norm; there are inescapable human and societal norms after all. It does highlight that it is essential for the continuity of the enterprise that the market is the ultimate standard against which it is evaluated. This holds for operational actions and it holds as well for the actions taken by its management and for its strategic choices. Therefore, in analysing business processes and in designing an enterprise information system to support those business processes the orientation on the markets and products of the enterprise should be the first criterion.

From the above considerations it follows that the metaphor of the enterprise as an organism is more appropriate than the rationalistic and mechanistic approach of the enterprise (De Geus, 1997). After all, an enterprise is a social phenomenon in which the actions are determined by social norms and by interpretation processes. This means that modelling business processes and information flows from a purely rationalistic-mechanistic view or weakening the strengths of an enterprise by reducing the number of possible solutions in the business processes have to be avoided in the development of an enterprise information system.

2.3.2 Semiotics

Social communication happens through sign systems and the interpretation of signs is partly determined by history (the way in which signs were interpreted in the past) and partly by context (and sometimes by the way in which they are uttered as well, a certain inflexion of the voice for example).

Within business processes the efficiency requires that much of the information can be processed by systems. The sign systems created to this end are of a formal nature: the meaning of variables and of possibly of value ranges is recorded in the systems in advance.

Within an organisation all kinds of capacities in which information can appear can be distinguished. Part of the information can be found in computer based systems, part is 'between the ears' by training, knowledge and experience and part is exchanged through all manners of ad hoc communication. The nature of the sign system determines the possible interpretations of the information given. In part because of the degree of formalisation.

Although semiotics remains in the background in the case study, semiotic insights certainly play their part in the analysis and modelling. This is especially evident in the meaning of sign systems and of interpretation processes in both the analyses and in the business processes. It is also visible in the prominent role played by social norms, in particular in understanding business processes against the background of the normative function of the markets and products of the enterprise (Stamper, 2000; Liu, 2000).

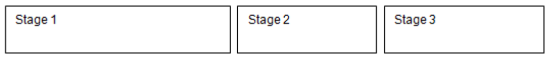
2.3.3 Process Modelling and Information System Development

Modelling business processes with the associated information flows, and validating the resulting model, is a communal activity of two different kinds of actors, each with a completely different background. On the one hand there is the analyst with communicative, analytic and modelling competencies (accustomed to formalised sign systems), on the other there is the practitioner with a detailed knowledge of what happens in practice, of organisational structures and procedures and equipped with lots of tacit knowledge.

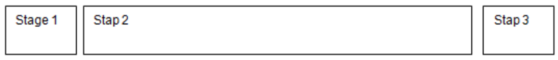
The difference in perception and background of the different actors cannot be bridged by the analyst transforming himself into a practitioner (or vice versa). As well as the time such a transformation would cost, it would mean a fundamental lack of recognition for the difference between the role of the analyst and the role of the practitioner. It might seem tempting to unite all of the required knowledge and experience in one person, but it would imply a major risk of consigning the process of modelling and analysis to the realm of tacit knowledge, with pernicious consequences for validation and maintenance of the model. In effect it would be a one man show.

The model that is to be constructed of the business processes and the accompanying information flows should represent the essential structure, thus forming a stable basis for the information system that is to be developed. As a model it is an abstraction and not 'true' or 'false', but suitable to a greater or lesser degree for the purposes for which it was developed. A basic condition is stability: it should be possible to support all kinds of variations of the business processes by one single model. A second condition is the reduction of complexity: the model should enable insight into the complex reality of concrete business processes by omitting all kinds of details that are irrelevant to the structure and by naming the separate elements of the processes.

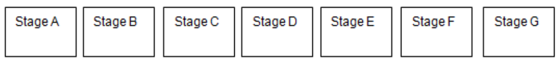
An abstract example of one aspect of modelling: say that a certain production process moves through three different steps and that these steps are modelled as they are observed in practice:



Later, the process is changed and the model with it:



However, if the process elements had been analysed further, the following model could have been the outcome:



In this last model with stages A through G both process variants could have been represented. Before the changes to the process stages A, B and C form the first sub-process, D and E the second and finally F and G the third sub-process. After the changes the sub-processes encompass different stages (A, B through F, G), but the basic model remains the same.

The major challenge is to distil those elements A through G from the concrete business processes with all of their details. It is not unusual to start from interviews with practitioners from different layers of the organisation combined with the analyst's own observations of the processes and information products. Often, documents regarding the organisation and those of the processes regarding quality control are important sources in arriving at an understanding of the processes. However, in practice this springs a number of problems. The first issue is the degree to which the formal documentation of the organisation and its processes agrees with the organisational reality. Giving prominence to these documents implies taking a position regarding the value of these documents, either negatively ('worthless paper truths of the managers') or positively ('we are trying to act in this way, but it was not possible just now'). In both cases the formal documentation is the leading norm in taking stock of and evaluating the processes.

The second issue is the effectiveness and efficiency of the Interviews with practitioners. On the one hand the analyst can drown in details; on the other essential elements of the business processes might remain undiscussed. The analyst does not know they are there, while to the practitioner they are so obvious that it does not occur to him to mention them. The same holds for looking into the information products. How does the analyst find out what is not there, what is left out because it is supposed to be known or because the information is obtained by other means?

Another approach is working from the underlying norms of the enterprise. This begins with an orientation on the markets and products of the enterprise. After all, the enterprise exists because it creates products for customers and this is given shape in the business processes. The organisation (and quality control) has to structure and stabilise the business processes, but that should happen to serve the higher purpose: to effectively serve customers in an efficient manner. Needless to say, other essential norms apply that lie outside of the enterprise. Those are in part societal norms and in part norms from specific stakeholders such as regulations by the government (requirements for the financial accounts are a striking example).

The norms that are based on markets, products and external stakeholders are in general more stable and accessible than all the ins and outs of the business processes (especially when the analyst has to work his way through lots of details before isolating what is structural from what is irrelevant for his purpose). On top of this, possible developments in those norms are essential for the internal structure of an enterprise. Those can be developments as a consequence of strategic decisions by the enterprise or external developments that the enterprise has to follow if it wants to remain in the market. The model of the business processes should be capable of following those developments without major structural changes.

2.3.4 Ontology and Ideal Type

Through the process logic the essential and stable elements of business processes and information flows should be mapped. This aim can also be distinguished in several ontological approaches. Essentially, the process logic is used to define a small and specific universe of discourse along with the associated operations. Using a classification of Poli the process logic could be placed under the term formalized ontology: "...to find the proper formal codification for the constructs descriptively acquired..." (Poli, 2010), with the essential difference however, that the intended constructs are not obtained by means of "collection of such prima facie information on types of items either in some specific domain of analysis or in general" (Poli, p2), but by a normative and critical analysis of the enterprise against its background of its products and markets.

The use by Dietz of the term ontology points in the same direction: "Our goal is to understand the essence of the construction and operation of complete systems, more specifically, of enterprises" (Dietz, 2006). In a very different time and against a very different background Max Weber was searching for a precise and consistent description of social patterns and their backgrounds in his main work Economy and Society: "In order to give a precise meaning to these terms, it is necessary for the sociologist to formulate pure ideal types of the corresponding forms of actions which in each case involve the highest possible degree of logical integration by virtue of their complete adequacy on the level of meaning" (Weber, 1968).

A marked difference between the ontology approach as used in ICT and the use of the concept of an ideal type of Weber is the way in which the resulting model is viewed. Is it a basic design to engineer the social world towards what it should be or is it an instrument to understand patterns of rule-based human action in a specific context? The thinking behind the former idea is formulated clearly by Dietz: "Contrary to many dissenting theories that have been advanced in the past century, organizations are artifacts. They are systems that are, and have to be, designed and engineered, like any other artifact" (Dietz, 2006). The latter is expressed by Weber in two ways, directly following the earlier quote: "…it is probably seldom if ever that a real phenomenon can be found that corresponds exactly to one of these ideally constructed pure types." (Weber, 1968) and "The more sharply and precisely the ideal type has been constructed, thus the more abstract and unrealistic in this sense it is, the better it is able to perform its functions in formulating terminology, classifications, and hypotheses" (Weber, 1968).

The approach of using the process logic as a means to arrive at an information architecture shares characteristics with both of the above approaches. The concept of process logic is based on a Weberian idealisation and it is based on an analysis of the underlying norms of human action. The information architecture that is based on process logic is an especially good example of organisational engineering: a formal and consistent model of the essence of the business processes in an organisation. However, since the organisation as a social phenomenon is anything but an engineered system, but rather an emergent system that is continuously changing itself, the information architecture is not a prescription to how the organisation ought to behave. It works in the reverse direction: when the organisation behaves and develops itself as described by Taylor and Van Every (Taylor, 2000) and when the actions of the organisations are at the same time determined by a number of inescapable rules, then it has to be possible to represent those matters within the capricious daily organisation reality that are essential to the business in the information architecture.

3 DESCRIPTION OF THE CASE

3.1 Introduction

AYS is a leading service and repair business for mainly audio-visual equipment of major brands operating nationwide. The enterprise carries out both on-site and carry-in repairs and has a network of six branches for the on-site and smaller carry-in repairs that service the different regions of the country. Larger carry-in repairs are performed centrally in Arnhem. The main contract partner is a leading brand (represented by its national importer), AYS is a certified partner and carries out all repairs in The Netherlands for audio-visual equipment of this brand. AYS is also active on a smaller scale in the repairs of other brands and of other kinds of electrical consumer products.

The key elements of AYS are:

- Both on-site repairs and carry-in repairs of audio-visual consumer products
- National coverage with six regional branches
- Strong affiliation with a strong brand
- About 100 employees

3.2 Structure of AYS

The legal structure and the structure of the business processes are rather different at first sight. AYS presents itself to the outside world as one homogeneous company with a specific service package. There is also a strong centralisation in terms of management and strategy; the head office defines the corporate identity and determines how the business is done. Legally there are a number of different entities (each a separate legal person) on three levels:

- Level 0: The holding
- Level 1: The main office and multiple entities that are not involved in the servicing and repairs and that will not be discussed here
- Level 2: The regional branches

The main office encompasses a number of central services, the main workshop with reception desk for carry in service and it provides the on-site service in its region. The regional branches provide the on-site service in their regions and they have a limited workshop facility with a limited reception desk service. The regional branches are either full subsidiary companies or fully owned by an independent entrepreneur.

3.3 Contracts, Agreements, Commitments

Curiously, there is only a very limited use of formal SLA's. The affiliation between the importer and AYS has much more the nature of a relational contract in which the details of the mutual obligations are not described as much as it is based on trust, established practice and, especially, on the binding effect of the settlement of financial claims of work carried out by AYS that are accepted or declined by the importer depending on the circumstances (circumstances that are not always known to AYS). Here, it is clear that this is not a symmetrical relationship; it is the importer who leads the way, who determines how matters must be handled both materially and financially. In practice, there are a multitude of agreements and expectations regarding the handling of repairs (turnaround times, success rates) and regarding the handling of the financial side. Current practice is mainly based on the knowledge and experience of a number of key figures in the AYS organization (which is both a weakness and a strength; a weakness because of the dependence on individuals, a strength because it is hard to reproduce and thus cannot easily be adopted by others outside of AYS).

For other products groups and brands the same pattern holds and the size of the contract partner is there too defining for the (a)symmetry of the relationship between AYS and its contract partner.

3.4 Strategy

AYS has a growing strategy in two directions. The first direction is diversifying the brands. Because there is a strong current dependence on one brand, AYS is investigating the possibilities to apply the current competencies for audio-visual consumer products to different brands. Potential new activities are not foreseen to demand new processes. However, it is possible that agreements and interactions with new parties will take on new forms (but that also holds true with regard to the current clients).

The second direction is to use the competencies and the nationwide network for new activities, in particular services to professional users. Potential such activities are the servicing of permanent audio-visual installations, both for companies to whom that is the core process (informing and/or advertising to its customers) and for companies where it concerns more internal presentation capabilities. A different possibility is to provide the entire handling of defect equipment for larger retailers (logistics service partner). Another possibility is to provide installations of new audio-visual equipment to professional users. Currently, there are some small-scale activities in these directions and growth towards full scale services is a real possibility.

3.5 Stakeholders

In principle AYS is dealing with one, two or three external parties and with one or two internal parties for one repair job. The external parties are the end user (usually a consumer but it can be a company), the direct supplier (big chains of nationwide operating retailers), and the importer as representative of the brand. The client of AYS is one of these three parties and the details of the ‘preceding parties often have to be registered as well (the consumer has two preceding parties and the importer has none). The contractor is one of the AYS branches which can subcontract the work in whole or in part to another branch of AYS.

Each stakeholder has its own way of providing and requesting (or demanding) information and of tracking the work and handling the financial side. Moreover, these patterns are subject to unpredictable change. The use of references by the stakeholders is also erratic. Standards for dealing with warranty conditions and for the execution of work differ per stakeholder. Market and power relations determine who is in charge, and as a smaller party AYS usually has to comply with the demands and expectations of the (much) bigger clients. Here, logic and facts can sometimes be set aside. The flexibility with which AYS deals with these complex and rapidly changing practices is an essential factor for the internal costs and for successfully getting the remuneration for the performed work.

NB: The term ‘customer’ is difficult to apply in the case of AYS, because there are so many kinds. Because of this, the term will be avoided as much as possible.

4 SOME PRACTICAL ISSUES IN THIS CASE STUDY

4.1 Creating Common Background between Analyst and Practitioner

The analysis and modelling took place in a series of open conversations and presentations with discussions with two of the three executive managers / owners. As indicated the aim was to arrive at a stable information architecture for the enterprise. The stability of the architecture requires that it is based on the underlying lasting patterns of the business processes, as well as on an understanding of the markets and products, trends and strategic scenarios. At the same time practitioners will take a perspective based on their everyday work and will mainly be focused on their current operational obstructions. To them, the benchmark for the description and model of the business processes is their daily practice, as it should be. However, at the end of the process of analysis and modelling the analyst should have a sufficient grasp on the operational processes and the company culture, while the practitioners should have a sufficient grasp of the abstracted view of their processes. Without this resulting communal basis it will not be possible to discuss and evaluate the result of the modelling in a fruitful manner.

The background to all of this work is formed by the norms imposed by the external stakeholders, the norms imposed by the nature of the products and the norms that originate from the enterprise itself. It is up to the practitioners to indicate these norms and it is up to the analyst to formulate these norms in a precise manner and to continuously test these norms against the background of the business environment. Here, it will often prove necessary to adjust the way in which the norms are formulated; either by adjusting the norm itself or by adjusting the circumstances under which the norm applies. One of the results of the analysis of the norms is that it becomes clear which norms are hard with hard conditions and context (and thus suitable for machine interpretation) and which norms are either ‘soft’ or significantly subject to circumstances (and which thus involve direct human interpretation and responsibility in applying the norm to a concrete case).

4.2 Rigid Principles Bring Practical Solutions

The purpose of this phase of analysis and modelling was not to solve current problems (other than the problem of replacing the current software package which could not be maintained), neither was it to evaluate and to take stock of the demands and wishes of AYS regarding the new information system. Because of this it was remarkable to see the following pattern emerge at a number of times (especially in the latter stages):

- Within the process logic a sharp distinction is drawn between two processes
- The practitioners react at first by projecting their view of current practice onto the model. This can result in an initial negative reaction: we do not recognise our processes!
- Next, a discussion emerges about the correctness of the formulated model against the background of the norms within the organisation (mainly in the area of responsibilities for the end result and for the costs) and those outside of the organisation (what do the external stakeholders demand)
- Sometimes the discussion results in adjustments to the norms and/or to the model (but not in most cases)
- Finally, the practitioners conclude that by working according to the formulated model a number of current practical problems will be solved, because those problems are the result of a muddying of the boundaries between two processes

An important example of this pattern is the introduction of the concept of transfers combined with the concept of a process step and the assignment of a service order to a single branch. One of the principal norms in the enterprise is that of turnaround time. External stakeholders link the remuneration for a repair to the meeting of the agreed upon turnaround time. Thus, the monitoring of turnaround times is a crucial component to the internal monitoring of the service orders. Because of the current out-dated information system, but also because of how the work floor is organised, this monitoring is at present a cumbersome and vulnerable process. Building software that supports the current practice would likely be a major task, with lots of maintenance afterwards.

By making the current practice explicit by modelling a number of successive process steps (receipt – administrative screening – technical screening – actual repair – preparation for shipping – shipping) and by explicitly naming the transition between process steps as a transfer, it becomes a trivial job to evaluate both internal and external turnaround times per process step in the new information system. The explicit transfer between process steps also improves organisational clarity: who is responsible for the service order? If desired, it is also possible to split the transfer in two: making it available to the next process step and the acceptance by the next process step. This system also allows for easy monitoring in such a way that internally accumulated delays can be compensated for by adjusting the turnaround time for the next process step for the particular case.

An accompanying concept to this concept of transfers is that of “on-hold” situations. When a service order is waiting for activities outside of the control of the relevant process step, the order will be on-hold. Here, it is recorded why it is on-hold, by whose actions the on-hold situation will be lifted, when this is expected to be the case and if the on-hold situation causes the turnaround time to be suspended.

And while this model was completely new to the practitioners, it cannot be called an invention by the modeller either. It was just giving shape to and formalising something that was very close to the surface of the business processes, but which was not viewed as such up to now. Rigorous modelling of the process steps, transfers and on holds led to clearly definable administrations and responsibilities, and less complex business processes.

5 EVALUATION

5.1 Process Logic

In this paper the term process logic has been used to distinguish it from the idiosyncratic characteristics of the enterprise. Usage of this term was founded in a number of considerations. First, it deals with a schematic representation of the inevitable structures within a certain line of business, valid within a specific social environment. One might say that these are the structures a student should be taught, while he does not yet know which specific company will employ him. Second, norms for completeness and consistency hold for this schematic representation. On the level of abstraction chosen, it should be capable of representing every scenario that arises in practice (a tall order and a real challenge!) and there should be no inconsistencies or ambiguities. This demands definitions of the elementary terms and a precise formulation of the underlying norms.

In applying a concrete model of process logic it is essential to realise that it is an instrument to represent situations and processes (description, and an instrument for analysis) and that it not intended to be used to dictate how processes and situations ought to be (prescription). At its core, the process logic is a formalised sign system to (1) gain an understanding of the processes in the analysis, (2) precisely formulate terms and rules and (3) describe an information architecture that because of its character forms the basis for later system development. At the same time, process logic has to help the enterprise avoid inconsistencies (for example by preventing the use of key terms such as “service order” to mean various things) and leave the enterprise to choose how it sets up and executes its processes. An enterprise with five experienced employees will have to organise itself very differently from an enterprise with 5 offices, each with 20 employees and 10 flex workers!

In the case at hand the attempt to uncover the process logic has worked well, both to establish a common background between the analyst and the practitioner and to arrive at precise definitions, rules and demarcations.

Process logic is an important element for a common background, because it is a shared search for the underlying structures. For the analyst a general orientation on the specific markets and products of the enterprise with its peculiarities combined with a general background and common sense is sufficient to play his part in the discussions. All kinds of details that are hard to understand for an outsider can be isolated in this stage and assigned to specific places in the structure, without first needing to be fully explored or understood. This approach also forces the practitioners to be explicit about what really matters.

The approach also clarified what actually happens in the current business processes, as the examples regarding the concepts of process steps and service order have shown above. This conceptualisation of current practice allows for a very precise and fitting way of modelling and monitoring the business processes and leads to a better understanding by the practitioners of their own processes.

5.2 Administrations, Identities and References

One of the pillars of process logic is the concept of an administration. The definition of an administration given by Starreveld is: The systematic collection, recording, processing and supplying of information for purposes of the managing and functioning of a household and for purposes of the accountability thereof". When we combine this definition with the idea that an administration concerns one specific domain, it seems obvious to directly name the required administrations when process logic is specified. Here, it is important to note that administrations concern product data and not master data.

The first criterion to arrive at an administration is a high degree of homogeneity and autonomy. It must be possible to view the data that are collected in one administration as a single coherent whole. Also, the direct interactions with and dependencies on other administrations have to be as few as possible. A second criterion is the responsibility for its management; each administration in an organisation should have a single person who carries the responsibility for the correctness, timeliness en completeness of the data in it. This responsibility should ideally be located close to the primary process, to ensure that those responsible are in touch with the reality represented in the administration.

In the case at hand this mainly means that each branch has its own administration and carries full responsibility for it. For example, there is no central registration of orders and stocks, but each branch maintains its own administration in these areas. Incidentally, in this concrete case it does not mean that they are free to choose their own systems. Everyone uses the same system, but within it every branch has its own administration. Of course, in the presentation layer connections can be made across the different administrations to enable central monitoring of the processes. And the serial number administration in which the service history of individual devices is registered is an example of an administration that must necessarily be kept centrally because of the nature of the data and the interaction of these data with external systems.

For the development of the new information system the specified administrations are composed of two parts. First, there is a basic structure with entities with their internal and external identities. Of course, within the database a single entity has a single unique identity, but inside and outside of the organisation an entity might have many alternative identities. Think of the number of a service order for example: internal and external stakeholders can use all kinds of references for themselves and use their own reference to request or provide data. Another example of this mechanism is the serial number: at first viewing this is a unique number. In practice, this number is unique number within a specific brand, product group or model. Thus, a serial number does not uniquely specify a single device or part while it is required to do so. The enterprise also has a need at times to refer uniquely to a part that does not have a serial number, which can be met by assigning it a particular number generated for this purpose. When the part is gone, the number is as well. Based on these considerations, it is prudent to primarily assign unique numbers generated by the enterprise itself to parts and devices and to consider the serial number on a device or part as an alias to arrive at the generated number. This system is always applicable and avoids the complicated composite identification that results from accepting the serial number as identifier.

From the very beginning, the structure of the administrations has to be erected along with the associated references inside and outside of the enterprise. Further dressing up and setting up of the administration with data relevant to the contents, further status information, et cetera, can be done afterwards, in parallel to the development of the applications that use these data.

5.3 Lean IT

The lean approach places a number of demands on the set-up of an enterprise information system. Positively formulated, the information system must contribute to the effectiveness and efficiency of the business processes and use the most appropriate means to do so. Negatively formulated, the system is not allowed to cause waste (e.g.: excess production of information), to place undesired limits upon the business processes and it may promote the autonomy of processes as long as this does not harm overall efficiency and effectiveness. Information from the system has to be reliable and relevant.

Put otherwise: employees have to keep being presented with information in the right way and feed back information themselves and they must have the freedom to make their own decisions within their domain. Two examples illustrate the application of these criteria: First, the registration of direct hours on service orders. From the management there was a strong desire to gain a detailed view of the usage of hours in the primary process. In computer systems nothing is simpler than granting this wish: registration per service order, per department, per activity of time used. In everyday reality however, such a system leads to unusable information. First, because there is a mismatch with the way in which the work is actually done. Second, because it results in an excess of registrations. Either the categories are too general and the registrations limited, or the categories are specific and the registrations time-consuming. In both cases the registrations will provide an unreliable view of reality. That is why we opted to start with registration per service order in just the repair department, where it is registered for each employee when he begins and ends with a service order and which activities he performed during this time (which does not provide the time per activity). In this way, insight is provided into the ratio of time spend on service orders to time spend on other activities. Insight is also provided into the cases in which a service order has been handled repeatedly, by whom and to perform which activities. These data provide the foreman with a measuring stick to monitor the performance of his crew and to pay additional attention to activities that seem to take up too much time.

A second example is insight in what tasks must be done and which tasks might be done. The turnaround time of service orders is one of the most important parameters for the performance of the enterprise with regard to the various stakeholders. When norms are created for the turnaround time as a whole and for the turnaround time of each individual process step, it is possible for the system to directly show which service orders have to be handled on a particular day in a particular team and which other work remains to be done with what time remaining according to the norms. This allows the team to make optimal use of its capacity by handling the service orders that are the best fit at that moment for the current activities and available resources. Self-control instead of central control should result in a significant advance in efficiency here.

5.4 Protecting and Strengthening the Distinctiveness of the Company

The strength of the enterprise is two-fold: nationwide coverage and a strong bond to a strong brand with strong partners. The downside of this bond to market heavyweights is that they determine to a large extent what the service conditions will be, both regarding the fees and regarding the mutual information supply. In this sense we are dealing with strongly asymmetric relationships. On the other hand, AYS is able to relieve large market parties of work that those parties are much less well equipped to handle and to carry out this work to a high standard. In other words, the enterprise has a clear place in the market.

Current legacy information systems of AYS have been built or adapted based on concrete developments in and requests by the market partners. Because of the aging core system solutions have been added to applications that were not originally meant for such. The new system should improve the ability of the enterprise to react to market developments through improved insight into the actual course of the processes by sharply demarcating the various administrations. This improved information position should result in an improved bargaining position with both existing and new contract partners.

A second contribution to the strategic position of the enterprise is also based on pulling apart the core administrations. In this way the enterprise is enabled to develop activities other than just service orders for defective equipment. The potential of nationwide coverage with service vehicles can be utilised for other activities as well. The information architecture allows both developments to be introduced gradually to expand into new markets, without major, risky, investment.

Finally, the flexible legal and financial structure is a major advantage. The current diversity with both fully-owned branches and branches exploited by independent entrepreneurs allows for rapid change both in acquiring work and in subcontracting it. AYS can profile itself as a strong market party because it presents a unified face to the client (in corporate identity and in home visits) and orchestrates the orders, while the work may be carried out elsewhere. Separating the diversity of the legal and financial structures from the unity of AYS as a business actor is an important requirement for the information architecture.

6 CONCLUSIONS

At the time of writing this paper the analysis was not fully completed yet. The manner in which the project went and the structures that have been identified as process logic are already sufficient to consider the project successful for the client AYS. Here, it is relevant to note that AYS commissioned an analysis of the business processes by a potential supplier and industry specialist about a year before this case study, thus supplying AYS with a comparison. This earlier analysis did not lead to new insights into their processes since it was fully based on inquiring after factual details of the business processes of AYS to prepare for the implementation of the package. Terms were mostly left undefined and the information gap between supplier and customer was never bridged.

In contrast, the approach in this case study encompassed a critical analysis of the business processes and resulted in many new insights for AYS into its own practice. Furthermore, the structure described is suitable to accommodate future developments.

Thus, the method of analysis of the process logic has withstood the test of practice. However, the form for the final product, the description of the process logic, has not yet been found. The description as a list of definitions, a specification of the administrations and a specification of references certainly forms a useful and testable foundation, but a more formalised form of core entities and their transitions would be desirable.

To conclude, to make the transition between daily actions by employees in the business processes on the one hand and the process logic and information architecture on the other, a number of layers will have to be distinguished in the enterprise information system:

- A core system with administrations and operations
- A number of rulebooks (sets of business rules that determine the behaviour of the operations)
- Application software
- Setup of the application software
- Agreements and procedures regarding information flows and use of the computer systems

- Actual information flows within and outside of the pre-determined paths for doing so (computer systems, forms, structured consultations) within the daily business processes

The core system as a stable basic structure and the more changeable rulebooks on top of it should be specified rigidly in an abstract and formalised manner. The actual information flows are a constituent part of the organisation as a socially emergent phenomenon and ultimately it is impossible to fully capture their dynamics and wealth of detail. The elements in between are a mixture of social conventions and logical modelling in practice. IT people and bureaucrats (in its Weberian meaning) prefer to work here with formal sign system, while the practitioners who need to deal with nuance and to weigh heterogeneous norms against one another in practice are much more inclined to non-formal sign systems such as free text, oral explanations and face-to-face discussions with colleagues.

A major recurring problem in the development of an enterprise information system is the balancing of model with reality, of System with Lifeworld (Habermas, 1986). Acknowledging the formal nature of the core system and the surrounding rulebooks and acknowledging the dynamics, the capriciousness and the intangibility of the daily organisational reality can help in developing applications in a more efficient way. On the one hand, these applications are supported out of necessity by the stringent core system, on the other they themselves have to support daily practice. The conscious decision to limit computer systems to those operations for which they truly add value to operational practice is an essential step. Too often requirements and wishes are drafted from the perspective of managers located too far from daily practice and too often slogans such as ‘paperless office’ or ‘everything inside the system, no dependence on people’ are adhered to. In the end, it is about the criteria of the Lean approach: add value, avoid waste and always use the appropriate means to the task (the latter is a translation to practice of the avoidance of waste).

This case study had a tentative character, trying out several ideas regarding both the theoretical background and regarding the application of the ideas in a real world project. As such, it succeeded; in a short time good results were obtained (especially in comparison to a previous analysis of the same company carried out by a potential software supplier).

REFERENCES

De Geus, A. (1997). *The Living Company*. London: Nicholas Brealey

Dietz, J.L.G. (2006). *Enterprise Ontology*. Berlin: Springer.

Habermas, J. (1986). *The Theory of Communicative Action*. Cambridge: Polity Press.

Kay, J. (1993). *Foundations of Corporate Success*. New York: Oxford University Press.

Liu, K. (2000). *Semiotics in Information Systems Engineering*. Cambridge, Cambridge University Press.

Poli, R. (2010). Ontology: The Categorical Stance. In Poli, R. and Seibt, J. (Eds.), *Theory and Appications of Ontology – Philosophical Perspectives* (pp 1-22). Berlin: Springer.

Stamper, R. (2000). Information Systems as a Social Science: An Alternative to the FRISCO Formalism. In Falkenberg, E.D., Lyytinen, K. and Verrijn-Stuart, A.A. (Eds.), *Information System Concepts: An Integrated Discipline Emerging*. Dordrecht: Kluwer Academic Publishers.

Suurmond, C.P. (2012). Administrations as Instruments for Dealing with Organizational Complexity. In Shishkov, B. (Ed.), *Business Modelling and Software Design – First International Symposium BMSD 2011* (pp 130-146). Berlin: Springer.

Suurmond, C.P. (2013). Sign Systems, Information Systems, and Engineering. In Shishkov, B. (Ed.), *Business Modelling and Software Design – Second International Symposium BMSD 2012* (pp 82-101). Berlin: Springer.

Taylor, J.R. and Van Every, E. (2000). *The Emergent Organization*. Mahwah: Lawrence Erlbaum Associates.

Weber, M. (1968). *Economy and Society*. Berkeley: University of California Press.

Consistency, Complementalness, or Conflictation of Enterprise Ontology and Normalized Systems Business Process Guidelines

Philip Huysmans, Dieter Van Nuffel and Peter De Bruyn
Normalized Systems Institute, University of Antwerp, Prinsstraat 13, B-2000 Antwerp, Belgium
{*philip.huysmans,dieter.vannuffel,peter.debruyn*}*@ua.ac.be*

Keywords: Normalized Systems, Enterprise Ontology, Business Process Modeling, Enterprise Engineering.

Abstract: Both Enterprise Ontology and Normalized Systems can be considered as design theories, which provide prescriptive guidelines to design systems. Enterprise Ontology explicitly focuses on the design of organizations as being social systems. Originally, Normalized Systems focused on the design of evolvable software systems. However, it has been shown that, building on the Normalized Systems design knowledge, prescriptions for other domains, such as the business process domain, can be proposed as well. This domain seems to overlap at least partially with the domain of Enterprise Ontology, which is used to establish claims concerning process design in various publications. However, both theories are based on completely different kernel theories. Therefore, this paper analyzes to which extent the guidelines proposed for the Normalized Systems Business Processes are consistent, complementing or conflicting with prescriptions from Enterprise Ontology. A consistent set of prescriptions could lead to a more integrated approach for designing integrated organizations, business processes and software systems.

1 INTRODUCTION

The design of organizations and their components such as the organizational structure, business processes, and software systems is an important topic in both practical and scientific communities (Galbraith, 1974; Avenier, 2010). Notwithstanding the attention for this subject, explicit design knowledge in these fields seems limited. For example, Mendling et al. argue that many theoretical frameworks are too abstract, and that more practically-oriented guidelines lack empirical and theoretical support (Mendling et al., 2010). As a result, design of organizational components is often considered as craftsmanship, rather than engineering.

The enterprise engineering paradigm introduces a set of prescriptive design theories which seek to remedy this issue (Dietz et al., 2013). It specifically mentions the β and ν theories as well-founded theories to guide design efforts. The ν-theory states that the design of a system is normalized when a change consists of a set of elementary changes, so that every elementary change does not trigger combinatorial effects (Dietz et al., 2013, p. 101). Normalized Systems (NS) provides concrete guidelines and design patterns to obtain such normalization in software systems (Mannaert and Verelst, 2009). Based on this approach, normalization of business processes has been researched as well (Van Nuffel, 2011). This research resulted in a set of guidelines which need to be adhered to during business process design. While both Normalized Systems and Normalized Systems Business Processes (NSBP) originally aimed at obtaining designs exhibiting stability as defined in systems theory (Kelly, 2006), it has been argued that the resulting guidelines are in line with existing heuristics of experienced designers. For example, the guidelines presented by Van Nuffel are related to the existing business process literature (Van Nuffel, 2011). Nevertheless, the main contribution of both approaches is the formulation of unambiguous and theoretically founded guidelines based on the single postulate of obtaining the systems theoretic concept of stability. As a result, an approach which resembles traditional engineering, rather than mere craftsmanship, arises on these levels.

The β-theory states that enterprise architecture should be defined as deliberate restriction of design freedom, which should address the function design, construction design, and implementation design of systems (Dietz et al., 2013, p. 100). For example, Enterprise Ontology (EO) prescribes how the construction design of an organization should be made (Dietz, 2006). EO prescribes a clear way of separating different abstraction levels to be considered in organizations (i.e., ontological, datalogical and infological) and a systematic recurring pattern to model the ontological level.

While the formulation of such theories has been demonstrated to further the field in practice[1], several issues remain. One important issue is the current lack of integration between the specific methods which integrate the different theories (Dietz et al., 2013). This issue has been documented in many studies related to enterprise architecture (Kaisler et al., 2005; Dreyfus, 2007). While certain frameworks, such as TOGAF, focus explicitly on a method to integrate high-level activities such as strategy formulation with detailed software design, no prescriptive methods are present in such frameworks. As a result, the integration of different prescriptive methods can remain very complex. Enterprise architecture researchers have shown that most reports focus on a single architectural layer, and do not address this integration (Schenherr, 2008). In practical projects, this results in local optimizations, which restrict the success of an organizational design as a whole (Kaisler et al., 2005; Dreyfus, 2007).

This means that, within the enterprise engineering community, additional research is required which works towards an integrated method consisting of different prescriptive design theories. For example, both EO and NSBP seem to provide a similar kind of guidelines when used in practical projects, which could indicate that both approaches could be used as complements in various projects. However, a clear obstacle when aiming to apply both approaches simultaneously, is their difference in theoretical backgrounds and abstraction. Therefore, an in-depth analysis regarding the possible compatibility of the guidelines resulting from both approaches is required upfront. Such approach would investigate the extent to which these guidelines are (1) similar in both approaches (i.e., *consistent*), (2) providing additional guidelines (i.e., *complementary*), or contradicting one another (i.e., *conflicting*). It should be noted that this approach does not result in a *theoretical* analysis of EO and NS(BP). On the one hand, NSBP cannot be theoretically EO-compliant, since the distinction axiom is not adhered to: no separation of ontological, infological and datalogical concerns is made. On the other hand, EO has not been developed based on the concept of systems theoretic stability. Notwithstanding this reservation, we are convinced that an analysis of the consistency, complementalness, or conflictation of the practical guidelines of both approaches can contribute to an integrated use of both EO and NSBP in various projects.

[1] See for example `www.demo.nl` for case studies

2 BACKGROUND

2.1 Normalized Systems

Normalized Systems theory is aimed at studying how modular structures behave under change (Mannaert and Verelst, 2009). Initially, the theory was developed by studying change and evolvability at the software architecture level, by applying concepts such as stability and entropy to the study of the modular structure of the software architecture. Considering the application of systems theoretic stability to software architecture, stability implies that a bounded input function should result in bounded output values, even as $T \to \infty$. In other words, stability demands that the impact of a change is only dependent on the nature of the change itself. If the amount of impacts is related to the size of the system, a *combinatorial effect* occurs. Research has shown that it is very difficult to prevent CE when designing software architectures. More specifically, it has been proven that CE are introduced each time one of four theorems is violated (i.e., separation of concerns, data version transparency, action version transparency and separation of states).

Various studies have shown that combinatorial effects do not occur solely on the level of software architectures (Van Nuffel, 2011; Huysmans, 2011). On the business process level, it has been argued that business processes at their most basic level (i.e., the "elementary tasks and elementary sequencing and design of these tasks" (Van Nuffel, 2011)) can be considered to be modular structures as well. In this context, business processes have been compared to production lines (Van Nuffel et al., 2009a). In this analogy, a business process flow performs operations on instances of a specific *life cycle information object*. Although production lines may seem highly integrated, they are actually loosely coupling. Every single processing step requires the completion of the previous steps on that instance of a particular product, but it does not require any knowledge of the previous processing steps, nor of the subsequent steps. As a result, changes to individual processes or tasks do not impact other processes of tasks (Van Nuffel, 2011). Put differently, no combinatorial effects occur. More generally, a business process which does not contain combinatorial effects is called a Normalized Systems Business Process (NSBP). In order to achieve such processes, a set of guidelines has been developed, which are based on the more fundamental theorems of Normalized Systems. Together, these guidelines allow the design of business processes without introducing combinatorial effects.

2.2 Enterprise Ontology

Enterprise Ontology (EO) provides an organizational theory (Dietz, 2006) which is based on the Language-Action Perspective (LAP). Consequently, it considers an organization as a social system, and focuses on actor roles as the essential components of organizations. This is important for the goal of this paper, since this background results in the claim that EO provides "a modular framework for business processes" (Dietz, 2003b, p. 1). The EO theory consists of four axioms (i.e., the operation axiom, the transaction axiom, the composition axiom and the distinction axiom) (Dietz, 2006). These axioms allow to specify in more detail what is meant with the "modular construction of business processes" (Dietz, 2003b, p. 18). Business processes are considered to consist of three levels of building blocks. A first type of building block (the atoms) refers to the individual acts performed by actors, as explained by the operation axiom. These atoms can be combined in higher-level building blocks (i.e., molecules), which represent the transactions as explained in the transaction axiom. Multiple transactions can be required to fulfill a certain service to a stakeholder. The collection of these transactions (i.e., a fiber) is then considered to be a business process.

Rather than merely defining business processes using EO concepts, various studies have focused on the design of business processes. For example, the main research question of the paper *Basic Notions Regarding Business Processes and Supporting Information Systems* is "how business processes can be understood in such a way that their continuous and concurrent (re)designing and (re)engineering can be performed more effectively than what is currently the case" (Dietz and Albani, 2005). Another example is the paper *Enhancing the Formal Foundations of BPMN by Enterprise Ontology*, which states 11 propositions which can be derived from EO axioms (Van Nuffel et al., 2009b). Based on the axioms, additional prescriptions for designing business processes are available. For example, the operational cycle (Dietz, 2006, p. 163) states that an actor role needs to be added when a transaction cannot be performed in the same cycle of other transactions. Put differently, this implies that the executor actor of an enclosing transaction needs to be the initiator actor of an enclosed transaction (cf. the composition axiom). Consequently, EO prescribes that certain end-to-end processes which are often defined in practice (e.g., order-to-cash processes) need to be separated.

Various claims have been made that EO can indeed lead to better results when (re)designing processes. The abstractions discussed in the distinction axiom are claimed to be "a tremendous advantage for discussing business process optimization" (Dietz, 2006, p. 183), (van Reijswoud, 1999). Moreover, the dedicated model within the DEMO methodology to represent business processes (i.e., the process model) has been claimed to "facilitate the discussion about the redesign of business processes" (Dietz, 2006, p. 183).

2.3 Is it possible to Compare Both Theories?

Caution should be applied when comparing the Enterprise Ontology and Normalized Systems theory, since their intentional application domains vary greatly. Normalized Systems theory focuses on evolvability of software architectures, while Enterprise Ontology attempts to describe coordination in organizations. Nevertheless, the Design Science paradigm argues that the application of theories of related fields is useful to make scientific progress. Moreover, Winter and Albani claim that different design theories can be combined in certain projects (Winter and Albani, 2013). Both the Normalized Systems and Enterprise Ontology theory have already been positioned in a Design Science research framework (Huysmans et al., 2012; Winter and Albani, 2013). Comparing these frameworks indicates an important difference between both theories: Enterprise Ontology builds on communication theories (i.e., the theory of communicative action, the language-action theory and systemic ontology) while Normalized Systems builds on system theoretic and thermodynamic concepts such as stability and entropy.

Notwithstanding this clear difference in kernel theories, remarkable similarities between Normalized Systems and Enterprise Ontology have been discussed as well (Huysmans, 2011). For example, consider the Separation of States theorem. It states that "the calling of an action entity by another action entity needs to exhibit state keeping in normalized systems" (Mannaert and Verelst, 2009). It therefore prescribes how action elements can interact. This impacts, for example, the workflow element, which aggregates action elements. A workflow can reach different states by performing state transitions. A state transition is realized by an action element. The successful completion of that action element results in a defined life cycle state. The workflow specification determines which state transitions can be made. Similarly, the state of a transaction in enterprise ontology is determined by the successful performance of acts. The result of such an act results in the creation of a defined fact. Despite the different terminology, a clear resemblance between Normalized Systems and Enterprise Ontology emerges: state-keeping is enforced in Normalized Systems by defining states, and in Enterprise Ontology by creating facts. These Normalized Systems states are the result of executing actions, whereas the Enterprise Ontology facts are the result of executing acts. Which actions can be performed is determined by the state transitions in Normalized Systems, and occurrence laws in Enterprise Ontology.

Moreover, other attempts have been made to integrate Normalized Systems and Enterprise Ontology theory more directly (Huysmans et al., 2010; Krouwel and Op't Land, 2011; Op 't Land et al., 2011). It should be noted that in these efforts, an inductive approach based on concrete artifacts is used, which can be contrasted to a more theoretical approach. Similarly, this paper does not attempts to provide a theoretical comparison, but aims to compare *similar components* of both theories on an *overlapping domain*. The similar components refer to the formulation of *prescriptive guidelines* by both theories. This is important, given the different kinds of theories available in literature (e.g., descriptive theories, explanatory theories, or design theories). In Normalized Systems, such guidelines are referred to by stressing the determinism of design (Van Nuffel, 2011). In Enterprise Ontology, we find clear references to the importance of such guidelines in the definition of architecture, which is "the normative restriction of design freedom" (Dietz, 2006). The *overlapping domain* is the domain of business processes, which is clearly addressed in Normalized Systems Business Processes (cf. Section 2.1. While business processes are defined within EO as well, it should be noted that we interpret the prescriptions of EO not only on the ontological level. In any organization, the ontological models eventually need to be extended to include the infological and datalogical layers, and to specify an implementation. Implementation means "the particular subjects that fulfill the actor roles at a particular time, the particular way in which C-acts are performed, and the particular way in which P-acts are performed." Several publications focused on this subject, which show that a design is obtained which is influenced by EO prescriptions, but which can no longer be considered to be a design of a social system by itself, or be entirely on the ontological level. For example, we mention research to define use cases for information systems based on DEMO models (Dietz, 2003a). This is in line with insights from the generic systems development process (GSDP) (Dietz, 2006, p. 71), which states that a functional specification of an object system needs to be made based on the constructional model of the using system.

3 APPROACH

In order to compare the guidelines of EO and NSBP, four categories should be considered: (1) *Consistent*: guidelines from NSBP and EO prescribe the same design; (2) *EO-ignorant*: an NSBP guideline which has no similar EO guideline; (3) *NSBP-ignorant*: an EO guideline which has no similar NSBP guideline; (4) *Conflicting*: a NSBP guideline, which prescribes a different design than an EO guideline, or vice versa. Certain guidelines are expected to be consistent, since both EO and NSBP consider business processes as modular structures, and propose guidelines to optimize their design. However, given the different kernel theories of both approaches, and their non-identical goals, certain conflicting guidelines could be identified. Moreover, neither EO or NSBP claim to be complete. The claim from Dietz that "we do not intend to claim that ... even the whole ψ-theory is a sufficient basis for achieving optimally performing enterprises" (Dietz, 2006, p. 81) indicates the validity of the EO-ignorant category. The claim from Van Nuffel that NSBP guidelines are necessary, but not sufficient, indicates the validity of the NSBP-ignorant category. We will adopt the work of Van Nuffel as our starting point as it explicitly lists a set of 25 guidelines, whereas the guidelines from EO have not been formally consolidated in such list exhaustively enumerating all guidelines incorporated in the method. Further, given this starting point to determine for each guideline to which category it belongs, the NSBP-ignorant category will not be required in this paper.

The authors of this paper independently made a classification of the NSBP guidelines. After integrating the result, differences were discussed, and the assessment was iteratively refined. All three authors have a sufficient background in both EO and NSBP. The NSBP PhD dissertation (Van Nuffel, 2011) and EO book (Dietz, 2006) were used as reference materials. Several academic publications were used for additional details. Moreover, several cases (see e.g., (Van Nuffel, 2011), (Dietz, 2006), `http://www.demo.nl`) were consulted as an application of the guidelines.

4 COMPARISON

Within this section, the actual comparison between the practical guidelines resulting from the two theoretical approaches is made. Our discussion will follow the division made within the PhD of Van Nuffel (Van Nuffel, 2011): first, the general guidelines with respect to identifying business processes are discussed. Second, the comparison continues with the three additional guidelines that in specific cases identify business processes. Third, the comparison continues with the guidelines determining individual tasks, and finally, the auxiliary guidelines are investigated. The business process patterns discussed in the PhD of Van Nuffel (Van Nuffel, 2011) focus on issues not discussed by Enterprise Ontology, and are therefore not taken into account. This section lists the names of the guidelines in italic and bold font. Next, the guideline is summarized in italic. Then, the consistency, complementalness or conflict with EO is discussed. An overview of these discussions is provided in Table 1.

4.1 General Business Process Guidelines

1.1 Elementary Business Process. *A Business process denotes a constrained sequence — i.e., sequence, iteration or selection — of individual tasks representing state transitions in the life cycle of a single life cycle information object.* Within Enterprise Ontology (EO), a P-fact is a factum, which is defined as *"the result or the effect of an act"* (Dietz, 2006, p. 42). Therefore, facta "can be conceived as status changes of . . . an object in some class" (Dietz, 2006, p. 42). Furthermore, the order in which facta occur is determined by so-called occurrence laws (Dietz, 2006, p. 43). The transaction is thus about a unique P-fact transcending the transaction pattern, which can be considered to be somewhat consistent with a NS business process which is about state transitions of a single life cycle information object as stated by NSBP. The one-to-one relationship between a transaction and a P-fact is in our opinion conceptually consistent with the one-to-one relationship between a single life cycle information object and a business process.

1.2 Elementary Life Cycle Information Object. *an information object not exhibiting state transparency is a life cycle information object.* Whereas NSBP prescribes the criterion of state transparency (i.e., when no proper state transitions should be made explicit (Van Nuffel, 2011, p. 118)) to define whether an information object is a genuine life cycle information object processed in a business process, Enterprise Ontology does not explicitly state a rule, criterion or law that in all circumstances denotes what a single P-fact is. There are evidently ways and requirements a P-fact should adhere to, but no general identification mechanism seems to be made explicit:

- "We conceive the result of a production act as a particular change in the state of the system's object world" (Dietz, 2006, p. 58);
- "The object world reflects the produced things (e.g., goods or services) that are delivered to the elements in the environment" (Dietz, 2006, p. 58).

As a consequence, – although it could be argued that only most fine-grained production facts exist (and therefore, that production facts are defined unambiguously), but that they can be aggregated to simplify models – it seems that identification of production acts in EO is not unambiguous: it depends on what is considered to be the system and environment, and different production facts can be identified depending on the aggregation level taken into account. Moreover, elementary life cycle objects can also refer to infological and datalogical production facts. Therefore, the authors categorize this guideline as EO-ignorant.

1.3 Aggregated Business Process. *In order to represent an aggregated business process, an aggregated life cycle information object has to be introduced.* In EO, a business process is based on the composition axiom: "a business process is a collection of causally related transaction types, such that the starting step is either a request performed by an actor role in the environment or a request by an internal actor role to itself" (Dietz, 2006, p. 103). Based on this definition, the operational cycle (Dietz, 2006, p. 163) can be understood, which specifies that certain end-to-end processes cannot be considered as causally related transactions. Since the NSBP guideline is explicitly aimed towards representing *any* required end-to-end process, both theories are conflicting in most situations.

1.4 Aggregation Level. *Tasks performed on a different aggregation level denote a separate business process.* Although in the PSD-diagrams the causal and conditional links are enriched with cardinalities that describe the relationship between different transactions, nowhere is indicated that when an analyst discovers an one-to-many relationship between two candidate transactions, both should be separated. Furthermore, this latter relates to the aggregation level on which production facts are defined, since a production fact defines a transaction. Again, this does not result in a guideline to actually separate the transactions. Therefore, EO seems to be ignorant with respect to this design issue.

1.5 Value Chain Phase. *The follow-up of an organizational artifact resulting from a value chain phase denotes a different business process.* While some arguments can be made for the consistency of this guideline, the most important argument seems to indicate a conflict. For example, the operation axiom might indicate value chain phases as separate transactions, although it is dependent on the aggregation level on which the P-facts are defined. Moreover, the composition axiom illustrates the possible nesting required to integrate the different phases. However, the transaction axiom results in design decisions like explicitly stating that the Order phase belongs to the Delivery process in a typical Customer Order process scope. With respect to the latter, NSBP clearly state that these phases should be separated as they denote separate concerns (Van Nuffel, 2011, p.132-34). In this way, NSBP seems to consider concerns a level "deeper" as it explicitly considers a delivery not to belong to the Order Phase, but as a separate process in the aggregated business process Customer Order. Therefore, both theories seems to disagree with respect to this design issue.

1.6 Attribute Update Request. *A task sequence to update an attribute of a particular life cycle information object that is not part of its business process scenarios, is represented by an Attribute Update Request business process.* The guideline prescribes to separate state transitions dealing with modifying an attribute of a life cycle information object that does not belong to the business process scenarios (i.e., included in the process). Enterprise Ontology however, considers such requests to change to be part of the transaction. Mostly, they can be represented by one of the four cancelation patterns. As such, this represents a conflict between the two theories, although they comply with each other on modifications that do belong to the business process scenarios.

1.7 Actor Business Process Responsibility. *Actor business process responsibility indicates a separate business process if different actors are responsible for a different set of tasks, of which the task allocation belongs to different process owners.* The operation axiom declares actor roles to denote chunks of authority, responsibility and competence. Furthermore, following Enterprise Ontology, a single transaction can only be executed by a single actor role. In this way, this notion is equivalent to stating that state transitions of a particular life cycle information object being part of the responsibility of a particular process owner denote a separate business process. Furthermore, in addition to EO, also NSBP identifies the only vaguely described notion of process ownership within literature. As a consequence, NSBP opts for a clear identification of such process ownership, which seems to be very closely related to EO's notion of authority.

1.8 Notifying Stakeholders. *Because notifying, or communicating a message to, stakeholders constitutes an often recurring functionality in business processes, a designated business process will perform the required notification.* EO considers notifying stakeholders as performing coordination acts, which are part of an ontological transaction that creates a single P-fact. However, NSBP identifies the concern of notifying stakeholders to clearly differ from the concerns taken care of by other business process (e.g., delivering an order, recruiting an employee, etc.): "delivering a message in the correct format to the intended recipients at the right time in an unchanged format, with the related fault handling" (Van Nuffel, 2011, p.143). These concerns refer to implementation details, which are not considered on the ontological level. Therefore, EO theory is ignorant with respect to this design guideline.

1.9 Payment. *Because paying a particular amount of money to a particular beneficiary constitutes an often recurring (technical) functionality in business processes, a designated (technical) business process will perform the required payment.* The payment business process/transaction is identified by both theories, and can be considered as consistent. Various DEMO cases illustrate this. It should be noted that NSBP requires that at least the execution phase of payment processes is implemented using a reusable business process, in order to prevent combinatorial effects. This is not clear from the DEMO cases, which explicitly define multiple payment transactions.

4.2 Business Process Guidelines

2.1 Product Type. *A different type of product or service denotes a main concern, and thus indicates a different business process.* The composition axiom seems to indicate that Enterprise Ontology also recognizes the existence of transactions that although being enclosed by the same transaction, do constitute individual and independent transactions based on a product structure. But again, no clear rules could be identified, implicitly stated by "one could apply a finer-grained product structure" (Dietz, 2006, p. 170). The notion of a product type defined by Van Nuffel (Van Nuffel, 2011, p. 149) allows some interpretation as well, namely the domain expert who will identify the characteristic dimensions on which product types exhibit similar properties. As a consequence, we categorize this design issue as an EO-ignorant one.

However, if the Logistics example discussed by NSBP is taken into account, the design issue also seems to indicate conflicting statements by the two theories. The NSBP separate the Logistics processes based on the following types: non-food, food, quickly rotating, slowly rotating, and so on. On the other hand, EO theory seems to declare that these product types do not cause another type of P-fact to be created, and thus no separate transaction to be executed. This could indicate a potential conflict.

2.2 Stakeholder Type. *Stakeholder type should principally be considered a cross-functional concern (i.e., a concern which does not require a life cycle information object by itself), expect for those business processes where the stakeholder type denotes the life cycle information object.* Whether the theories comply, comes down to the question: does EO consider a transaction to be independent from the actor role for which it is potentially performed? In the PhD of Van Nuffel, a case about Human Resources (HR) processes is discussed in which it is clearly demonstrated that the assignment processes for a statutory employee and a non-statutory employee differ. Based on the authors' knowledge, EO does not provide any rule or prescription about the potentially different nature of a transaction. For example, in the Educational Administration case, no separate transactions are created based on different student types.

2.3 Access Channel. *The concept of an access channel indicates a cross-functional concern.* In EO publications no explicit referral to this design question could be found. However, implementation is explicitly out of scope for EO: EO "fully abstracts from the implementation [of C-acts]", which includes "the particular way in which C-acts are performed" (Dietz, 2006, p. 83). Consequently, it can be argued that the theories comply as EO does not explicitly states a different access channel denotes a separate transaction. Consider in this context the pizzeria case (Dietz, 2006, p. 166). The transaction *T01: Completion* contains all access channels to place an order.

4.3 Task Guidelines

3.1 A Single Functional Task - Overview. *A task represents a functional entity of work that either results in a single state transition of a single information object type, or refers to an Update or Read task on a single information object type.* Where NSBP specifically describes what a single task (or step within a business process) can be, our analysis of the EO fails to find equivalent rules. Of course, the transaction axiom identifies single acts (resulting in facts) within the transaction which might indicate consistency. However, the authors seem to find more evidence to categorize it as EO-ignorant. For instance, consider the acceptance of a stated P-fact consisting of an evaluation of its quality by performing three quality tests and then communicating the outcome to the initiator actor role which is authorized, responsible and competent to accept the P-fact, who will communicate it to the executor. EO considers this example to be part of the Accept C-act whereas NSBP prescribes to separate it in five different tasks, and two instances of the Notification business process. Thus, based on our analysis, we consider it to be EO-ignorant.

3.2 CRUD Task. *Each of the Create - Read - Update - Delete (CRUD) operations constitutes a single task.* Since these tasks are on the infological and datalogical layers, this guidelines is EO-ignorant.

3.3 Manual Task. *Every manual task of which the initiation and completion has to be known, has to be designed as a separate task.* EO makes abstraction of the implementation of C- and P-acts (also see discussion of 2.3 Access Channel). Therefore, EO is ignorant with respect to this guideline.

3.4 Managing Time Constraint. *The management of a time constraint denotes a separate task because it represents the individual concern of managing a particular time constraint.* In EO, a time aspect only seems present in the time-aspect of the proposition of a P-fact (Dietz, 2006, p. 84) and self-initiating transactions (Dietz, 2006, p. 99). However, EO makes no claims whatsoever with respect to (not) separating an individual time constraint. As such, we categorize the guideline to be EO-ignorant.

3.5 Business Rule Task. *A single business rule should be separated as a single task.* An individual business rule should be isolated in its designated task following NSBP. EO acknowledges that business rules can sometimes be existential laws, as expressed in the state model, or action rules, which are expressed in the action model (Dietz, 2006, p. 196). In this sense, both seem to be consistent. However, EO does not explicitly states that every single business rule should be isolated. Therefore, EO seems to be rather ignorant to this design issue.

3.6 Bridge Task. *When a business process instance operating on an instance of life cycle information object type I has to create a business process instance of another life cycle information object type L, this functionality is designed as a bridge task that initiates the creation of the instance of the life cycle information object L, and represents a state transition on the instance of I.* As already illustrated above, the composition axiom of EO denotes the nesting of transactions. As such, it is illustrated that the Request C-act can be “triggered” by another transaction (i.e., the executor of an enclosing transaction can initiate an enclosed transaction). The Result structure analysis step of the DEMO methodology also adds to this. Conceptually, this is what a bridge task represents: it triggers the execution of another business process.

3.7 Synchronization Task. *When a business process instance operating on a life cycle information object I has to inform a business process instance of another life cycle information object L, a synchronization task, representing a state transition on the instance of I, alters the state of the business process instance of L.* The NSBP synchronization task conceptually equals the waiting conditions specified in the EO model based on the Result structure analysis, following the composition axiom.

3.8 Synchronizing Task. *A synchronizing task represents the task receiving information from another business process’s execution, in order to continue the business process control flow.* Equivalent to the Bridge task, also the Accept C-act in the EO transaction pattern represents conceptually the same as a synchronizing task. It allows the enclosing transaction/business process to continue, and thus is the end of the waiting condition.

3.9 Actor Task Responsibility. *A task cannot consist of parts that are performed by different actors.* Here NSBP is consistent with EO, as the operation axiom states that actor roles are elementary chunks of authority, responsibility and competence. Thus the fact that another actor role is authorized, responsible and competent to perform a particular task, suffices to split this task from any other task another actor role is authorized, responsible, and competent to execute.

4.4 Auxiliary Guidelines

4.1 Unique State Labeling. *Each state of a life cycle information object has to be unique.* The first auxiliary guideline, *Unique State Labeling*, states that each state of a life cycle information object should be unique. Thus, it indicates the necessity to uniquely define the states a business process can transverse. Also EO identifies unique labels as each coordination act and each transaction are uniquely labeled; and even more it states that facts can be created, but cannot be undone (Dietz, 2006, p. 82). Thus theories are considered to be consistent.

4.2 Unique State Property. *A life cycle information object instance can only be in a single state at any time.* Also EO declares a transaction has a unique status: the last performed fact, which is defined in EO as a state transition in the C- or P-world (Dietz, 2006, p. 82). Thus theories are considered to be consistent.

4.3 Explicit Business Process End Point. *If a business process type has multiple possible outcomes, each of these scenarios should have its dedicated end point reflecting the respective end state of a business process instance.* EO specifies through its transaction patterns (basic-standard-cancelation) that every scenario should be explicitly described. In this way, it is consistent with NSBP as every business process’ execution results in a specific end point/state, and not in a general state “finished”.

4.4 Single Routing Logic. *A split/join element in a business process’s control flow should only represent a single split or join routing expression.* Essentially EO does not discuss this proposed guideline, so it is considered to be EO-ignorant. However, it can be argued that both theories are consistent because within the transitions between the different C-facts and P-fact that are exhaustively described in the transaction pattern, no violation to the NSBP guideline was identified. Further research should identify whether this non-violation is purposefully – and thus the theories are consistent – or rather by chance – and thus remains EO-ignorant.

5 DISCUSSION

Table 1 summarizes the comparison made in the previous section. A bullet denotes that the identified category is determined without any doubt. An open circle means the categorization still needs further elicitation as a unique categorization could not be identified.

When scanning the table, it can be argued that the theories comply on many points (i.e., at least 10 out of 25 guidelines are consistent), indicating that a surprising overlap exists between guidelines prescribed by EO and NSBP, given their different the-

Table 1: Consistency of NSBP guidelines and EO.

	Consistent	EO-ignorant	Conflict
1.1 Elementary Business Process	●		
1.2 Elementary Life Cycle Information Object		●	
1.3 Aggregated Business Process			●
1.4 Aggregation Level		●	
1.5 Value Chain Phase			●
1.6 Attribute Update Request	○		○
1.7 Actor Business Process Responsibility	●		
1.8 Notifying Stakeholders		●	
1.9 Payment	●		
2.1 Product Type		○	○
2.2 Stakeholder Type		●	
2.3 Access Channel	○	○	
3.1 A Single Functional Task - Overview		●	
3.2 CRUD Task		●	
3.3 Manual Task		●	
3.4 Managing Time Constraint		●	
3.5 Business Rule Task		●	
3.6 Bridge Task	●		
3.7 Synchronization Task	●		
3.8 Synchronizing Task	●		
3.9 Actor Task Responsibility	●		
4.1 Unique State Labeling	●		
4.2 Unique State Property	●		
4.3 Explicit Business Process End Point	●		
4.4 Single Routing Logic	○	○	

oretical backgrounds. The EO-ignorant category is mostly discovered in the NSBP task rules. Almost all observations can be contributed due to the different abstraction level (EO does not consider these design questions), or the lack of a clear available answer in the different publications (e.g., Stakeholder Type). Consequently, NSBP seems to answer design questions EO does not answer or does not consider. Regarding the conflicting guidelines, some genuine contradictions (e.g., Aggregated Business Process) were identified. These conflicts should be clarified in future research, especially because most conflicts occur in the core (i.e., the first twelve) NSBP guidelines.

Moreover, more in-depth analysis is required, since the reason for consistent design decisions may differ. For example, consider the library case. The transaction *T03: Reduced fee approval* is a separate transaction in EO because it is executed by a different actor (Dietz, 2006, p. 144). In NSBP, it is a different transaction because "it denotes a separate concern, because it recurs in at least two situations" (Van Nuffel, 2011, p. 217) (i.e., when creating a new member and when collecting the yearly fee). Additionally, the NSBP-ignorant category needs to be elaborated upon as well. For example, various coordination acts are not required to be modeled in NSBP, for example when they are implicit. The explicitation of this category could especially aid the completeness of NSBP models.

Nevertheless, the authors hypothesize that — given the consistency between both theories and under the condition that the different abstraction levels on which they clearly operate do outweigh the contradictions, or that contradictions could be resolved by clearly identifying the abstraction levels on which both theories have their proven scientific importance — a method combining both theories to analyze businesses can be proposed. We will further elaborate on this method and its applications in future research.

6 CONCLUSIONS

In this paper, we explored to which extend the prescriptive guidelines related to the business process domain of EO and NSBP are consistent, complementary, or conflicting. We explained how both approaches offer theory-based guidelines to design business processes, and discussed in detail the assessment of the various NSBP guidelines. Moreover, we suggested several possibilities for further research, to work towards an integrated method for organizational design.

ACKNOWLEDGMENTS

P.D.B. is supported by a Research Grant of the Agency for Innovation by Science and Technology in Flanders (IWT).

REFERENCES

Avenier, M.-J. (2010). Shaping a constructivist view of organizational design science. *Organization Studies*, 31(9-10):1229–1255.

Dietz, J. (2003a). Deriving use cases from business process models. In Song, I.-Y., Liddle, S., Ling, T.-W., and Scheuermann, P., editors, *Conceptual Modeling - ER 2003*, volume 2813 of *LNCS*, pages 131–143. Springer Berlin Heidelberg.

Dietz, J. L. G. (2003b). The atoms, molecules and fibers of organizations. *Data Knowl. Eng.*, 47:301–325.

Dietz, J. L. G. (2006). *Enterprise Ontology: Theory and Methodology*. Springer.

Dietz, J. L. G. and Albani, A. (2005). Basic notions regarding business processes and supporting information systems. *Requir. Eng.*, 10:175–183.

Dietz, J. L. G., Hoogervorst, J., Albani, A., Aveiro, D., Babkin, E., Barjis, J., Caetano, A., Huysmans, P., Iijima, J., Van Kervel, S., Mulder, H., Op t Land, M., Proper, H. A., Sanz, J., Terlouw, L., Tribolet, J., Verelst, J., and Winter, R. (2013). The discipline of enterprise engineering. *International Journal of Organisational Design and Engineering*, 3(1):86–114.

Dreyfus, D. (2007). Information system architecture: Toward a distributed cognition perspective. In *ICIS 2007 Proceedings*. Paper 131.

Galbraith, J. R. (1974). Organization design: An information processing view. *Interfaces*, 4(3):28–36.

Huysmans, P. (2011). *On the Feasibility of Normalized Enterprises: Applying Normalized Systems Theory to the High-Level Design of Enterprises*. PhD thesis, University of Antwerp.

Huysmans, P., Bellens, D., van Nuffel, D., and Ven, K. (2010). Aligning the constructs of enterprise ontology and normalized systems. *Lecture notes in business information processing*, 49(1):1–15.

Huysmans, P., Oorts, G., and De Bruyn, P. (2012). Positioning the normalized systems theory in a design theory framework. In Shishkov, B., editor, *Proceedings of the Second International Symposium on Business Modeling and Sofware Design (BMSD2012)*, pages 33–42, Geneva, Switzerland.

Kaisler, S. H., Armour, F., and Valivullah, M. (2005). Enterprise architecting: Critical problems. In *Proceedings of the 38th Hawaii International Conference on System Sciences*, volume 8, page 224b, Los Alamitos, CA, USA. IEEE Computer Society.

Kelly, D. (2006). A study of design characteristics in evolving software using stability as a criterion. *Software Engineering, IEEE Transactions on*, 32(5):315–329.

Krouwel, M. and Op't Land, M. (2011). Combining demo and normalized systems for developing agile enterprise information systems. In Albani, A., Dietz, J., and Verelst, J., editors, *Proceedings of EEWC 2011*, pages 31–45. Springer-Verlag.

Mannaert, H. and Verelst, J. (2009). *Normalized Systems—Re-creating Information Technology Based on Laws for Software Evolvability*. Koppa, Kermt, Belgium.

Mendling, J., Reijers, H. A., and van der Aalst, W. M. P. (2010). Seven process modeling guidelines (7pmg). *Inf. Softw. Technol.*, 52(2):127–136.

Op 't Land, M., Krouwel, M. R., van Dipten, E., and Verelst, J. (2011). Exploring normalized systems potential for dutch mod's agility. In Harmsen, F., Grahlmann, K., and Proper, E., editors, *Proceedings of PRET 2011*, pages 110–121. Springer-Verlag.

Schenherr, M. (2008). Towards a common terminology in the discipline of enterprise architecture. In *ICSOC Workshops'08*, pages 400–413.

Van Nuffel, D. (2011). *Towards Designing Modular and Evolvable Business Processes*. PhD thesis, University of Antwerp.

Van Nuffel, D., Mannaert, H., De Backer, C., and Verelst, J. (2009a). Deriving normalized systems elements from business process models. In Boness, K., editor, *Proceedings of ICSEA 2009*, pages 27–32. IEEE Computer Society.

Van Nuffel, D., Mulder, H., and Van Kervel, S. (2009b). Enhancing the formal foundations of bpmn by enterprise ontology. In *CIAO! / EOMAS*, pages 115–129.

van Reijswoud, V. (1999). Model based business system transformation. In Pries-Heje, J., Ciborra, C., Kautz, K., Valor, J., Christiaanse, E., and Avison, D., editors, *Proceedings of ECIS 1999*, pages 600–615, Copenhagen.

Winter, R. and Albani, A. (2013). Restructuring the design science research knowledge base. In Baskerville, R., De Marco, M., and Spagnoletti, P., editors, *Designing Organizational Systems*, volume 1 of *Lecture Notes in Information Systems and Organisation*, pages 63–81. Springer Berlin Heidelberg.

Text Analysis with Ontology Reasoning

Anna Rozeva
Department of Computer Systems and Informatics, University of Forestry, Kliment.Ohridski blv.10, Sofia, Bulgaria
arozeva@hotmail.com

Keywords: Text Analysis, Text Mining, Domain Ontology, Context Model, Ontology Reasoning, Knowledge Management.

Abstract: The analysis of unstructured text when performed by text mining machine learning algorithm results in mining model holding rules for relationships and dependencies among terms extracted by text pre-processing techniques. The obtained mining model represents knowledge derived from the analyzed text which is hard to interpret as it lacks context. Enhancement of its semantic value can be obtained by implementing logic based approach providing formally defined meaning and interpretation mechanism. The generally accepted form for representation of knowledge for existing domain is by domain-specific ontology. The aim of the paper consists in designing a framework for performing ontology instantiation and population with the structures of a complex mining model involving classification and association rules. Procedures have been designed for annotating them with domain concepts and semantic types. The framework provides for turning the mining model into a context model. Ontology reasoning is implemented to validate the input mining model by rule semantic disambiguation and dependency conceptualization. The framework implementation provides for outputting validated domain-related knowledge base in explicit and machine-readable form as a resource that can be adapted for decision support.

1 INTRODUCTION

Text analysis has become a topic attracting increasing research interest and efforts in the recent decade. The reason consists in the enormous and continuously growing amount of published information on Internet. Searching for specific subject area related documents more efficiently as compared to the traditional keyword-based techniques has emerged as one of the most challenging and value-adding problems concerning it. Classification of text documents has turned out to be the most important task in text analysis so far as the retrieval of information from the web is concerned. The text analysis performed by implementing classical syntactical and statistical text mining techniques results in mining models that generally lack sense and ignore the semantics of the input documents.

Most recent researches present techniques and approaches for ensuring the lacking semantic integration in a text mining model and improving machine-learning based text classification (Albitar, Fournier and Espinasse, 2012; Garla and Brandt, 2012). They implement semantic resources like thesauri and ontologies for text "conceptualization" (Albitar et al., 2012) or taxonomical structures for improving feature ranking and semantic similarity measures for projecting text into a feature space (Garla et al., 2012) thus enriching the semantics of the classification output.

Machine accessible semantics of text mining models is achieved by means of ontologies (Horrocks, 2008). They represent hierarchically structured conceptual schemas that give formally defined meanings to concepts referring to a specified domain. As a model of a piece of the world it provides vocabulary describing the aspects of the modelled domain as well as an explicitly stated meaning of its content. The meaning generally represents classification information.

Ontology-based methods have been implemented for refining text analysis and information extraction process. Extraction of links between concepts for capturing domain semantics by using domain specific ontology has been considered in (Morneau and Mineau, 2008).

Ontologies provide semantic frameworks for the interpretation of analyzed information. This becomes possible by using logical formalisms for their representation. Description logic (DL) (Baader et al., 2007) is a knowledge representation formalism distinguished by formal semantics and provision of inference services. It is the foundation of ontology languages like OWL DL (Motik et al., 2012). Ontology based on description logic provides reasoning tools and services for designing and maintaining qualitative ontologies, answering queries over its classes and instances, integrating and mapping ontologies. Therefore ontology reasoning facilitates the design and development of ontologies as well as their deployment in applications.

The intensively researched problem areas mentioned so far provide the motivation for the current work. The aim is to design a framework for implementation of domain ontologies and reasoning services in text document analysis enabling the design of qualitative and validated domain related knowledge. The rest of the paper is organized as folows: Section 2 examines approaches, platforms, mechanisms and applications of ontologically based text analysis; Section 3 presents the conceptual structure of our framework for ontological text analysis; Sections 4 and 5 highlight framework implementation issues and Section 6 concludes with discussion and directions for future work.

2 REVIEW ON ONTOLOGIES IN TEXT ANALYSIS

Text analysis is used in the sense of automatic processing of huge volumes of textual information in order to facilitate its retrieval, management and structuring for research purposes. On the other hand this notion concerns also the extraction of context and meaning from the processed text corpus. Each of these aspects is performed by the implementation of specific information technology.

The technology which converts unstructured or semi-structured natural language text in a form that is suitable for machine processing which results in models of extracted facts, discovered implicit links and generated hypotheses, is referred to as text mining. The models obtained by text mining represent knowledge which needs to be properly interpreted, shared and integrated. This can be accomplished by combining them with corresponding conceptual models which reflect the structure of the subject area the knowledge is referring to and defines the interpretations of its terms, i.e. ontologies. They specify concept meaning and their relations to other concepts in machine-readable form. Ontologies therefore provide for the automatic semantic interpretation of textual information and enhance the benefits of text mining. Further on text analysis will be distinguished as text mining analysis and ontological analysis.

Figure 1 shows a framework for text mining analysis performed on unstructured text corpus that is presented in (Rozeva, 2011a).

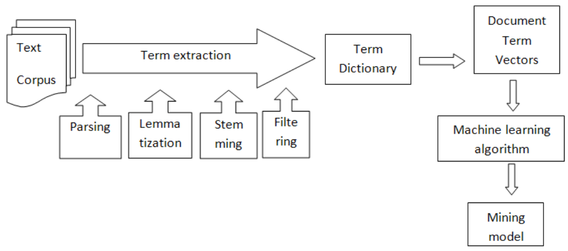


Figure 1: Text mining analysis framework.

Within the above shown framework text is lexically and syntactically processed for the extraction of descriptive terms. They are determined on the basis of evaluating frequency of appearance in a document. Different methods for frequency calculation can be implemented for enhancing the relevance of the extracted terms to the document content. The extracted descriptive terms provide for presenting documents by term vectors. Each document is represented by a vector, which contains the extracted terms and their frequency of occurrence in it. Obviously the document vector matrix is sparse. The matrix is processed with machine learning algorithms and in result mining model is obtained. It stores patterns in the form of classifications, groups and associations mined from the input documents.

Approach for implementation of text mining model within a specified platform for predicting unknown features of new documents is presented in (Rozeva, 2011b).

2.1 Ontological Text Analysis

Basic aspects and practices of using ontologies with text mining are reviewed in (Spasic et al., 2005). So far as ontology provides the terminology of a domain, terms are set to express specialized domain concepts. The mapping of term to concept is considered as the basis for the semantic interpretation of text mining models. Different layers of text annotation by means of ontologies are examined, i.e. lexical, syntactic and semantic.

A solution for the term ambiguity problem as a general text mining challenge is reported in (Bratus et al., 2009). It concerns cases when a term refers to multiple concepts which are normal for natural language texts. The textual case-based reasoning system developed performs taxonomic indexing of text archives which provides for ontology-guided search. It requires text classification and disambiguation. The implemented classification method locates key phrases in lexical taxonomies. Disambiguation uses context to determine the taxonomy regions, which are likely to be most relevant. Algorithm has been designed for ontology-guided text disambiguation.

Use of ontologies during the text analysis process and as an export format for the results obtained is shown in (Witte et al., 2007). Software document ontology has been designed for being automatically instantiated by text mining module. The concepts to be included therein are the ones related to: document structure or lexica; lexical normalization rules; relations between classes and software specific entities. The text mining system for the instantiation of pre-modelled ontology (ontology population) has been designed within the GATE framework (Cunningham et al., 2011) for text engineering, language processing and text analysis. Entities resulting from natural language processing (NLP) populate the ontology which is further on processed for named entity recognition. Entities are matched against a list of terms and in case of match an annotation is added. An ontology-aware component incorporating mappings between term lists and ontology classes is used to assign the proper class in case of term match. Specially designed grammar rules are used to detect and annotate complex named entities. They refer to the annotations created and evaluate the ontology directly. They detect semantic units by combining ontology-based look-up information and extracted noun phrases from documents. By implementation of fuzzy-set based co-reference resolution system detected entities are grouped into chains. Normalization with a set of lexical rules provides entities with canonical names. Rules are stored in their corresponding classes in the domain ontology which allows the inheritance of rules through subsumption. Relations between entities are detected by grammar rules and syntactic analysis. The obtained relations are filtered through the ontology for discarding the semantically invalid ones.

Integration aspects of formal ontologies in OWL-DL format with text mining systems for supporting NLP are discussed in (Witte et al., 2007). Ontology is implemented both as a container for mining results and as a language for document processing. Requirements for the necessary information to be included in ontology for supporting text mining tasks are defined. Issues concerning interfacing ontologies to text mining systems have been considered as well.

Ontology population methods have been considered in (Wang et al., 2005) and (Damljanovic et al., 2009). Learning and populating ontology from linguistic resources is presented in (Wang, et al., 2005). Special attention is turned to the extraction of related concepts and the method applied prevents them to be far apart in the concept hierarchy. Creating adapted workflows for semi-automatically ontology population with text from diverse semantic repositories is reported in (Damljanovic et al., 2009).

Framework for ontology-based text categorization which performs ontology learning by mining input documents and uses the ontology further on for refining the categorization with both supervised and unsupervised machine learning approaches is presented in (Bloehdorn et al., 2005). The framework provides ontology learning algorithms such as for constructing taxonomies with conceptual clustering algorithm; for classifying instances into the ontology by using lexico-syntactic patterns; for extracting labeled relations and specifying their domain and range and for semantic enrichment by finding the appropriate concept from a given ontology, etc. The lexica and concept hierarchies of ontologies are involved in performing text clustering and classification by enhancing the common term representation of documents with concepts extracted from the used ontologies. Ontologies enable specific concepts found in text to be replaced by more general concept representations, i.e. the corresponding superconcept along the path to the root of the concept hierarchy. More recent review on attempts to combining machine learning techniques and ontologies by investigating mining of semantic web data sources with inductive learning techniques is presented in (Bloehdorn and Hotho, 2009).

2.2 Ontology Querying and Reasoning

Basic advantage of the implementation of formal ontologies in text analysis is that it enables the definition of semantic queries on concept instances and automated reasoning based on DL. Reasoning (Horrocks, 2008) has to check that ontology knowledge is meaningful, correct, minimally redundant, richly axiomatised, that queries can be answered over its classes and instances, that individuals matching a query can be retrieved and that the knowledge base is consistent.

Many inference tasks are reduced to subsumption reasoning and to satisfiability.

Efficiency of query answering reasoning task is discussed in (Pothipruk and Governatori, 2005). DL-based reasoning performs inference on a knowledge base consisting of terminological axioms (Tbox) and assertion axioms (Abox). Tboxes refer to the schema of concepts and Aboxes to the names of individuals. Reasoning on the terminological hierarchy, specified by DL first order formulas, checks the fulfillment of the following logical requirements: concept satisfiability; class subsumption; class consistency; instance checking.

A subset of reasoning rules (Wang et al., 2004) is shown in Table 1.

Table 1: Sample OWL ontology reasoning rules.

Transitive_ Property	(?P rdf:Type owl: TransitiveProperty)∧ (?A ?P ?B)∧ (?B ?P ?C)⇒ (?A ?P ?C)
subClassOf	(?a rdfs:subClassOf ?b) ∧ (?b rdfs:subClassOf ?c) ⇒ (?a rdfs:subClassOf ?c)
subPropertyOf	(?a rdfs:subPropertyOf ?b) ∧ (?b rdfs:subPropertyOf ?c) ⇒ (?a rdfs:subPropertyOf ?c)
disjointWith	(?C owl:disjointWith ?D) ∧ (?X rdf: type ?C) ∧ (?Y rdf: type ?D)⇒ (?X owl:differentFrom ?Y)
inverseOf	(?P owl:inverseOf ?Q) ∧ (?X ?P ?Y) ⇒ (?Y ?Q ?X)

The proposed optimization approach concerns Abox reasoning, as the basis for query answering. Two types of queries allowed by DL are considered. The Boolean query represents instance checking and the non-Boolean consists in retrieving Abox content. The approach addresses the problem of efficient answering a query in a DL-based semantic web system implementing single ontology and multiple data sources.

Architecture for ontology reasoning is proposed in (Pan, 2007). It supports ontology languages providing for the definition of customized data types and customized data type predicates. It allows new data type reasoners to be added into the architecture without affecting the basic concept reasoner.

Ontology for modeling context and supporting context reasoning has been designed in (Wang et al., 2004). Its upper level captures general concepts about the basic context. It allows hierarchical adding of domain-specific ontologies. Logic reasoning on this ontology consists in checking the consistency of context information and deriving high level implicit context from low level explicit context.

Ontologies provide the vocabulary that enables the semantic markup of web resources. Thus they express the terminological part of knowledge structured in taxonomy of concepts and properties. DL based language as OWL enables reasoning for ontology checking, classification and recognition of class instances. Rules on the other hand describe logical dependencies between the ontology elements and as such represent the deductive type of knowledge. Rule types are: standard – for chaining ontologies' properties; bridging – for reasoning across several domains; mapping – ontologies in data integration; querying – for expressing complex queries in ontology vocabulary and meta – for facilitating ontology engineering (acquisition, validation and maintenance).

As considered in (Golbreich, 2004) completeness of inferences can be obtained on the basis of the whole domain knowledge available both in ontologies and rule bases. Approach for reasoning by combining them is designed based on Semantic web standards language OWL and SWRL (Horrocks et al., 2004). The platform for combining DL and rule reasoning is shown in Figure 2.

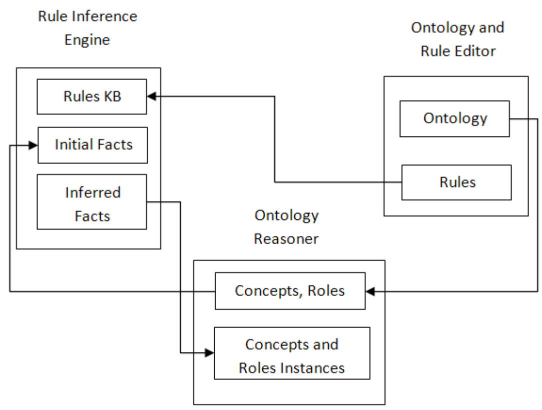


Figure 2: Combining ontology and rule reasoning.

Modeling approach for developing rule-based application for the web is presented in (Canadas, Palma and Tunes, 2009). It implements ontologies for describing the concepts and their relationships in the application domain and rules for formalizing the inference logic thus providing for increasing the amount of knowledge represented in ontologies.

3 CONTEXT-BASED TEXT ANALYSIS FRAMEWORK

The review of related work on approaches, methods, technologies and tools presented in the previous section has motivated the design of a framework for analysing web text documents resulting in a knowledge base that can be queried with automatic semantic reasoners for inferring implicit facts. In our previous work (Rozeva, 2012) we've presented an approach for implementing pre-defined ontology at:

- Text pre-processing step for filtering terms that don't map to ontology individuals;
- Post-processing the rules learnt by implementing mining algorithm.

The approach implemented in the current work uses the rules model obtained by processing the text with machine learning algorithm for instantiation of domain ontology. This provides for enhancing the efficiency of the representation of the analysed domain by turning the mined model into a context model. The context model being ontologically based ensures logically validated classification and logic reasoning. The framework is shown in Figure 3.

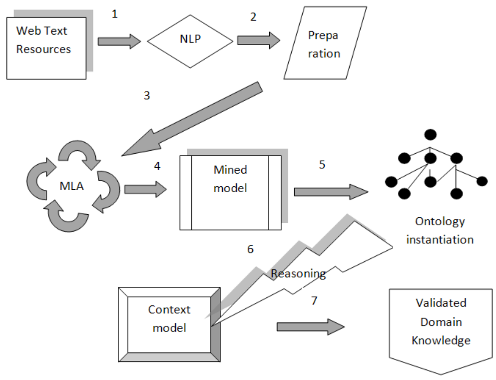


Figure 3: Ontological text analysis framework.

The modules in the framework are:

- NLP – natural language text processing resulting in term extraction;
- Preparation of document vector matrix for algorithmic processing;
- MLA - machine learning algorithm producing mined model;
- Ontology instantiation with the mined model producing semantically labelled context model;
- Ontology reasoning for inferring new facts resulting in validated domain related knowledge.

MLAs that are usually applied for text analysis purposes are: discovering groups of text documents by clustering; classification (grouping according to pre-defined categories) and associations.

The core of the proposed framework is the instantiation of ontology from the rules of mined model obtained by processing the text with machine learning algorithm. This process refers to defining classes, arrangement of classes in a taxonomic hierarchy, defining properties of classes and allowed values for them, filling in the property values for the individuals of the classes. The framework implements classification and association rules machine learning algorithms. The classification algorithm provides for building the ontology class hierarchy. The association rules algorithm enables the definition of non-taxonomic relations between classes. The method for ontology building is inspired by the method presented in (Elsayed et al., 2007). Their approach uses the rules obtained by mining structured data for mapping tree nodes to OWL classes, tree branches to OWL classes and leaf nodes to individuals. The ontology building algorithm implements functions for: getting the branches of a node and the branch of a leaf node; getting the class that represents a branch and for creating individual for a leaf node. This algorithm is adapted in the proposed framework for taking input from rules mined from text and is enhanced with the definition of additional objectType property of the classes. The method for enriching the ontology with non-taxonomic relations is based on similar research held on lexico-syntactic patterns from domain specific dictionary for extracting relations between concepts shown in (Maedche and Staab, 2000) as well as on an approach for mining dictionary databases for ontology generation purposes presented in (Deliyska et al., 2012).

3.1 Building Ontology Taxonomy

The proposed algorithm for ontology instantiation comprises two modules. The first one creates ontology from a decision tree mining model. The second one performs ontology enrichment by creating relations between ontology classes from the results of an association rules model. The general structure of decision tree model shown in Figure 4 provides the input for the ontology building module, i.e.:

- Predictable attribute name;
- Node ID and node name;
- Node type – root, interior or distribution;
- Node children cardinality – 2 or 0;
- Node parent;
- Node description – inputAttributeValue_ missing or inputAttributeValue_existing;
- Node rule, containing attribute and predicate values;
- Node distribution – probability of predictable

attribute values.

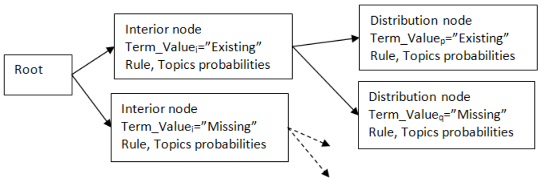


Figure 4: Decision tree text analysis model.

The predictable attribute in a text classification task is the document category (Topic). The tree is split by the presence or absence of an input attribute value. The input attribute in the text classification is Term with values extracted in the NLP stage. Cardinality for interior nodes is two, denoting existing or missing term value. Each node has probability distributions of predictable attribute values attached. The algorithm steps for mapping the decision tree to ontology components are:

Step 1: Define independent class for the distribution with classification topics and probability;

Method:

```
Class Distribution = new(OWL:class)
Distribution.Id= Distribution.name
DatatypeProperty DistributionDP=new
(owl:DatatypeProperty)
Class Topic = new(OWL:class
subClassof Distribution)
Topic.Id= Topic.name
DatatypeProperty TopicDP=new
(owl:DatatypeProperty)
Class TopicProbability =
new(OWL:class subClassof
Distribution disjointWith Topic)
TopicProbability.Id=
TopicProbability.name
DatatypeProperty
TopicProbabilityDP=new
(owl:DatatypeProperty)
```

Step 2: Define independent class for the nodes' rules;

Method:

```
Class NodeRule = new(OWL:class)
NodeRule.Id= NodeRule.name
DatatypeProperty NodeRuleDP=new
(owl:DatatypeProperty)
Class Predicate = new(OWL:class
subClassof NodeRule)
Predicate.Id= Topic.name
DatatypeProperty NodeRuleDP=new
(owl:DatatypeProperty)
Class TermValue = new(OWL:class
subClassof NodeRule disjointWith
Predicate)
TermValue.Id= TermValue.name
DatatypeProperty TermValue DP=new
(owl:DatatypeProperty)
```

Step 3: Define independent class for the nodes' probability;

Method:

```
Class Probability = new(OWL:class)
Probability.Id= Probability.name
DatatypeProperty ProbabilityDP=new
(owl:DatatypeProperty)
```

Step 4: Define the root class for the taxonomy of the interior and terminal nodes;

Method:

```
Class All = new(OWL:class
disjointWith Distribution
disjointWith NodeRule disjointWith
Probability)
All.Id= All.name
DatatypeProperty All=new
(owl:DatatypeProperty)
```

Step 5: Define the class taxonomy from the interior and terminal nodes;

Method:

```
BEGIN
 For each node DN
 Class C=new (owl:Class)
 C.Id= DN.name
 DatatypeProperty DP=new
(owl:DatatypeProperty)
Dp.Id= DN.name+"_Value"
Dp.AddDomain(C)
     For each ChildNode CN of Get-
Children(DN)
 Dp. AddDomain (CN.Get-Class ())
     endfor
 endfor
End
```

Step 6: Define properties and make the relation of the class taxonomy to the independent classes;

Method:

```
ObjectProperty hasDistribution =
new(Owl:FunctionalObjectProperty)
ObjectProperty hasProbability =
new(Owl:FunctionalObjectProperty)
ObjectProperty hasRule =
new(Owl:FunctionalObjectProperty)
AddSubclassOf(All, hasDistribution)
AddSubclassOf(All, hasProbability)
AddSubclassOf(All, hasRule)
```

Step 7: Create individuals for ontology classes;

Method:

```
BEGIN
For each node DN
 Individual I = new
(owlclass:Individual)
     Foreach Topic T
Distribution.I+=
 T+TopicProbability
     Endfor
 NodeRule.I+= Predicate+TermValue
 ADDType(Distribution.I, I)
 ADDType(NodeRule.I, I)
Endfor
End
```

The ontology instantiated by this module involves the hierarchical relations between the defined classes. By including non-hierarchical relations as well further enrichment and refinement of the generated ontology will be obtained.

3.2 Ontology Enrichment

Ontology enrichment is achieved by the second module in the framework which introduces non-hierarchical relations between ontology classes. These relations are extracted by applying the association rules algorithm of Srikant and Agrawal (as cited in Maedche and Staab, 2000) on the processed text documents. Association rules are in the form X →Y with measures for confidence and support, where ‘X’ and ‘Y’ represent items in a transaction set.

When applied to hierarchically structured ontology classes the right side of the rule involves the ancestors of the particular item as well. The resulting rules contain ‘Y’ that isn’t ancestor of ‘X’ and the rule X→Y isn’t subsumed by one involving their ancestors. The association rules produced by the text mining algorithm are to be implemented as properties between ontology classes’ individuals. The algorithm implements a method which assumes that individual corresponding to antecedent ‘X’ of the association rule exists in ontology. It involves the following method:

Method:

```
For each rule X→Y
If exist(owlclass.Individual=”Y”)
Begin
 ObjectProperty Prop_Name =
 new(Owl:ObjectProperty)
 AssertObjectProperty (X,
 Prop_Name, Y)
End
ElseIf
Begin
 Class Association = new(OWL:class
 disjointWith class.X)
 Association.Id= Association.name
 DatatypeProperty Association =new
 (owl:DatatypeProperty)
 Association:Individual=”Y”
 AssertObjectProperty (X,
 Prop_Name, Y)
End
Endif
EndFor
```

The method checks the presence of an individual ‘Y’ in ontology and creates object property which relates it to the individual ‘X’. If not present new class is added to ontology as disjoint with the class individual ‘X’ belongs to. Individual ‘Y’ is added to this class. Object property is created and is asserted to relate individuals ‘X’ and ‘Y’.

4 FRAMEWORK IMPLEMENTATION

The text mining module of the framework has been implemented in Microsoft SQL Server 2008 (Microsoft SQL Server, 2013). The classification model obtained by mining the text with Microsoft Decision Tree algorithm is shown in Figure 5.

NodeName	00000000000
NodeCaption	(e-governance) = Missing
NodeType	3 (Interior)
ChildrenCardinality	2
ParentNode	000000000
NodeRule	predicate, attribute
MarginalRule	predicate, attribute
NodeProbability	0,755555555555556
MarginalProbability	0,871794871794872
NodeDistribution	Topic Distribution
NodeSupport	34

Figure 5: Decision tree mining model.

Topic’s distribution created for each tree node is shown in Figure 6. Term extraction and term lookup transformations have been implemented for text preparation. The document corpus consists of conference papers on e-Governance.

ATTRIBUTE_NAME	ATTRIBUTE_VALUE	SUPPORT	PROBABILITY	VARIANCE	VALUETYPE
Topic	Missing	0	0	0	1 (Missing)
Topic	A	6	0,184210526315789	0	4 (Discrete)
Topic	B	5	0,157894736842105	0	4 (Discrete)
Topic	C	10	0,289473684210526	0	4 (Discrete)
Topic	D	13	0,368421052631579	0	4 (Discrete)

Figure 6: Classification topic distribution.

For instantiating the ontology the approach for mapping database schema to ontology presented in (Yankova et al., 2008) has been implemented. The general form of the script statement mapping table column to ontology classes is:

```
_columnToURI.putForward("Table.Field",
"Ontology:hasField)
```

The instantiated ontology with the decision tree model by implementing the algorithm in Protégé 4.2 (Protégé 4 User Documentation, 2013) is shown in Figure 7.

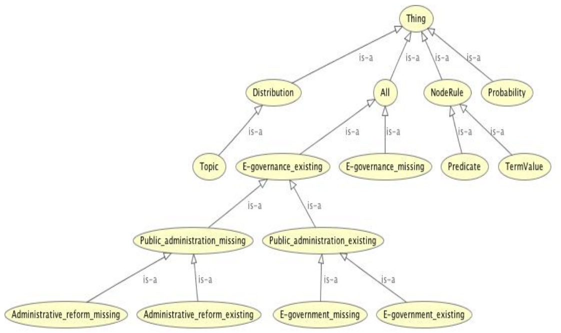


Figure 7: Ontology instantiated with decision tree model.

The association rules mining model obtained by processing the text corpus with the Microsoft Association Rules algorithm is shown in Figure 8.

Support	Size	Itemset
2	2	EU E-GOVERNANCE = Existing, e-governance center = Existing
2	2	public sphere = Existing, public service = Existing
2	2	public service = Existing, public administration = Existing
2	2	knowledge economy = Existing, public sector = Existing
2	2	civil servant = Existing, public service = Existing
2	2	public management = Existing, public administration = Existing
2	2	public management = Existing, public service = Existing
2	2	company management = Existing, quality management = Existing
1	2	administrative reform = Existing, e-government = Existing
1	3	administrative reform = Existing, public sphere = Existing, e-government = Existing
1	3	administrative reform = Existing, public service = Existing, e-government = Existing
1	3	administrative reform = Existing, public administration = Existing, e-government = Existing

Figure 8: Association rules mining model.

The ontology enriched with the mined rules is shown in Figure 9. The rule "administrative reform" = Existing → e-government = Existing has been inserted in ontology as shown with arrows by:

- Creating new class Reform;
- Creating individuals for the class, i.e. Administrative, Management and Budget;
- Creating object property "involves";
- Asserting the "involves" property to "e-government" individual of class "E-government_existing" with the individual "Administrative" from the Reform class.

The ontology instantiated by the proposed framework has been classified with Pellet reasoner (Pellet Reasoner Plug-in for Protégé 4, 2013) – Figure 10.

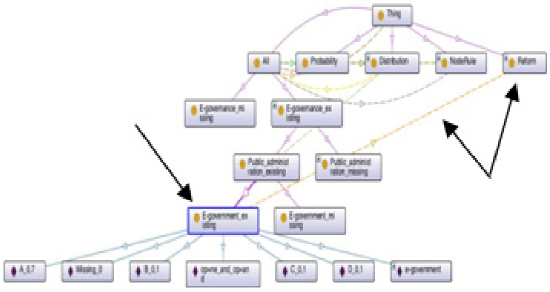


Figure 9: Enriched ontology with association relations.

Figure 10: Classified ontology.

5 REASONING EXAMPLES

A classified ontology can be searched by logical queries. Figure 11 presents sample DL query for retrieving class individuals with restriction on dataType property. Class queries get subclasses or descendents in the hierarchy or the superclass of a class.

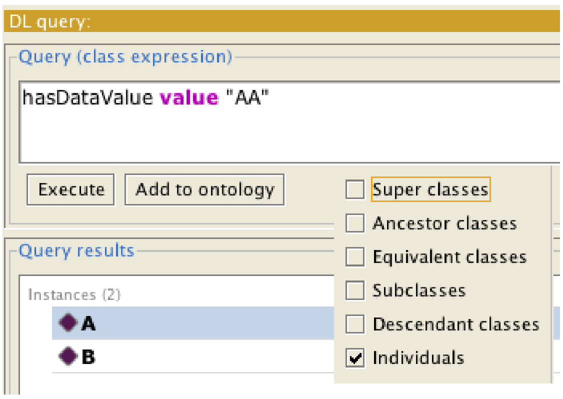


Figure 11: DL query on classified ontology.

Figure 12 presents sample OWL2 query in graph view. The TBoxes, ABoxes and RBoxes are used to provide concept relations, relations between individuals and concepts and rules to the query.

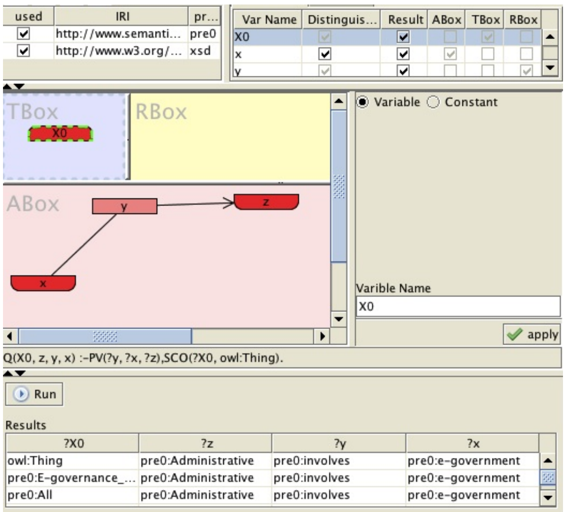


Figure 12: Owl2 Query on classified ontology.

6 CONCLUSIONS

A framework for text analysis involving the building and instantiation of ontology from analysis model obtained by text mining has been proposed. It has been designed with the aim of enhancing the semantic content of analysis model rules and turning them into a knowledge rule base which enables processing with logic reasoning. Most of the reviewed work on ontologies in text analysis considers natural language processing and validations with dictionaries and domain ontologies. Current approach addresses algorithmically processed text. It is considered that ontology obtained from mining model can be treated as more relative representation of text corpus overall content. By structuring a rule model into ontology the proposed framework ensures model's semantic enrichment. The contribution of the paper is the defined approach for representing a complex text mining model as concise domain ontology. The examined mining model is complex because it contains both classification and association rules. Thus ontology proves to be the means for integrating knowledge rules from different types of machine learning text processing. It also provides the processing tools for extracting the logical content and meaning of the integrated rules.

The framework couples a text mining and ontology instantiation modules. Methods are created for the automatic mapping of rules into ontology elements – classes, object and data properties and instances. The framework has been implemented for classification and associations analysis of selected text corpus. Classification as the most typical analysis task produces rules that enable the natural mapping to ontology class hierarchy. The framework implementation resulted in ontology that has been successfully classified by a reasoner. Examples for searching the classified ontology with description logic and OWL2 queries have been provided. It's claimed that the ontology mapping besides for mining models integration turns the mining model into a context model with enhanced semantic meaning of its rules, initially extracted from text by machine learning algorithm.

The framework presented has been focused primarily on the terminological part of the process of ontology building. Future work is intended in enhancing its axiom part as well as on mapping of other text mining models on ontology.

REFERENCES

Albitar, S., Fournier, S., Espinasse, B., 2012. The Impact of conceptualization on text classification. In X. Wang, I. Cruz, A. Delis and G. Huang (Eds.), *Web Information Systems Engineering – WISE 2012*, Lecture Notes in Computer Science, Vol.7651, pp.326-339, Springer-Verlag Berlin Heidelberg.

Baader, F., Horrocks, I., Sattler, U., 2007. Description logics. In *Handbook of Knowledge Representation*. Elsevier.

Bloehdorn, S., Cimiano, P., Hotho, A., Staab, S, 2005. An ontology-based framework for text mining. In *LDV Forum 20(1):87-112*.

Bloehdorn, S., Hotho, A., 2009. Ontologies for machine learning. In S.Staab and R.Studer (Eds.), *Handbook on Ontologies*, International Handbooks on Information Systems, pp.637-661, Springer-Verlag Berlin Heidelberg.

Bratus, S., Rumshisky, A., Magar, R., Thompson, P., 2009. Using domain knowledge for ontology-guided entity extraction from noisy, unstructured text data, In *Proceedings of the Third Workshop on Analytics for Noisy Unstructured Text Data*, pp.101-106, ACM, New York, NY, USA.

Canadas, J., Palma, J., Tunes, S., 2009. InSCo-Gen: A MDD tool for web rule-based applications, In M. Gaedke, M. Grossniklaus, O. Diaz, (Eds.), ICWE, Lecture Notes in Computer Science, Vol.5648, pp.523-526, Springer.

Cunningham, H., Maynard, D., Bontcheva, K., 2011. *Text processing with GATE (Version6)*, University of Sheffield Department of Computer Science.

Damljanovic, D., Amardeilh, F., Bontcheva, K., 2009. CA manager framework: creating customized workflows for ontology population and semantic annotation. In *Proceedings of the Fifth International Conference on Knowledge Capture*, pp.177-178.

Deliyska, B., Rozeva, A., Malamov, D., 2012. Ontology building by dictionary database mining, In *Proceedings of the 38th International Conference Applications of Mathematics in Engineering and Economics (AMEE'12), AIP Conference proceedings*, Vol. 1497(1), pp.387-394

Elsayed, A., El-Beltagy, S., Rafea, M., Hegazy, O., 2007. Applying data mining for ontology building, In *Proceedings of ISSR*.

Garla, V., Brandt, C., 2012. Ontology-guided feature engineering for clinical text classification, *Journal of Biomedical Informatics*, Vol. 45(5), pp.992-998.

Golbreich, C., 2004. Combining rule and ontology reasoners for the semantic web, In G. Antoniou, H. Boley,(Eds.), *Rules and Markup Languages for the Semantic Web*, pp.6-22.

Horrocks, I., 2008. Ontologies and the semantic web. *Communications of the ACM*, 51(12): 58-67.

Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., Dean, M., 2004. SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission 21 May 2004, http://www.w3.org/Submission/SWRL/.

Maedche, A., Staab, S., 2000. Mining ontologies from text, In R. Dieng, O. Corby (Eds.), *EKAW 2000, Lecture Notes in Artificial Intelligence*, Vol.1937, pp.189-2002.

Microsoft SQL Server, 2013. SQL Server DevCenter, http://www.msdn.microsoft.com/en-us/sqlserver/default.

Morneau, M., Mineau, G., 2008. Employing a domain specific ontology to perform semantic search. In P. Eklund and O. Haemmerle (Eds.), *Conceptual Structures: Knowledge Visualization and Reasoning*, Lecture Notes in Computer Science, Vol.5113, pp.242-254, Springer-Verlag Berlin Heidelberg.

Motik, B., Patel-Schneider, P.F., Parsia, B., 2012. OWL2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation 11 December 2012, http://www.w3.org/TR/owl2-guide/.

Pan, J., 2007. A flexible ontology reasoning architecture for the semantic web, *IEEE Transactions on Knowledge and Data Engineering*, Vol.19 (2), pp.246-260.

Pellet Reasoner Plug-in for Protégé 4, 2013. Clark & Parsia, http://www.clarkparsia.com/pellet/protege.

Pothipruk, P., Governatori, G., 2005. A formal ontology reasoning with individual optimization: a realization of the semantic web, In M. Kitsuregawa et al.(Eds.), *Web Information Systems Engineering (WISE 2005)*, Lecture Notes in Computer Science, Vol.3806, pp.119-132, Springer.

Protégé 4 User Documentation, 2013. http://www.protegewiki.stanford.edu/wiki/Protege4UserDocs.

Rozeva, A., 2011a., Approach for mining text databases, In *Proceedings of the III International Conference "E-governance"*, Sozopol, Bulgaria, pp.82-87.

Rozeva, A., 2011b., Mining model for unstructured data, In *Proceedings of the sixth International Conference "Computer Science'11"*, Ohrid, Macedonia, pp.411-415.

Rozeva, A., 2012. Application of ontologies for knowledge generation. *Proceedings of the IV International Conference "E-governance"*, Sozopol, Bulgaria, pp. 162-170.

Spasic, I., Ananiadou, S., McNaught, J., Kumar, A., 2005. Text mining and ontologies in biomedicine: Making sense of raw text. *Briefings in Bioinformatics*, Vol.6 (3), pp.239-251, Henry Steward Publications 1467-5463.

Wang, T., Maynard, D., Peters, W., Bontcheva, K., Cunningham, H., 2005. Extracting a domain ontology from linguistic resource based on relatedness measurements. In *IEEE/WIC/ACM International conference on Web Intelligence (WI'05)*, pp.345-351.

Wang, X.H., Gu, T., Zhang, D.Q., Pung, H.K., 2004. Ontology based context modeling and reasoning using OWL. In *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp.18-22.

Witte, R., Li, Q., Zhang, Y., Rilling, J., 2007. Ontological text mining of software documents. In Z. Kedad, N. Lammari, E. Metais, F. Meziane, Y. Rezgui (Eds.), *NLDB 2007*, Lecture Notes in Computer Science, Vol.4592, pp.168-180, Springer-Verlag Berlin Heidelberg.

Witte, R., Kappler, T., Baker, C., 2007. Ontology design for biomedical text mining. In C. Baker, K. Cheung (Eds.), *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, Vol.13, pp.281-313, Springer.

Yankova, M., Saggion, H., Cunningham, H., 2008. Adopting ontologies for multisource identity resolution, In A. Duke, M. Hepp, K. Bontcheva, M.B. Villain (Eds.), *OBI Vol.308 of ACM International Conference Proceeding Series*, pp.6-15, ACM.

Abstracting Imperative Workflow to Declarative Business Rules

Lex Wedemeijer
Department of Computer Science, Open University, Valkenburgerweg 177, Heerlen, The Netherlands
Lex.Wedemeijer@ou.nl

Keywords: Business Rules, Requirements Engineering, Workflow Model, Controlflow Pattern, Relation Algebra.

Abstract: Large business administrations rely on workflow systems to coordinate their business processes. In practice, workflow models are the blob-and-arc diagrams that outline required activities for dealing with an incoming event. In general however, user understanding is served better by the business rules approach. The Business Rules Manifesto advocates to express in declarative business rules what should be complied with, but to abstract from how to accomplish that by way of procedures. In this paper, we transform the main procedural components of imperative workflows to declarative business rules. The transformation results in two rules that still reflect the procedural nature of workflow, but more abstract than the corresponding workflow model. Once a workflow is transformed to declarative rules, these rules can be merged with other, content-aware business rules or pruned for unnecessary restrictions. The declarative rules and relations may capture business requirements about work processing better than blob-and-arc diagrams of imperative workflows.

1 INTRODUCTION

Have you ever done a Sudoku puzzle? Its rules are surprisingly simple, yet the challenge of Sudoku is that there is no simple workflow how to solve the puzzle. The same applies to workflow models in business: the rules governing the day-to-day work are rather simple, yet the implemented workflows and procedures that prescribe how business workers and applications should execute the work, are much more complicated.

Our point is that work should be done to comply to the rules set by the business, but an operational workflow may impose additional restrictions for implementational reasons having little business relevance. In keeping with the Business Rules Manifesto (2003), we believe that business rules should be expressed as explicit constraints on behavior, independent of how the rules may currently be implemented in process descriptions or workflow diagrams. In practice, users regard workflow models often just as blob-and-arc diagrams that depict how an incoming event should be dealt with. Such diagrams tell the users what to do and when, but not why. Transforming the imperative workflow model into the format of declarative business rules opens the road to identify the rules based on legitimate business requirements, and to eliminate the ones that were added for implementational reasons.

Using the Relation Algebra approach for rules coined by (Joosten, 2007), restrictions of the workflow are captured in a single declarative rule (section 5) which builds upon binary relations corresponding to the various structural components of common workflow models. Next, we express the business goal of the workflow by a second rule (section 6). Thus, the imperative constraints and the goal reached by the workflow are exposed at the same level of abstraction, and in a format compatible to other business rules. The workflow rules, previously encapsulated in descriptions or diagrams, become amenable for practicable validation by the user community, and for conflict analysis and optimization by rule designers.

The paper outline is as follows. Section 2 sets the stage for basic workflow models. Section 3 outlines declarative business rules and notions used in the paper. Sections 4 and 5 explain how the basic constructs of workflow models are transformed to assertions of Relation Algebra. Section 6 outlines how the workflow process is driven by way of the Control Principle. Section 7 discusses elaborations. Section 8 concludes the paper and indicates some directions for further research.

2 BASIC WORKFLOW MODELS

We outline features of conventional workflows, and explain the core notion of imperative workflow.

2.1 Basic Workflow Models

The Workflow Management Coalition (1999) defines a workflow process as a formalised view of a business process. It is presented as a coordinated set of process activities aimed to achieve a common business goal. Figure 1 depicts a typical workflow model as a blob-and-arc diagram. The example, adapted from the WMC'99 technical report, contains the four components of typical workflows: sequence, parallel flow, selective flow, and iterative loop (Aalst, Hofstede, et al., 2003). Sections 4 and 5 discuss how to transform each to relations and declarative rules.

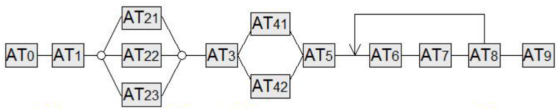


Figure 1: A typical workflow model.

A workflow model may be read in several ways.

The more usual interpretation is of the flow as a kind of roadmap. Once an activity is completed, the roadmap is consulted to answer the 'now what, where to go next' type of question. The answer is what may be called a 'statement of advice' (Witt, 2012). Upon completion of activities, it indicates which activities in the processing chain may be executed next. Typical wordings of this kind are 'you may now start activity A', or 'activity B is now enabled'. This interpretation of workflow is forward-looking in time: what may come next, and we will refer to it as the indicative view of workflow.

We prefer a more rigorous interpretation: the flow specifies compulsory precedence. Prior to completing an activity, all of the preceding activities should also have completed, out of necessity. The question here is 'what must have come before', and the answer takes the form of a business rule. The rule has 'immediate' enforcement, i.e. the preceding activities must have completed, and this may never be violated. This interpretation looks backward in time: what must have come before. We will refer to this as the imperative view of workflow.

In this paper, we are only interested whether an activity has completed or not. The timestamp of its completion or duration of the activity are not recorded. We abstract from details such as activity life cycle (Russell, Aalst et al., 2006) comprising steps like enabling, allocation to resource, work initiation, data transacting or recording the data outputs. We also abstract from issues such as resource responsible for enactment, execution cost, etc.

2.2 Case Management

A workflow models how an incoming case is processed. Therefore, it always features some trigger (sometimes even more than one) where a new case may enter the business process. Successive activities are then executed for the case (in parallel and/or in series) until all work is done and, by assumption, the intended business goal is achieved. Spontaneous generation of new cases somewhere along the line is prohibited in the imperative view of workflow. Interestingly, the indicative view of workflow allows new cases to suddenly emerge, but it is a tacit assumption that cases should start only at a trigger.

The notion of 'case' or 'workflow instance' constitutes an essential difference between workflow models and business rules in general. A declarative business rules model specifies rules that should be complied with, but it does not require the notion of any particular 'case' being managed. If any rule is violated, there is work to do, regardless how or what caused the violation.

Hence, in order to transform the imperative workflow model into declarative rules, we need to include the concept of 'case' or 'working instance' into our models. Surprisingly, we find that just this concept, together with the concept of 'activity type', provide a basic structure (figure 2) that is sufficient to transform an imperative workflow model into its equivalent declarative rule model.

3 DECLARATIVE RULES

This section outlines features of declarative business rules, and the structural components of the approach for business rules that we will be using in the paper.

3.1 Relation Algebra

We use binary Relation Algebra (Maddux, 2005) to specify and formulate our declarative rules. For readers familiar with relational database modeling, a few major differences may be mentioned. The notion of concept as we use it, is comparable to entities, but our concept is just a single column which is key, and has no attributes. Binary relations as we use them, are not foreign-key pointers, but are defined as subsets of the Cartesian Product. They have many-to-many cardinality, unless specified otherwise.

Time is not a native notion of binary Relation Algebras. Indeed, none of the formulas and rules that we discuss in this paper will refer to timestamps or intervals. For this reason, some authors call rules formulated in binary Relation Algebra 'invariant'.

3.2 Related Approaches

Our terms declarative and imperative are interpreted differently in (Mendling et al., 2009). Their understanding of 'declarative' is simply that the given behavior satisfies all requirements. The basic workflow models that we consider will satisfy all requirements, perhaps not instantly but eventually, still we do not consider them to be declarative.

Linear Temporal Logic (LTL) is an excellent approach to study workflows in great detail (Maggi et al., 2011). LTL extends first-order logic with a linear, discrete model of time. A workflow suite, called Declare, uses LTL to model and execute business processes. A main difference with our approach is again the notion of time, which is a prominent feature in LTL, but absent from ours. For example, precedence is sometimes interpreted to mean that an activity should not be *started* prior to the completion of the preceding one. This would imply that activities have a certain duration, but as noted before, time is irrelevant to our approach and we do not follow this interpretation.

Protocol modelling (McNeile and Simons, 2006) also does away with temporal aspects and the notion of cases. Aspect oriented models enable state transitions while taking multiple crosscutting concerns and business constraints into account. This approach takes the indicative view of workflow when it labels state transitions as 'desired' (Wedemeijer, 2012).

3.3 Structure of the declarative model

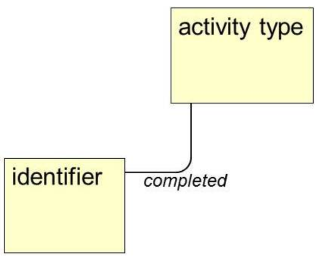


Figure 2: Basic structure of the declarative model.

To transform imperative flows to declarative rules of Relation Algebra, we must first specify a suitable structure to capture the terms, facts and rules of imperative workflows. The basic structure of our declarative model (figure 2) is attractively simple, containing just two concepts as defined below. We will expand this basic structure with other features as needed:

concept	semantics
[identifier]	is: (a pool of) available case identifiers, each associated to a particular workflow case, e.g. a case 'customer request 123'
[activity type]	is: (the set of) activity types. Each activity type may be executed (instantiated) any number of times, in order to achieve the goal of the business process

The declarative model for workflows specifies a number of binary relations on these concepts. The most important one, called *completed*, records workflow progress. A tuple (i,A) in this relation *completed*, sometimes called a transition, records that this particular case identifier, i, has successfully been processed by the particular activity type, A. For ease of reading, we write relation names in italics:

relation	semantics
[identifier] *completed* [activity type]	is: (the recording that) all work of the activity type has been successfully completed for (the case associated with) identifier i.

The *completed* relation represents the audit trail of the work done on a particular case. In accordance with compliance regulations and good records-keeping (McKemmish et al., 2006), tuples may be added into this relation, but they may never be altered or deleted thereafter: an activity cannot be un-completed. And to safeguard referential integrity, we cannot delete an identifier or activity type once it is recorded in the *completed* relation.

Notice that we abstract from a lot of attributes commonly included in audit trails, such at deadlines being set, the exact times of start and completion, business resource that executed the work, or the actor taking responsibility for the work done.

We assume the *completed* relation to be total, i.e. a case identifier is recorded only if it *completed* at least one activity. This is because we are interested only in identifiers associated with actual work done, not in possible future work. The reverse is not required: an activity type may exist even if no case has ever completed that activity.

Instead of *completed*, an *started* relation might have been modeled. Again, we are interested only in actual work done, not in ongoing execution of activities. A similar argument is used in Petrinet theories, the formal foundation of most workflow models.

4 TRANSFORMING FORWARD FLOW

Many business processes can be represented in simple workflow models where activities are executed in sequence, or perhaps in parallel. This section outlines how common components are transformed: sequence, synchronize (AND-join), and selective OR-splits (disable, exclusive split), as in figure 3.

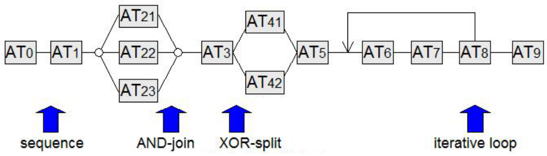


Figure 3: Common components in workflow.

4.1 Sequence / Precedence

4.1.1 Sequencing in Workflows

This is the most common pattern in workflows. Most workflow diagrams depict sequencing by an arc from one blob, representing some activity type A_0, to some next blob labelled type A_1.

The indicative interpretation of such sequencing is: after A_0 is completed, then A_1 should be executed next. Not having completed activity A_1 after A_0 means that the rule for sequencing is temporarily violated. There is work to do while the violation lasts.

In the imperative interpretation of workflow, the arc from A_0 to A_1 represents strict precedence: if the second activity A_1 is completed for some case instance, then A_0 must have already been completed for that case instance. Or: if completion of A_0 is not on record, then the rule says that completion of A_1 is impossible. Not having completed A_0 prior to A_1 means that the imperative workflow rule is violated, and this violation is never permitted.

4.1.2 Rule for Precedence

To capture precedence as a declarative rule, the *precedes* relation on activity types is used (see table).

relation	semantics
[activity type] *precedes* [activity type]	is: the precedence relation among activity types. A tuple (A0,A1) in this relation means that only if an activity of type A0 has completed, may a corresponding activity of type A1 also be completed.

The precedence rule can be formulated in first-order logic by way of the *precedes* relation:

> for any i∈[identifier], and A_1∈[activity type], we have: if i *completed* A_1 then there must exist some activity type A_0 such that i *completed* A_0 and A_0 *precedes* A_1.

which in Relation Algebra reads:

$$completed \subset completed \circ precedes \qquad (1)$$

4.1.3 Precedence for Triggers

A problem with the sequencing rule is that not every activity type is preceded by another. Such activity types are customarily called triggers, or initial activities, and they are important because they set the workflow in motion.

At first glance, triggers invalidate assertion (1), as completion of an identifier cannot be recorded for an initial activity type because a proper tuple in the *precedes* relation is absent. We solve this by adjusting the definition of relation *precedes*: initial activity types are recorded by way of self-referring tuples (A_0,A_0). By recording such tuples in *precedes*, assertion (1) also covers initial activities.

4.1.4 About the *Precedes* Relation

Basic properties of the binary *precedes* relation correspond nicely to important features of imperative workflows. As we are merely concerned with transforming the workflow to declarative rules, we refrain from a deeper analysis of this and other relations to be defined. We take quality issues for granted, such as the workflow being well-designed with respect to liveness, deadlock etc.

The *precedes* relation is not univalent, and an activity type may well precede several others, corresponding to a so-called split in the workflow. It establishes what may be called a multiple-instance pattern (Aalst et al., 2003). The subsequent activities may be executed and completed in parallel along separate branches of the flow.

It is not total. An activity type may be a last one, a terminating activity in the workflow. Nor is *precedes* an injective relation, as more than one activity type may precede an activity type A_X. Assertion (1) will ensure that at least one precedent is completed prior to the completion of A_X. This is the common OR-join of workflow models.

We explained above that self-referring tuples are recorded to capture triggering (initial) activities. As a result, the binary *precedes* relation is surjective.

4.2 Synchronizing AND-Joins

4.2.1 Synchronization in Workflows

Sometimes an activity may only be completed after completion of more than one activity. Known as AND-join or more formally as synchronization point in workflow models, it is not captured by rule (1).

4.2.2 Rule for Multiple Precedence

To capture AND-joins in a declarative rule, we introduce a relation *multi_precedes* for activity types.

relation	semantics
[activity type] *multi-precedes* [activity type]	is: activity type precedes the next activity type and executions must be synchronized. A tuple (AM,AN) means that only if the activity of type AM and certain others too have completed, may a corresponding activity of type AN also be completed.

The synchronization rule is that no compulsory precedent has not completed. In first-order logic:

for any i∈[identifier] and A_Y∈[activity type], we have: if i *completed* A_Y, then it holds that never an activity type A_X *multi_precedes* A_Y and the identifier i has not *completed* activity type A_X.

In Relation Algebra, such a double negation is known as left demonic composition operator (Backhouse, van der Woude, 1993). We can denote it as:

$$completed \subset \neg(\neg completed \circ multi\_precedes) \qquad (2)$$

4.2.3 About the *Multi-precedes* Relation

This relation looks a lot like the regular *precedes* relation. In fact, the only difference is at the join-points in a workflow. Whereas the *precedes* relation captures OR-join behavior, *multi-precedes* models AND-join behavior. As workflows can display both types of behavior, both relations are needed.

The *multi-precedes* relation constitutes a partitioning of activity types. Most partitions are uninteresting, consisting of just a single activity type. Just a few partitions, that correspond to the AND-joins, contain more than one activity type, meaning that those must synchronize: the workflow may only continue once all of them have completed.

4.3 Selective OR-Splits and Disabling

4.3.1 Selections in Workflow

Selective flow, also known as conditional branching, means that one activity precedes two (or more) activities that are placed in parallel but not all of these succeeding activities are allowed to complete.

Select in a workflow diagram is depicted by a so-called XOR-split: two (or more) arcs go out from a single point. Also, a business condition that determines which arc should be enacted (or not) is often indicated, but as we are concerned only with transforming the imperative flow into declarative rules, we abstract from such business knowledge.

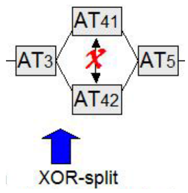


Figure 4: Mutual exclusion of activity types.

4.3.2 Rule for Selective / Disabling Activities

Consider the XOR-split after activity type A_3 which is the precedent for both A_{41} and A_{42}, but only one of them is allowed to complete (figure 4). The first consideration is normal sequencing, as has been dealt with before. We cover the new restriction of mutual exclusion by way of a new relation *disables* on activity types:

relation	semantics
[activity type] *disables* [activity type]	is: the disabling relation among activity types. A tuple (AX,AY) in this relation means that never if an activity of type AX has completed, may an activity of type AY be completed by the same identifier.

The rule for selective flow can now be formulated in first-order logic:

for any i∈[identifier] and A_X,A_Y∈[activity type] we have: if i *completed* A_Y then it is never true that i *completed* A_X and A_X *disables* A_Y.

Denoted as a Relation Algebra assertion it reads:

$$completed \subset \neg (completed \circ disables) \qquad (3)$$

4.3.3 About the *Disables* Relation

Most activities are not involved in disablings, and therefore the *disables* relation is neither total nor surjective. One activity type may disable, or be disabled by several others, hence the relation is neither univalent nor injective. Evidently, the homogeneous *disables* relation is irreflexive, while nothing can be said about its being transitive or not.

The important point however is that, in our context, the *disables* relation is symmetric by nature. A tuple (A_X,A_Y) in the *disables* relation means that never if an activity of type A_X has completed, may a corresponding activity of type A_Y also be completed. The reverse is then automatic: if an activity of type A_Y is recorded as completed, then no corresponding activity of type A_X may be completed.

Imperative workflows also know a *disabling* feature, which however is not symmetric by nature. A workflow may model that activity type A_3 is disabling for activity type A_2, meaning that completion of A_2 is allowed prior to, but not after A_3. This allows a sequence of activities a1-a2-a3, but not a sequence like a1-a3-a2. The exact sequencing is determined by the actual timestamps of completion, an attribute that we have explicitly omitted from our declarative rules.

4.4 Forward-flow Rule

The three basic patterns of workflow analyzed so far all ensure a forward flow, in contrast to the flow that we will be analyzing in the next section. The three Relation Algebra assertions (1), (2) and (3) acquired so far, easily combine into a single assertion:

rule forward_flow **as**
completed **must imply**
((*completed* ○ *precedes*)
∪ ¬(¬*completed* ○ *multi_precedes*))
\ (*completed* ○ *disables*)

For later reference, we will refer to the righthand side of this assertion as the *forward-flow* relation:

relation	semantics
[identifier] *forward-flow* [activity type]	is: the relation with a tuple (i,A) indicating that at least one (regular) precedent or all of its multi-precedents are completed, and none of its disabling activities has completed.

Using this relation, a forward flow rule may be stated: 'if identifier i *completed* activity type A, then tuple (i,A) must be in relation *forward-flow*'.

The rule has immediate enforcement: completion is always prohibited if the tuple is absent from the *forward-flow* relation. But, as the naming suggests, the rule holds for forward flows only, and does not apply for loops or 'backward' flows.

5 TRANSFORMING ITERATIVE FLOW

The previous section transformed forward flows. this section, we deal with the transformation of iterative loops, which slightly more complicated.

5.1 Iterative Looping in Workflows

Handling a workflow case may sometimes involve the repeated execution of a series of activity types until some condition is met. But the forward-flow rule described above cannot deal with a flow that loops back onto itself, so we must adjust the rule. A peculiarity is that binary relations may record a tuple once, but not several times over. Hence, *completed*, as a binary relation, cannot record iterations as required. Our solution is to employ a new identifier for each iteration of the loop. By expanding the definition of the identifier concept in this way, our forward-flow condition remains valid.

5.1.1 Relations for Modeling Iterations

We capture iterations by imagining the execution of the workflow-case to pause at the looping activity where it may fire zero, one or more iterations, as depicted in figure 5. Execution of the looping activity by the the workflow-case can be thought of as being suspended, and only when all iteration(s) have been dealt with, can it complete the looping activity, and proceed in the normal way.

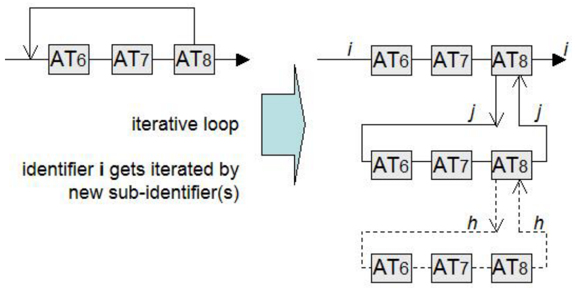


Figure 5: Iterating by way of sub-identifiers.

But this image cannot be taken literally, as our approach has abstracted from duration of an activity and we record only its completion. Instead, we just prohibit the activity to complete if iterations are running. This is somewhat different from the standard understanding of iterations, where a loop starts after completion of the looping activity, and not during its execution. As far as we could assess, this detail has no consequences for validity of the transformation.

Several new relations are needed to help us capture iteration loops as declarative rules.

First, relation *loops_to* records the iteration loop as drawn in the workflow diagram. To keep things simple, we will assume that this relation on activity types is univalent and injective. Moreover, the target activity type (where it loops to) is assumed to be a precedent for the source activity type. This is to ensure that a case looping back, will eventually return to the activity type that fired it.

relation	semantics
[activity type] *loops_to* [activity type]	is: the iterative-loop relation of activity types. A tuple (AN,AK) in this relation means that an activity of type AN, before its completion, may invoke no, one or more iterations of the loop starting from AK.

As pointed out before, a binary relation such as *completed* cannot record that the same activity type is repeatedly completed by a single identifier. We solve this by assigning a new identifier for every loop iteration, in the same way as used in audit trailing tools, activity logging and process mining (van Dongen, van der Aalst, 2005). To track which iterative loops are running for what workflow cases, we coin two more relations, *fired_from* and *iterates*:

relation	semantics
[identifier] *fired_from* [activity type]	is: which subordinate case identifier has started from the activity type. A tuple (j,AM) in this relation means that identifier j is fired from the activity type AM.
[identifier] *iterates* [identifier]	is: the iteration of identifiers. A tuple (j,i) in this relation means that the identifier j is fired for the identifier i, which is deemed necessary as an activity executes for the workflow case associated with identifier i. The identifier that *iterates* another is referred to as subordinate or sub; the other identifier is called the main case.

The *iterates* relation actually is a function. Remark that a sub may again fire its own sub-subordinates, and a stack of arbitrary depth can be created. Moreover, the *iterates* relation allows to fire several subordinates at once, for instance if a complex scheduling problem is broken down into several smaller scheduling alternatives, to be analyzed in parallel. Still, an identifier cannot iterate itself, so the *iterates* relation and its transitive closure must be irreflexive.

The *fired_from* relation, which is also a function, prevents possible confusion about which identifier originates where, in case a workflow contains more than one loop. For the sake of consistency, activity types recorded in the *fired_from* relation must also be present in the *loops_to* relation, but here again, we take this quality issue for granted.

Firing-from an activity type can occur when the case at hand has not completed the activity yet, and some business worker or condition determines that iteration is required. In this paper, we abstract from the specific conditions or business knowledge that controls invocation of the iterative loop.

5.1.2 Rules for Iteration

To model iterations in imperative workflow, three restrictions regarding (the identifier of) the iteration must be considered:

- allow the completion of its first activity,
- ensure completion of subsequent forward-flow activities, and
- constrain the sub-identifier to stop at precisely the activity type where it was fired-from, and no further activities are to be completed.

Regarding the main case, we must ensure that:

- the main case identifier must wait for all of its iterations to complete, prior to continuing.

First, remark that the activity type where an iteration begins, is rarely marked as a trigger. Hence, the forward-flow condition would normally prohibit that the activity type is *completed* by this sub, as it lacks its proper precedent, and spontaneous generation of new cases is explicitly forbidden in the imperative view. The solution of course is that another identifier acts as a substitute for predecessor, viz. the main case that is firing this iteration. The proper condition for the starting activity of the subordinate case is captured in first-order logic as:

> for any j∈[sub] and A_K∈[activity type] we have: if j *completed* A_K, and A_M *loops_to* A_K, and j *iterates* the identifier i, then the tuple (i,A_M) is in the forward-flow relation.

Rephrased in Relation Algebra, it reads:

$$\begin{gathered} completed \subset \\ iterates \circ forward\_flow \circ loops\_to \end{gathered} \quad (4)$$

In rephrasing the first-order logic, we used that both relations, *iterates* and *loops_to*, are functions. Also remember that formula (4) applies only for the initial activity to be completed by the subordinate identifier.

Once an iteration has *completed* its initial activity, it has to go forward through the entire loop, up to the activity type where it was fired. This is already described by the forward-flow condition, and no additional rules are needed.

Third, we must ensure that the iteration termina tes at its point of origin, where it is fired from. It may never go beyond that point and complete some activity further down the flow. In particular, the firing activity type never *precedes* an activity type that is being *completed* by a subordinate identifier. In first-order logic:

> for any $j \in$[sub] and $A_P \in$[activity type], we have: if j *completed* A_P, then it cannot be that j is *fired_from* some activity type A_M preceding A_P.

Rephrased in Relation Algebra, it reads:

$$completed \subset -(fired\_from \circ precedes) \qquad (5)$$

Finally, we need to consider the main case. That main case must wait at exactly the activity type where it fired subordinate identifier(s). Which is to say that this main case might have *completed* this activity under normal circumstances, but if some iteration(s) are running then must wait for them to complete. Otherwise, a running iteration becomes orphaned, executing activities to no avail. Thus, at the looping activity may be *completed* by the main case only if all of the iterations that it fired from there, have all run their course to completion. This condition to wait for iterations can be formulated in first-order logic:

> for any $i \in$[identifier] and $A_M \in$[activity type] we have: if i *completed* A_M, then it is never true that some sub exists that *iterates* this identifier i, and that sub was actually *fired_from* A_M (remind that a workflow may contain other loops), while it has not yet *completed* the activity type A_M (which is to say: that sub is still running).

Using double negation again, we can write this as a Relation Algebra assertion:

$$completed \subset \neg(iterates\sim ; (fired\_from \cap \neg completed)) \qquad (6)$$

The assertion is trivially satisfied if no iterations are fired (Backhouse and van der Woude 1993). The assertion is also satisfied if iterations for a case exist, but were fired from other activity types in the workflow than the one about to be completed by the case. Notice how assertion (6) applies recursively, i.e. nesting is allowed. If a subordinate identifier fires sub-subordinates of its own, it too will wait for its own sub-subordinates before completing.

5.2 Imperative Workflow Rule

5.2.1 Forward-Flow and Subordinates

Conditions (4) and (5) determine a scope for a sub ordinate identifier. They govern the inception and termination of each subordinate, i.e. the activity type where it starts, and where it terminates. Evidently, these two activity types coincide exactly with one corresponding tuple in the *loops_to* relation. Conditions (4) and (5) plus the forward-flow condition describe behavior of the subordinate, which is expressed in a Relation Algebra assertion as follows:

rule subordinate_workflow **as**
completed **must imply**
(*forward_flow*
∪ *iterates* ○ *forward_flow* ○ *loops_to*)
/ (*fired_from* ○ *precedes*)

Notice how for main workflow cases this subordinate_workflow rule coincides with the regular forward-flow behavior, except at activity types where looping may occur. Hence, we only need to merge condition (6) that controls behavior at looping activities into the rule above. The declarative business rule for imperative workflow becomes:

rule imperative_workflow **as**
completed **must imply**
¬ (*iterates~* ; (*fired_from* ∩ ¬*completed*))
∩ ((*forward_flow*
∪ *iterates* ○ *forward_flow* ○ *loops_to*)
/ (*fired_from* ○ *precedes*))

Remarkably, this rule, although we produced it in accordance to the imperative view of workflow, provides us with an indicative view. The righthand side of the rule assertion indicates for a case identifier i which activity types either have completed, or are allowed to complete, always in full compliance to the imperative workflow constraints. It is fairly easy to deduce from this rule an *is_enabled* relation that, for a given identifier, will determine exactly which activity types are allowed to complete, but have not *completed* yet.

6 CONTROL PRINCIPLE

The previous sections detailed how to capture the various aspects of imperative workflow. This section outlines how the workflow process is driven by way of the Control Principle.

6.1 Completing the Flow

A general assumption is that a workflow case, to fulfil the intended business goal, will always and automatically run from start to finish. A recorded trigger will always progress to its final tasks, or terminal activities. Likewise, we assumed that any subordinate iteration fired from some looping activity type will always return to its point of origin.

The indicative view of workflow states what should come after by way of enabling activities. By assumption, every activity that is enabled, ought to complete in due course, unless its completion is no longer desired or disabled (McNeile, Roubtsova, 2008). By another assumption, the terminating activity of the workflow will be enabled and completed eventually, and so the process goal is achieved.

In the imperative view of workflow, the goal of the process is not achieved as a matter of course. If a process halts in mid-term, nothing goes wrong. No rule is violated, there is no signal that there is work to do, or that a deadline has expired. The imperative workflow rule dictates what must come before, but not what ought to come after. Nothing controls that a case shall be handled start to finish. To remedy this situation, a new rule called the Control Principle is formulated.

6.2 About the *Progresses_to* Relation

Whereas the imperative workflow rule has immediate enforcement (it may never be violated), the Control Principle does allow deferred enforcement: violations are allowed but only temporarily so. Every violation should be remedied sooner or later, and work should proceed until there are no more violations. The Control Principle reasons that every workflow trigger should always progress to all of its terminal activities:

> for any $i \in$[identifier], and $A_0, A_Z \in$[activity type]: if i *completed* A_0, with A_0 trigger and A_Z terminating activity type for the workflow, then i must (eventually) also have *completed* all such A_Z.

We coin a relation *progresses_to* from trigger to terminating activity types. In fact, this relation applies to not just one, but to all workflows that an engineer may consider. We define it as follows:

relation	semantics
[activity type] *progresses_to* [activity type]	is: the relation that describes the overall start-to-finish structure of workflows. A tuple (A0,AZ) in this relation means that A0 is a triggering activity type and AZ is a corresponding terminating activity type in the workflow.

The Control Principle in Relation Algebra reads:

rule control_principle **as**

> *completed* ○ *progresses_to* **must imply** *completed*

Under this rule, work continues as long as any one the final tasks has not yet been *completed*. All outcomes must always be produced eventually; the Control Principle does not allow to disregard or skip some of the final tasks.

Inspecting the derived relation *is_enabled*, it will be clear which activities may be *completed* in compliance to the imperative workflow rule. Thus, work-to-do can be allocated to available actors, human or machine. In due course, activities are recorded as *completed*, and the rules can once again be inspected to determine violations and appropriate actions.

6.3 Workflow Execution

The Control Principle and the imperative workflow rule act independently and in harmony to realize the behavior as described in WMC'99 report.

Separate, each rule is valuable as a means to understand and interpret workflow.

The imperative workflow rule dictates that work must always be done in compliance to the workflow, and specifies in exact detail how. Violation of this rule is never tolerated. Applying this rule in a business environment without the Control Principle means that the work will certainly be performed in the correct order, but there is no guarantee that the process goals will be realized.

The Control Principle dictates that every workflow trigger should always progress to its terminal activity or activities. Violation of this rule is permitted, and it means that there is work to do. Applying this rule in a business environment without the imperative workflow rule means that business workers know that there is work to do, but there is no guidance as to the correct order of their activities.

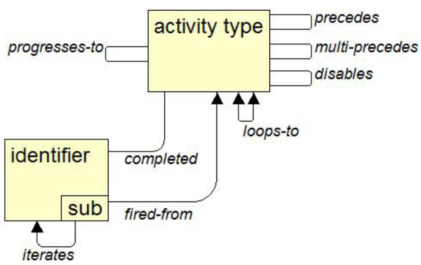


Figure 6: Elaborated structure of the declarative business rules model.

The real benefit of our approach emerges when the two rules are combined. Together, the rules ensure that the process activities execute in a well-co-ordinated fashion, exactly as pictured by the workflow diagram. In the end, all violations are resolved, all terminating activities have completed, and the process goal is achieved.

Figure 6 is a diagram of the concepts and relations used to formulate our declarative business rules for imperative workflow. An important observation is that this declarative structure is not restricted to a single workflow only. Other workflow models are readily captured in the same declarative structure, merely by populating the various binary relations with the structural knowledge encapsulated in the workflow models.

More details about the model, scripted in the Ampersand toolset, will be available at the site http://wiki.tarski.nl/index.php/Research_hub. There, a realistic example will be available to show and explore the benefits and issues of our approach. It lists the binary relations, the exact formulas, and the violations of the declarative business rules of a workflow model fashioned like figure 1.

7 DISCUSSION

7.1 Advanced Flow Structures

This paper covered only the principal structures as seen in workflows. We are convinced that other, more advanced structures can also be transformed. How, requires further analysis.

For instance, an 'empty' activity type may show up in a workflow diagram. As we abstracted from the actual work executed by an activity, the empty activity type is treated like any other: it has precedents, it may be the precedent of other activity types, it turns up in the *completed* relation etc.

In practice other exceptions exist that operational workflows must deal with, such as lack of resources, user-initiated aborts, and crosscutting events (e.g. a client dies, or an order is cancelled). Likewise, quality problems may arise in workflows, such as deadlock, irregular termination, or loops that never terminate. Merely transforming to a declarative rules model cannot be expected to solve the quality problems. This area of research is beyond the scope of this paper.

7.2 Limitations of Our Approach

The sequencing of activities in a workflow is the outcome not only of business requirements, but also of design decisions and implementation choices. Another designer may propose a different sequence that also complies with the essential business rules. Hence, precedence analysis is required to bring out what aspects of the flow is due to design choices, and which are based on actual business needs. To some extend, is a matter of opinion whether a workflow constitutes legitimate business rules, or whether it is just a way to implement underlying, more fundamental business rules (Hofstede, Aalst, et al., 2003). It can also be debated with users which flow features must have immediate enforcement, and which ones may be relaxed to allow temporary violations.

Moreover, flow rules such as precedence, disabling and the like, are just one of the many kinds of business rules. Business rules in general support not only the consecutive steps of process flows but also the rules to assess the business conditions and facts as activities are being executed. For instance, a workflow diagram often specifies the business rules that determine whether iteration is required, or which branch in an XOR-split to follow, but our relations cannot records such business conditions.

In our approach, iterative loops are dealt with in a slightly different way. The usual interpretation is that an iteration branches off immediately after completing a looping activity. Our interpretation is that iterations are recorded by way of subordinate identifiers, prior to completion of the looping activity by the main case identifier.

Furthermore, our approach was found to be limited in dealing with a disable, when there is time-dependence involved. An example is a workflow model with a constraint that 'an activity of type C3 may complete, but not before an activity of type C2'. Such a non-coexistence rule would allow to record the sequence of activities c1-c2-c3-c4, but disallow the sequence c1-c3-c2-c4. Like all of the declarative rules, our disabling rule is time-independent, and therefore must be symmetric. It cannot distinguish between the allowed sequence, c1-c2-c3-c4, and the forbidden sequence, c1-c3-c2-c4. Hence, transforming this into declarative format is not possible. A work-around may be to adjust the original workflow model to capture the precedences in another way.

7.3 Lack of Temporal Features

Our approach is founded on Relation Algebra, which does not provide temporal capabilities. Therefore, a main limitation of our approach is the lack of time in all of our formulas. This is not a drawback, instead we regard it as a major advantage. Indeed, we demonstrated how main components of workflow can well be captured without referring to time.

Still, deadlines or deadline expiry are not handled in our approach. Having abstracted from time altogether, we do not record whether an activity has begun, nor the time when its execution started. Additional binary relation such as Identifier *started* Activity_Type might be added, but it will require extra constraints, such as: *completed* must imply *started*. Furthermore, Relation Algebra provides no clock mechanism that allows to inspect which activities have started but did not complete within the allotted time. In all, we think that this paper is not the place to investigate these aspects and how to deal with them within the context of Relation Algebra. Finally, it must be pointed out that time is also not a native feature of Petrinets, the formal foundation of most workflow models.

7.4 Transforming Rules to Workflow

We conducted one-way analysis: from implemented workflow structure to more abstract business rules. The workflow was transformed without information loss, and reverse transformation will not prove to be hard. However, this is not true in general. Once the users edit, improve and rephrase the abstracted business rules, there is no guarantee that reverse transformation is easy, or that it produces a compatible flow structure. Engineering a given set of abstract business rules into a corresponding workflow model involves implementation choices, and design skills.

8 CONCLUSIONS AND OUTLOOK

The research in this paper covers the main structural components of imperative workflows. We outlined how to transform a workflow into just two rules.

The first one, called imperative-workflow rule, captures the structure of the imperative workflow, and it allows no violations at any time (immediate enforcement). This comprehensive rule comprises two parts. The easy part is called the forward-flow rule, and it adequately captures normal and parallel sequence, multiple precedence, and exclusions (selection). The more complicated part captures the iterative loops.

The second rule is called the Control Principle, which drives the workflow through to its end. This rule does allow violations, but while violations exist, there is work to do resolving them (deferred enforcement). Case handling is finished when there are no more violations, and the goal of the business process is reached.

The 8 binary relations and 2 rules that we describe can be characterized as follows:

- they capture all the knowledge about the workflow (sequencing, precedence, etc.),
- they are declarative, not procedural in character, and involve only (persistent) states, not the volatile events or transitions,
- the notion of time is not needed, the rules and relations are time-invariant, and do not refer to 'before' or 'after'
- the imperative-workflow rule and the Control Principle apply independently and simultaneously, there is no priority among the two.

We conclude that imperative workflows can be transformed into declarative business rules following the format of binary Relation Algebra. We demonstrated in detail how to do this for each of the four basic flow structures.

To accomplish the transformation of imperative workflows to declarative rules, two concepts suffice. One is the identifier concept, representing the workflow cases to be handled, and possibly the subordinates when cases trace iterative loops in the workflow. The other concept is activity type, representing the 'blobs' of workflows. The various types of 'arc' in workflow models are captured in a number of binary relations, the majority being homogeneous relations on the activity type concept.

We have shown how the procedures of workflow may be mapped into declarative business rules. This constitutes tangible evidence that the way of doing business may indeed be captured in a business rules model that meets all the demands of the Business Rules Manifesto. On the other hand, the business rules and relations that we describe are basically procedural in character, while the Business Rules Manifesto encourages to capture the business rules in a non-procedural format.

Future work is to augment our two declarative workflow rules with content-aware rules, such as the criteria for iterations and OR-splits, and also the implicit decision rules in activities encapsulated in automated services or applied by knowledge workers.

The Control Principle in its current formulation requires that all final tasks must eventually be *completed*. In practice, a workflow process may finish even if not all outcomes have been produced. For instance, when a customer order is rejected, then the workflow produces only a rejection message, and not the intended order delivery. The Control Principle should be adapted and improved to cover such practical circumstances. Furthermore, the connections between the *progresses_to* relation of the Control Principle, and the various relations that capture the details of the imperative-workflow rule, need to be analyzed and clarified.

As a result, we envision a ruleset that is consistent and comprehensive, that reflects the processing needs of the business, but without the restrictions of workflow models. A next step is to check with business users how the workflow precedences and the like, now captured in binary tables and declarative rules, correspond to the requirements of the business.

We expect that declarative rules, developed along these ideas, will capture the business requirements about the processing of incoming work better than rigid rules of imperative workflows do. The ruleset will provide an essential basis for improved models to coordinate business processes. Indeed, the reverse exercise, to derive an imperative workflow compliant to even a small set of declarative rules, may not be as straightforward, as may be illustrated by your next Sudoku puzzle.

REFERENCES

Aalst W van der, Hofstede A ter, et al., 2003. *Workflow Patterns*. In: Distributed and Parallel Databases 14(1) p.5-51.

Ampersand. At: http://wiki.tarski.nl

Backhouse R, van der Woude J, 1993. *Demonic operators and monotype factors*. In: Mathematical Structures in Computer Science 3(4) p.417-433.

Business Rules Manifesto, 2003. At: www.business rulesgroup.org/brmanifesto.htm. Version 2.0. Edited R.G. Ross. Last accessed 24 march 2013.

van Dongen B, Aalst W van der, 2005. *A meta model for process mining data*. CAiSE Conference Proceedings.

Joosten S, 2007. *Deriving Functional Specification from Business Requirements with Ampersand*. Available at http://icommas.ou.nl/wikiowi/images/e/e0/

Fahland D, Mendling J, et al., 2009. *Declarative versus Imperative Process Modeling Languages: The Issue of Maintainability*. In *ER BPM.*

Hofstede A ter, Aalst W van der, et al., 2003. *Business Process Management: A Survey.* In: Business Process Management. M. Weske, Springer Notes 2678.

Maddux R, 2006. *Relation Algebras. Studies in Logic and the Foundations of Mathematics, vol. 150.* Elsevier Science.

Maggi FM, Westergaard M, et al., 2011. *Runtime Verification of LTL-Based Declarative Process Models*. In: *RV 2011*, Khurshid and Sen (editors) LNCS 7186, p.131–146.

McKemmish S, Acland G, et al., 2006. *Describing records in context in the continuum: the Australian Recordkeeping Metadata Schema* Archivaria 1(48).

McNeile A, Simons N, 2006. *Protocol modelling: A modelling approach that supports reusable behaviour abstractions.* In: Software and Systems Modeling 5(1) p.91-107.

McNeile A, Roubtsova E, 2008. CSP parallel composition of aspect models. In: *Proceedings of the 2008 AOSD workshop on Aspect-oriented modeling.* Brussels, Belgium, ACM p.13-18.

Russell N, Hofstede A ter, et al., 2006. *Workflow Control-Flow Patterns: A Revised View*. At: www.workflowpatterns.com/patterns

Russell N, Aalst W van der, et al., 2005. *Workflow Resource Patterns: Identification, representation and tool support*. Advanced Information Systems Engineering, Springer.

Wedemeijer L, 2012. *A comparison of two business rules engineering approaches.* In: BMSD 2012, p.113-121

Witt G, 2012. *Writing Effective Business Rules*. Morgan Kaufmann. ISBN 978012-385051-5.

Workflow Management Coalition, 1999. *Terminology & Glossary*. Tech.Report WFMC-TC-1011 issue 3.0.

Cognition Capabilities and the Capability-affordance Model

Vaughan Michell
Informatics Research Centre, Henley Business School, University of Reading, Whiteknights, Reading, RG6 6UD, U.K.
v.a.michell@ henley.reading.ac.uk

Keywords: Cognitive Informatics, Distributed Cognition Theory, Capability-affordance Model, Affordance Mechanism, Affordance Path.

Abstract: Much research has been done on physical and cognitive affordances in designed objects, but little has been done on human cognitive capabilities. This paper applies the capability-affordance model to cognitive agent capabilities and affordances. It develops a cognition-affordance model by identifying cognition resources using the SRK model and cognitive task analysis. It proposes four cognition mechanisms and suggests cognitive capability depends on cognitive mechanisms interacting with knowledge. Affordance possibilities depend on different knowledge paths where existing or new agent knowledge is applied/grown by copying or mutation. Mutation may use existing logic creating new knowledge directly applicable to the real world, or, new theoretical knowledge affordances of imagination. We propose a two axis model to link cognitive affordance and imagination. We propose how perceived and cognitive affordances relate to the perception-action axis and that epistemic-axiological axis relates to the theoretical models of thought to account for creativity in human-agent cognition.

1 INTRODUCTION

Our work focuses on the capability of the object in the environment in terms of people and natural and human designed objects/systems ie 'what could the object do?' and 'how do we measure what it could do? The term 'cognitive affordance' defined by Norman et al., (1988) is widely used to explain combinations of object uses in relation to what is perceived in the environment. Cognitive affordance theory is often used to design human computer interfaces (Hartson, 2003) and in ecological design of systems. However, firstly it does not explain the interactions of internal human resources that provide the cognitive capability (Norman, 1988). Perceived affordance depends on representations in long term memory and the way the agent brain processes these. Secondly Norman's approach can confuse the perception of how objects and features of objects in the real world could be used by the agent, with our focus, the way knowledge can be used in the brain in terms of the affordance of reasoning and creativity to produce new ideas and thoughts (Albrechtsen, 2001). We therefore use the term cognition affordance to explain the possible interactions of mental resources that produce creative thought possibilities.

1.1 The Capability-affordance Model

Our previous papers introduced the idea of the capability resource model (Michell, 2011) and capability-affordance model using the work of Gibson and Norman. We reasoned that the capability of two interacting resources Ri was dependent on an affordance mechanism AM and the possible 4d space-time path AP to execute the affordance (Michell, 2013).

$$\text{Capability} = f(AM(Ri) \times AP(Ri)) \text{ and } R = f\{Aij\}$$

This was used to explain physical resource combinations or directly perceivable affordances acting in 3 dimensions plus time (Barentsen and Trettvik, 2002). However, we paid little attention to the way knowledge can be used in the brain to explain cognition affordance. We used Stamper et al. (2000) to differentiate between two types of behaviour substantive (or physical action) eg a doctor injecting a patient and semiological action of the doctor making sense of signs using his knowledge and cognition to diagnose the patient.

1.2 Definitions

Further details and explanations can be found in (Michell, 2013).

Table 1: Definitions.

Term	Definition
Ri	A resource object in an environment
Aij	Affordance between resources I, j
C	Capability - the potential transformation of a system of resources = f(AMxAP)
AM	Affordance mechanism - the natural science of transformation
AP	Affordance path - the space-time interaction
Agent	Human/intelligent machine able to transform resources

1.3 Objectives

In this paper we explore how the capability-affordance model can be used to describe cognition and semiological actions with five objectives:

a) To set out the reasoning for semiological affordances and their relationship to cognitive affordances
b) To develop a model of semiological affordances and their relationship to existing work
c) To identify semiological mechanisms
d) To identify semiological paths
e) To account for creativity.

Section 2 explores the definition of cognitive affordance and its role in semiological resource interaction and defines cognitive capability as an interaction of agent cognitive and knowledge resources. Section 3 reviews cognitive resources using human cognitive behaviour. Section 4 reviews agent knowledge resources based on cognitive architecture research and identifies cognitive mechanisms based on semiotics models. Section 5 proposes an integrated cognition model. Section 6 discusses the implications of the model and Section 7 and 8 concludes and identifies future work.

2 SEMIOLOGICAL CAPABILITY

As mentioned before, this section considers cognitive affordance and its role in semiological resource interaction, and defines cognitive capability.

2.1 Cognitive vs Semiological

An action (a) is a transformation of resources (Michell, 2011). A semiological action uses signs perceived by the agent from the environment to process possible actions (Stamper et al., 2000) Semiological action depends on information and knowledge from sensors and cognitive actions in the mind of the agent. The interaction between the cognitive mechanisms of the brain and knowledge create possible ideas for action – cognition capabilities Ccog. Using the capability affordance model the cognitive affordance mechanism relates to the interaction of cognitive resources (Boy, 1998), one of which must be an agent's mind and its cognitive mechanisms Rcog. The second represents tangible or intangible resources ie data, information and Knowledge: Rk, obtained from the environment as only this is able to interact mentally in an affordance pair in the agent.

Ccog = f(agent cognitive mechanism x data, information, knowledge interaction) = f(Rcog x Rk)

To understand these affordances we need to understand theories of cognitive brain function. However, while the interaction mechanism of non-human objects is well understood, the mechanism of the brain is not.

2.2 Distributed Cognition Theory

Distributed cognition theory provides a framework for cognition spaces based on cognitive psychology (Zhang and Patel, 2006). We extend Zhang's model using a physical internal space Sp and a cognitive space Sc. Sp comprises the biochemistry and physiology resources of the body (B) ie the mechanism of biochemical reactions and the physical structures such as bone and flesh, synapses., muscle (P) that provide the path for the animal to work and select physical affordances. The cognitive space Sc comprises the mechanisms of perception and cognition (C) and the data information and knowledge (K) and relates to semiological affordances. The External space represents the environment and the natural and manmade (technology) structures (S) than provide physical and cognitive affordance possibilities as a result of agent sensor information (I) and actuators such as hands.

We now identify the agent cognition and knowledge resources and the mechanisms AMcog (C) and affordance paths APcog relating to data, information and knowledge (K) that enable semiological capability.

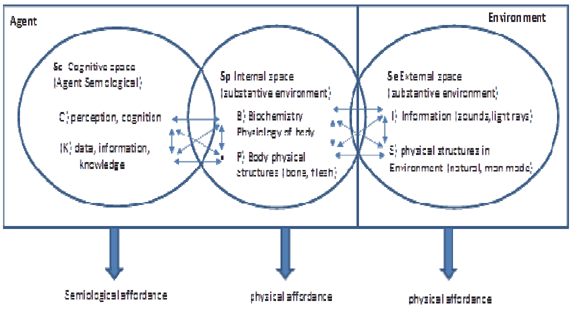


Figure 1: Affordance Spaces (adapted from Zhang).

3 AGENT COGNITIVE RESOURCES

3.1 Cognitive Behaviour Modelling

To develop an understanding of the cognition resources and their mechanisms AMcog we investigate applied psychology cognitive behaviour literature. Norman's 7 stages theory of action can be used to model cognition (Zachary et al., 1998). A worker will have specific goals G and actions to execute E to achieve them eg a surgeon examining a patient. This may involve a cascade of sub-goals eg visually examine patient talk to them, feel them. Perception P involves recognising patterns of speech, images from sensors and haptic patterns.

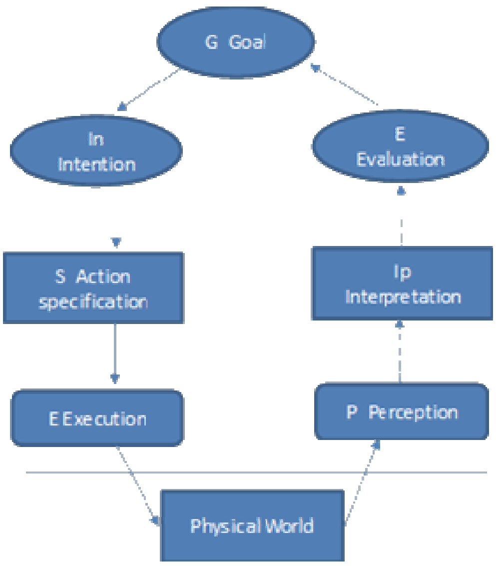


Figure 2: Norman's 7 Stages of Action (adapted).

The agent interprets the perception Ip in terms of how it relates to a planned course of action. The agent evaluates options for action E vs the goals and selects the best plan of action to achieve the goal. The user then plans his actions in his mind makes the intension In to act and specify the sensor-motor driving actions S. The agents actuators eg hands, limbs execute the action E. The result of the action is then perceived through the senses and compared to the goal and any corrective action applied. This P, Ip, E, G, In, S, E loop can be considered to be carried out at different levels of cognition.

3.2 The SRK Model

Rasmussen identified three types of cognitive behaviour in the skills, rules, knowledge in the SRK model of human decision making in high risk work-system domains (Rasmussen, 1983). Skill based behaviour corresponds to sensory motor performance during unconscious actions and unconscious control where the human sensor-motor system acts automatically based on the agents tacit knowledge of learnt tasks using learnt perceived patterns (Albrechtsen, 2001). Continuous 4d space-time data signals from agent sensors update a model of the environment in episodic memory. Skill based behaviour SBB relates to learnt sensory motor patterns based on previous experience, eg changing gear when driving etc and is termed unconscious control because of its automatic response directly from perception through the 7 stage model to sensor-motor action. Conscious control involves greater cognition using rule based behaviour RBB where recognition of signs and cues from the sensors drive an 'if-then rule' behaviour based on stored action rules related to cues perceived from the environment. This depends on perceptions of familiar patterns in a familiar environment matching the necessary cues/signs associated with the action rule conventions (Vicente and Rasmussen, 1992). Cues/signs relate either to experiences or learnt/cultural behaviour encoded as rules labelled for 'states/situation or goals and tasks' (Rasmussen, 1983). Rule based behaviour cannot generate new rules, but this occurs in highest level of behaviour, knowledge based behaviour KBB where the agent's mental model of the world enables the formulation of new rules, goals and strategies and predictions of the response of the environment. KBB can use both sensor and history information to construct the conceptual mental model (Albrechtsen, 2001). Rasmussen's SRK model identifies the three mechanisms of cognitive processing as data signals (SBB), as rules or cue-rule-action mappings or symbol/concept based problem solving and analysis actions (Vicente and Rasmussen, 1992).

Table 2: Cognitive tasks and actions.

	a) Miller's cognitive tasks and resources	
Ref	Cognitive agent task	Meaning
SE	Search	look for
ID	Identify	what is it/its name
CD	Code	tranlate in meaning
CM	Compute	identify logical/maths answer
CN	Control	adjust action to meet goal
PL	Plan	Matching resources to times
CT	Categorise	define/name a thing vs group
	b) Bloom's classification knowledge actions	
Ref	Cognitive Behaviour	Detail
RE	recall knowledge	recall information about concept or rule
CO	comprehend knowledge	understand the meaning of a concept or event
AP	apply knowledge	apply the concept/rule to a specific situation
AN	analyse	separate into concepts/parts to understand their structure
SN	Synthesise	assemble parts/concepts to form a new meaning
EV	Evaluation	make judgements about values and concepts
CR	Create	produce new knowledge and concepts

3.3 Cognitive Task Analysis

Work on analysis of cognitive tasks (CTA) for human computer interfaces (Hall et al., 1995) (Norman, 1986) and ecological interface design (Wong et al., 1998) provide additional models that relate to cognition. We use the convention of underlined letters as shorthand for cognitive tasks. Miller's vocabulary of actions for mental processes, human information processing resources and their related task agents (Lee and Sanquist, 1995), (Table 1a) represents the output of human cognition resources. Bloom's analysis of learning (Anderson et al., 2005) identifies cognitive behaviours related to cognition and learning. Recalling and remembering knowledge is considered the lowest cognitive level of activity that we can relate to SBB. With the action of comprehending or understanding knowledge at a higher level, that then enables the ability to apply the knowledge as rules and actions. The analysis activity is considered a higher level activity still with evaluate and finally create (new knowledge) and problem solving as the highest level of cognitive activities in terms of complexity and abstraction in the cognitive process. This relates to KBB (Table 1b), where cognitive tasks relate to transformation actions of agent mental cognition resources.

4 AGENT KNOWLEDGE RESOURCES

4.1 Cognitive Architectures

There are three main types of knowledge. Knowing what or 'learning by using relates' to the use of systems or technology as encoded in human episodic memory and can be loosely approximated to SBB. Knowledge from 'learning by doing' 'know how' corresponds to RBB (Carud, 1997). Rules are evident in the external environment as procedures, policies, processes and as tacit codified rules in agents. KBB relates to 'know why' or knowledge gained by 'learning by studying' as well as concepts and relationships and tacit mental models created in the mind (Carud, 1997). Cognitive architecture theories; SOAR, EPIC, ACT-R, PARI) (Laird et al, 1987), can provide insight into knowledge interaction. We use the COGNET knowledge framework (Zachary et al., 1998) based on cognitive psychology and goal oriented models based on Rasmussen's and Norman's approach (Figure 3).

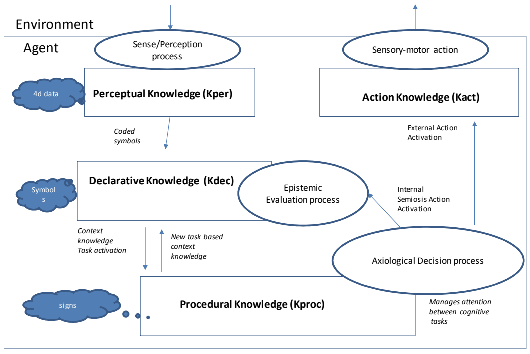


Figure 3: COGNET Knowledge Framework (adapted).

4.2 Perceptual Knowledge and Reasoning

In the COGNET model information from the agent's visual, aural haptic etc sensors is converted into 4d space time signal information. Sensory cues are perceived as visual (eg images) and auditory (eg conversation) patterns (Zachary et al, 1998). This sensory cue information in the form of signs becomes meaningful by the process of perception recognising patterns in the signals. COGNET uses the term 'sensory demons' to refer to system interface displays or patterns of natural phenomena eg a red spotted mole disease pattern. Auditory demons include speech terms and acts that have specific semantic significance. The interpretation of signs in the environment is the process of semiosis (Pierce, 1935) and a cognitive abduction process of inference to select the best semantic meaning of a sign eg for visual and aural recognition of patterns (Magnani and Bardone, 2006) as in perception related to a knowledge base of experience. In Pierce's process of semiosis, sensor signals for objects in the environment are perceived as signs and symbols representing objects and their meaning for the agent are decoded in a process of connotation (Benfell et al., 2013). The resulting perceptual knowledge Kper relates to pattern recognition models that link meaning to a visual model based on working memory and with experience from long term memory these provide 'coded symbols' that can be used by other cognitive processes.

4.3 Declarative Knowledge and Reasoning

Declarative knowledge, Kdec, includes the agent's mental conceptual model of the world based on the concepts identified through symbols and semantics mentioned earlier (Clark and Feldon, 2006) ie 'knowledge as a conceptual structural model' (Albrechtsen, 2001). It comprises abstract construct symbols which unlike rules cannot be reduced to signs (Rasmussen, 1983). It focuses on what and why and is based on propositions, facts and is hierarchically structured and uses episodic memory to record environment events and model them (Clark and Feldon 2006). The knowledge ranges from the real world ie 2D/3D models of the physical environment to logical relations and logic rules, facts, beliefs and solution strategies and cases to behavioural models and abstract models that have no equivalent in the real world environment. Declarative knowledge is used for problem solving and includes the history of relevant objects to the task and also plans and solution strategies to achieve a goal (Zachary et al., 1998). KBB relies on the individual tacit declarative mental model that the agent constructs that differs from agent to agent. Cognitive interpretation processes operate on the declarative model schema to make connections between symbols that enables insight and new knowledge to be developed to support problem solving. As Pirolli et al asserts Information=> schema=> insight=> solution (Pirolli and Card, 2005). The power, capability and reliability of agent KBB depends on the power, capability and reliability of the symbolic mental model and conceptual processing and related affordances. The range and complexity of declarative conceptual model covers predicates, definitions semantic relations (structural, causal, functions) and simple and complex associations as well as rules, facts and beliefs (Aamodt, 1991).

4.4 Procedural Knowledge and Reasoning

Procedural knowledge resources (Kproc) relates to models of rules, for example national language rules, job context rules eg clinical rules, mathematics rules etc. The rules are encoded with cues for when a task is relevant. Organisational semiotics models these rules in the form of norms (Stamper et al., 2000). The COGNET model suggests the goal and procedural knowledge form a cognitive task which directs the use of the knowledge through the cognitive mechanisms to execute a semiological or substantive task if the goal cue is recognised.

$$Tcog = f(\text{goal, procedural knowledge}) = f(G, Kproc)$$

Cognitive tasks are managed at a meta-cognitive level of reasoning to decide which course of action to take and when to take it. These evaluation mechanisms are axiological mechanisms (Benfell et al., 2013) for decision making and selection of strategies related to agent internal resources affording decision making and evaluation. Meta-cognitive tasks adjust the priority of these tasks if interrupted by perceived events in the external environment eg bells ringing, 4d scene changes etc.

Interpretation Mechanisms. Reasoning involves interpreting information (sign/rule processing and symbols) about an environmental situation.

This could be from sensors and processing this information in conjunction with knowledge to produce a given goal. Reasoning may be rule based as in RBB. Sign/rule processing relates to the if-then reasoning using procedural knowledge ie selection of the best rule according to sign cues from the external environment. Alternatively it may be based on models and cases in the symbolic conceptual model as in KBB. Symbolic processing relates to the processing of symbols and read, write and update of the declarative knowledge model. This knowledge based processing capability unlike rule based processing is adaptive to new environments where new knowledge and rules can be developed by knowledge based reasoning (Albrechtsen, 2001). At higher level this involves processing mechanisms for inference methods. These methods may be deductive based on logical mental models and theories or inductive (knowledge of events and instances) developed from experience, education and training

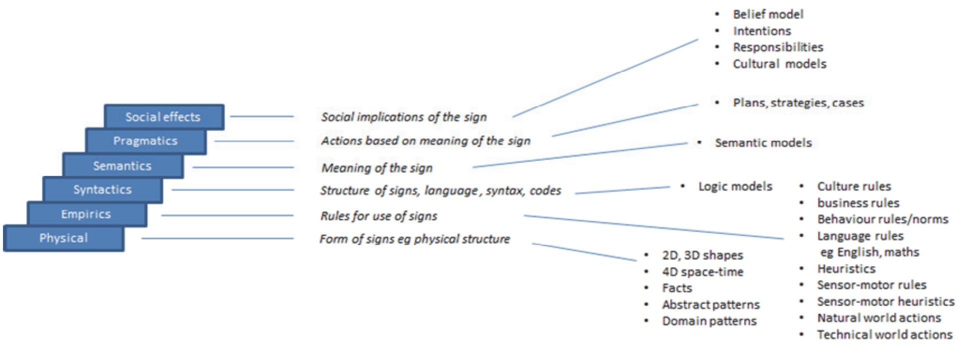


Figure 4: Semiotic knowledge hierarchy (after Stamper).

(Aamodt, 1991). Other forms of reasoning include induction and deduction based on neural connections made during learning these techniques from others. Deductive reasoning is based on logic models that may be learnt or culturally developed. Inductive reasoning involves testing against hypotheses.

4.5 Action Knowledge and Reasoning

Action relates to substantive action tasks on the technical or natural environment. For example, pressing buttons, moving objects. Alternatively they may relate to semiological actions such as communication (Zachary et al., 1998) or thinking that changes the state of the mind but has no external impact. Action knowledge relates to sensor-motor knowledge of how to drive and control the agent actuator bio-mechanics to control movement eg of hands/body. This relates to skills under automatic control in terms of hand-eye coordination driven by environmental cues for physical tasks. Action knowledge includes sensory-motor knowledge for natural objects as well as man-made technology eg hand-eye coordination for drug injection or routine mental maths calculations.

4.6 Knowledge Summary

We have seen an agent's knowledge covers a range of semiotic ladder levels. Rules and schemas may relate to the physical world. They may relate to use of language and the syntax for example sentence construction or mathematical expressions. They may relate to expected behaviour encoded from experience or business rules to cultural rules about behaviour. Rules may relate to formal models eg laws or they may be informal may be developed from experience, for example rules of thumb. Rules may relate to both physical and mental behaviours (Benfell et al., 2013). See Figure 4.

5 THE COGNITION-AFFORDANCE MODEL

Cognition affordance relates to a) the interaction of cognitive reasoning processes of the brain with knowledge to model potential actions and strategies for the real or imaginary agent world and b) the selection the best course of action either physical or mental. Cognition affordance depends on semiosis of perceived signs interpreted as sensor signals from the real world and/or from the conceptual world of the imagination of the agent using these signals, signs, symbols to model the real and imaginary worlds to plan actions. Cognition capability depends on the mechanism of cognitive tasks (Ct) acting on cognitive knowledge ie Ccog = f(Ct xCknow). The Cognition Model (figure 5) based on organisational semiotics EDA model (Liu et al., 2013) identifies and integrates cognitive processes in conjunction with the models discussed earlier.

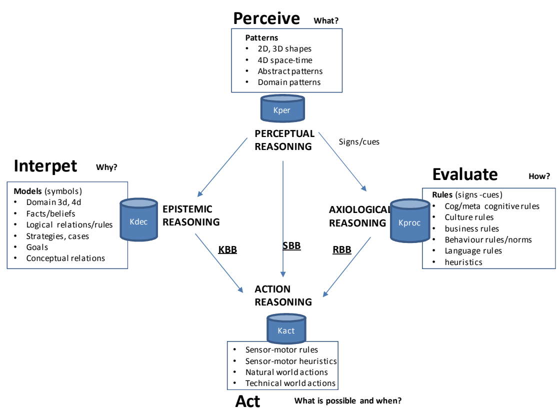


Figure 5: Cognition model.

Perception relates to pattern recognition reasoning on perception knowledge Kper. Epistemic reasoning relates to inference processes and reasoning about conceptual declarative knowledge Kdec. Axiological reasoning relates to rule based decisions about behaviour operating based on procedural knowledge Kproc. See Figure 5.

5.1 Perception Capability and Reasoning

Perceiving patterns in the environment relates to lower order cognitive tasks such as recalling RE and comprehending CO. This depends on matching perceptive knowledge ie the range of pattern databases the agent possesses or making sense of new patterns via epistemic reasoning. For example a clinician in seeking to identify a disease needs to match the cues from the patient in terms of visual/aural/haptic information after examining and talking with the patient and reading their notes. The clinician may build a perceptual model of the patient based on disease patterns, physiology patterns and models of consequences (Chapman et al., 2002). This produces a number of affordance options as possible disease models that need to be interpreted based on their plausibility relative to perception. This represents a cycle of Norman's model to meet the goal and intention of identifying the disease model for the patient.

Cper= f (perceptual reasoning tasks x Kper)

Audio perception is a function of the ability to both record and to match aural patterns sensed in the environment as a language of sounds with meaning. A language affordance eg 'can speak English' depends on the action of the cognitive processing resource recalling (RE) and matching information from the language patterns and the quality; depth, range of the agent vocabulary Klang.

Cper= f(Alang) = f(RE x Klang)

Similarly, visual perception affordances Avip are a function of the ability to both record and to match visual patterns sensed in the environment to meanings. For example visual perception affords recognition of disease patterns by recalling (RE) and comprehending (CO) disease patterns and cues that best match the sensed disease pattern which depends on the cognitive action of recall and its interaction with disease knowledge.

5.2 Interpretation Capability Cepi

Cepi - Epistemic reasoning involves higher cognitive tasks such as analysis AN, synthesis SN, problem solving PS and creativity CR as useful strategies in unfamiliar situations. Reasoning strategies such as induction, abduction and deduction may be used. The use of epistemic reasoning in medicine is often referred to as hypothetico-deductive reasoning (Chapman et al., 2002) and involves establishing a hypothesis for the problem illness, gathering data to support or refute the hypothesis followed be evaluation to establish the best causal reasoning (ie know why') for the symptoms. This requires conceptual knowledge models of illness, disease functionalities, mathematics etc. Affordance options relate to the different problem- solution models and their plausibility vs goal/evidence ie Cepi = f (prob solving reasoning x Kdec). The capability of epistemic reasoning as in expertise, is complex and in any individual will vary with the ability to reason and conceptually model the world and the depth, specificity and form of the knowledge the agent is able to develop (Aamodt, 1991).

5.3 Evaluation Capability Caxi

Caxi refers to: Axiological reasoning.

It relates to cognitive tasks using RBB and decision processes to select the best rule given environmental or mental cues. Affordance options here relate to the permutations of the possible meta-cognitive actions and their sequences and the cognitive task based are different rule models and their plausibility vs cues and the action goal.

Caxi = f (rule reasoning x Kproc)

Cognitive tasks relate to actions on procedural and declarative knowledge. Cognitive tasks include understanding and problem solving where obvious rules can't be invoked and declarative knowledge is required. This may include information processing strategies such as Miller's cognitive tasks planning PL and controlling CT (Lee and Sanquist, 1995). The rule evaluation takes place at different levels in the semiotic ladder. From the evaluation of laws and policy rules down to process and action rules. The rules act as a constraint on the possible actions. A clinician has many different rule sets to follow. At high level may be policies and WHO guidelines at the process level clinical pathways can be selected to guide possible team actions. At the action level algorithms (eg for inserting catheters) and the clinicians own heuristic rules developed from experience. The affordance options relate to different disease/illness rule models and the cognitive process involves the clinician deciding which rules to apply by assessing a series of permutations of cues and disease patterns.

5.4 Action Capabilities Cact

Action capabilities relate to automatic actions ie unconscious thought and skill based actions. This includes human-environment sensory motor skills eg grasping, human-technology skills eg using a mouse to drag and drop and human-human interactions eg shaking hands etc. A medical example might be identifying the actions and behaviours to stabilise an emergency patient (Chapman et al., 2002). Affordance options are different action models and their plausibility vs cue/stabilisation goal. Here the clinicians react instinctively to act based on experienced action knowledge of the steps to take how to behave and use equipment and human resources based on the cues for action from the patient, colleagues, technology resources and the situation ie Cact= f (action reasoning x Kact).

6 DISCUSSION

Cognition axes are proposed in this section as well as cognitive capabilities.

6.1 Cognition Axes: Real Vs Imaginary

We can say cognition capability is a tuple of these four capabilities:

Ccog = f (Cper, Cepi, Caxi, Cact)

In all the above cases the cognitive capability of the clinician will depend on experience, cognitive ability and cognitive resources. As Gibson notes: interpretation depends on the agents culture, experience and intentions (Benfell et al., 2013). The following sections discuss differences in cognitive capability. The agent cognitive affordance space Sc relates to how the mental models and reasoning process can provide the agent with alternative action possibilities. This represents the interaction of the cognitive reasoning mechanism resources with memory and knowledge. The axis of linkage between perception-action represents affordance possibilities in the real world of seeing and doing ie perceived and cognitive affordances. HCI design depends on making possibilities of using the technology as obvious as possible so they can directly be used for action. We suggest the linkage between interpretation and evaluation represents the imagination where possible concepts and possible actions can be modelled and the implications tested before action is decided. Another aspect of cognitive affordance is the possibility to imagine or model new imaginary interactions and imaginary logics and languages. Imagination suggests a conceptual environment that can model a) the real world and its features and use it to identify possible future states and secondly b) to model imaginary worlds with different rules, logic and beliefs. This capability enables great works of literary fiction (Harry Potter), art (Salvador Dali), science (relativity theory). Techniques such as brainstorming and creative methods where normal logic and beliefs are suspended can sometimes highlight new possibilities where the imaginary world highlights a new possibility or creative affordance applicable to real world logic. Perhaps the process of dreaming is nature's mechanism for trying out possible illogical affordances that would not have occurred to the conscious animal having to make sense of a real physical world!

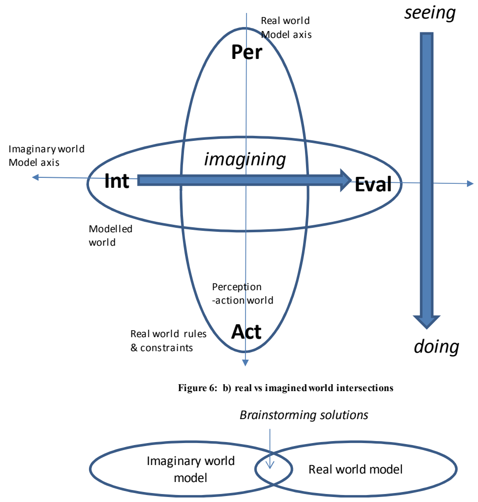


Figure 6: Cognition Axes – real vs. imaginary.

6.2 Comparing Cognitive Capabilities

Experience and practice in each of the cognitive mechanisms will be determined by the roles and work the agent carries out and how much involves real world vs imaginary world models. Some job roles involve a greater focus on seeing and doing eg a nurse, artist. Other roles focus more on imagination and conception without action as in the theoretical world eg novelist, scientist. Different roles will exercise and focus different cognitive functions. Some involve a combination eg a knee surgeon, may have good levels of perception of disease, 3d structures and have good haptic perception with much experience of manipulating joints. He may have lots of logic and problem solving experience, some experience of using clinical pathway rules, but will have less experience in policy rules and in reasoning about them compared to a Medical Director. An anaesthetist may have better abilities for reasoning about drugs based on repeated experience. In contrast an artist may be more creative than the surgeon because they focus their life on painting which involves seeing-doing action experience in terms of visual perception and painting heuristics. Their imagination is less structured and may involve creating and using imaginary rules and concepts. In contrast an architect's creativity will be more structured as it is limited by the rules of physics. Natural abilities to perceive, follow rules, to reason will also depend on the brain physiology as well as experience.

Ccog = f(Experience Ex x Reasoning Ability Ab)

6.3 Knowledge Paths

Knowledge is developed by the process of learning from experience and/or use of cognitive capabilities and mental modelling or taught or communicated by others. In each and every case both the knowledge and the cognitive capabilities is potentially growing depending on the brain physiology and individual cognitive capabilities ie intelligence of the individual. Benfell et al., (2013) mentions the link between affordance and memes in which affordances ie ideas are communicated 'by reading, watching television etc' It is this exposure to affordance examples that enables direct copying. Alternatively we may use our existing knowledge and cognitive skills to playing with and mutate ideas. We may use existing logic to extend our knowledge or, depending on our capabilities, create new logic to produce new knowledge. This happens at different levels of the semiotic ladder from the knowledge of the physical world through formal to informal abstractions such as culture knowledge with learned formal and or informal rules/heuristics. See Figure 7.

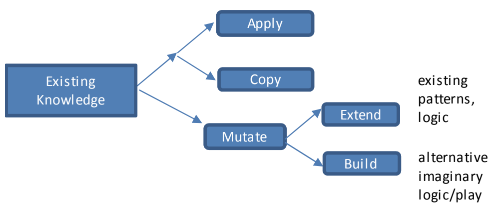


Figure 7: Knowledge paths.

7 SUMMARY AND CONCLUSIONS

This paper has shown how the capability-affordance model and others can provide an approach to explain the possible interactions of mental resources that produce creative thought possibilities and cognition affordances and meet our objectives. Section 2 explained the reasoning for semiological affordances by identifying the internal cognitive space and its relationship to cognitive affordance - objective a). To develop a model of semiological affordances (objective c) in Section 3 we identified cognition resources using the SRK model and cognitive task analysis. In section 4 we proposed agent knowledge resources based on cognition architecture and we suggested how semiotics and Peirce's model relate to cognition mechanisms. In section 5 we showed that Cognitive capability depends on cognitive functions interacting with knowledge and proposed 4 mechanisms for cognition. We proposed in section 6 that the cognitive path (objective d) depends on the knowledge paths where existing knowledge is applied and grown by copying or mutation. Mutation can occur as a result of mind games, mental playing and imagination. This mutation may use existing logic of Ccog to creative new knowledge directly applicable to the real world. or, to create new knowledge that is not directly applicable as theoretical knowledge. To account for creativity (objective e) we proposed how perceived and cognitive affordances relate to the perception-action axis and the epistemic-axiological axis relates to mental theoretical models to extend the real world model or create new imaginary worlds as in creativity.

8 FUTURE WORK

The complexity of cognition (the process of using cognitive actions and knowledge) means we have only scratched the surface. Further work is needed to identify the detailed mechanisms of cognitive affordance permutation with specific examples using cognitive task analysis. For example how does a novelist or artist think compared with a surgeon. However whilst methods such as Cognitive task analysis can provide useful insights into the processual mechanisms, the detailed models rely on developments in cognitive psychology and medical research.

REFERENCES

Aamodt, A. (1991). A knowledge-intensive, integrated approach to problem solving and sustained learning. Knowledge Engineering and Image Processing Group. University of Trondheim, 27-85.

Albrechtsen, H., Andersen, H. H. K., Bødker, S., & Pejtersen, A. M. (2001). Affordances in activity theory and cognitive systems engineering.

Anderson, L.W., Krathwohl D.R.,Bloom B.S.. (2005) A taxonomy for learning, teaching, and assessing. Longman.

Barentsen, K., and Trettvik, J. (2002). An Activity Theory Approach to Affordance, Second Nordic Conference on Human Computer Interaction: ACM, pp. 51-60.

Benfell A., Williams R., Liu K. (2013)Enriching enterprise resource planning systems for strategic advantage: A semiotic motivated approach. Proceedings of ICISO, Stockholm.

Boy, G. A. (1998). Cognitive function analysis (Vol. 2). Praeger Pub.

Carud, R. (1997). On the distinction between know-how, know-why, and know-what. Advances in strategic management, 14, 81-101.

Chapman, D. M., Char, D. M., & Aubin, C. D. (2002). Clinical decision making. Rosen1s Emergency Medicine concepts and clinical practice, 125-133.

Clark RE., Feldon D.F.(2006): "Instructional implications of cognitive task analysis as a method for improving the accuracy of experts' self-report." Avoiding simplicity, confronting complexity: Advances in studying and designing (computer-based) powerful learning environments 109-116.

Hartson, R. (2003). Cognitive, physical, sensory, and functional affordances in interaction design. Behaviour & Information Technology, 22(5), 315-338.

Hall, E. P., Gott, S. P., & Pokorny, R. A. (1995). A Procedural Guide to Cognitive Task Analysis: The PARI Methodology (No. AL/HR-TR-1995-0108). ARMSTRONG LAB BROOKS AFB TX HUMAN RESOURCES DIRECTORATE.

Laird, .J. E., Newell, A., Rosenbloom P.S..(1987) Soar: An architecture for general intelligence. Artificial intelligence 33.1 1-64.

Lee, J. D., & Sanquist, T. F. (2000). Augmenting the operator function model with cognitive operations: Assessing the cognitive demands of technological innovation in ship navigation. Systems.

Liu S., Liu K., Li W. (2013) A multi-agent system for pervasive healthcare. Proceedings of ICISO, Stockholm.

Magnani, L., & Bardone, E. (2006). Designing human interfaces. The role of abduction. Computing, Philosophy and Cognition. College Publications.

Michell, V.A.(2011) A Focused Approach to Business Capability. First International Symposium on Business Modelling and Software Design – BMSD 2011, Sofia, Bulgaria, pp. 105–113.

Michell V. (2013) The Capability Affordance Model: Comparing Medical Capabilities. In: B. Shishkov (Ed.) Business Modeling and Software Design – BMSD'12 Revised Selected Papers, Springer-Verlag – Lecture Notes in Business Information Processing, Berlin-Heidelberg.

Pierce C. S. (1931-35), Collected Papers, (6 volumes), Hartshome C. & P. Weiss (eds.), Cambridge, Mass. Harvard U.P.

Pirolli, P., & Card, S. (2005, May). The sense making process and leverage points for analyst technology as identified through cognitive task analysis. In Proceedings of International Conference on Intelligence Analysis (Vol. 5).

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. Systems, Man and Cybernetics, IEEE Transactions on, (3), 257-266.

Stamper, R., Liu, K., Hafkamp, M., Ades, Y.(2000)Understanding the Role of Signs and Norms in Organisations. Journal of Behaviour & Information Technology 19(1), 15–27.

Vicente, K. J., & Rasmussen, J. (1992). Ecological interface design: Theoretical foundations. Systems, Man and Cybernetics, IEEE Transactions on, 22(4), 589-606.

Wong, W,B,L.Sallis P.J., O'Hare. D., (1998) The Ecological Approach to interface design: Applying the Abstraction Hierarchy to intentional domains Computer Human Interaction Conference, Proceedings. Australasian. IEEE.

Zachary, W., Ryder, J. M., &Hichinbothom, J. H. (1998). Cognitive task analysis and modelling of decision making in complex environments. Making decisions under stress: Implications for individual and team training, 315-344.

Zhang, J., & Patel, V. L. (2006). Distributed cognition, representation, and affordance. Pragmatics & Cognition, 14(2), 333-341.

Modelling and Validation of KPIs

Ella Roubtsova[1] and Vaughan Michell[2]
[1]*Open University of the Netherlands, Valkensburgersweg 177, Heerlen, The Netherlands*
[2]*Henley Business School,University of Reading, Whiteknight, Reading, RG6 6UD, U.K.*
Ella.Roubtsova@ou.nl, v.a.michell@henley.reading.ac.uk

Keywords: Key Performance Indicators, Goal Modeling, Concept Modeling, Executable Process Modeling, Validation of Properties of Key Performance Indicators.

Abstract: Competition for funding between organizations attracts attention to their Key Performance Indicators (KPIs). KPIs are usually designed with a top-down approach as families of measures for a group of business units and often do not take into account the difference in goals and business processes of organizations at the strategic, tactical and operational level. This results in unreliable, inefficient and sometimes inconsistent KPIs. Validation of KPI properties is typically postponed until the KPI is implemented, and databases are populated with values. The reason is the absence of intuitive and simple methods for KPI modelling that relate the strategic and tactical models and executable operational models. We propose such a method for KPI modelling and validation of their properties. Our method combines ideas of goal, conceptual and executable process modelling. Models at all levels are derived from KPI definitions. The conceptual modelling techniques are used to relate the strategic and tactical models. The synchronous semantics of protocol modelling is used to relate the tactical and the operational models. The executable operational and tactical models enable derivation the KPI values, testing KPIs against the desired properties and identification of ambiguities in KPI definitions that need to be resolved to improve KPIs.

1 INTRODUCTION

Key Performance Indicators (KPIs) are well established measures of business performance (Parmenter, 2010). They are designed almost for any domain of our life and there are KPI standards for many sectors, including industry, medicine, education, and services (KPIStandard, 2013; Garengo et al., 2005).

Usually the KPIs are defined by strategic and governmental bodies for a group of organizations in a branch in order to make the organizations comparable. There is already an announcement of the Launch of a Committee on Standards and KPIs for Brand & Audience Campaigns in Europe (IAB Europe, 2013).

The methods of KPI design (Neely et al., 2000; Strecker et al., 2012; Frank et al., 2009; Popova and Sharpanskykh, 2010) include identification of the most critical processes in the organizations of the branch. Such generalization often leads to definitions that use professional jargon, undefined notions and forgotten elements of the business processes. There is a risk of KPIs being interpreted differently in different organizations.

(Berler et al., 2005) note that KPIs often fail to truly represent performance but instead highlight problems of performance measurement. Therefore, there is a need for a method of KPI modelling that can adequately support the KPI definitions and localise misleading KPIs.

Validation of KPI properties (listed, for example, in (Peter Kueng, 2000)) is typically postponed until the KPI is implemented, and databases are populated with values. The reason is the absence of intuitive and simple methods for KPI modelling that relate the strategic and tactical models and executable operational models. Leaving validation of KPIs for the implementation phase can result in ineffective KPIs (Berler et al., 2005).

Therefore, there is a need for a method for validation of the underlying business process used for the KPI definition and the KPI properties.

In this paper, we propose a method that addresses both needs: (1) it identifies the aspects of the business process relevant for a modelled family of KPIs in the KPI definitions; (2) it enables validation of the KPI properties on the abstract business process. These useful features of our method are influenced by the carefully chosen combination of the ideas from goal (Dardenne et al., 1993), conceptual (OMG, 2003) and protocol modelling (McNeile and Simons, 2006). We explain the ideas of those methods and show the advantages of the chosen combination for KPI modelling.

Design of KPIs is closely related to non-functional requirements engineering as KPIs can be used to judge the operation of organizations (Golfarelli, 2009). It is well known that the requirements engineering methods for non-functional requirements are far from being developed (Golfarelli, 2009). Our method contributes to methodological support of requirements engineering of non-functional requirements.

Layout of the paper: Section 2 presents desired properties of KPIs defined in literature. We explore what those properties mean for KPI modelling and formulate the requirements for a method for modelling of KPIs and validation of their properties. Section 3 describes our method of modelling and discusses validation of KPIs. The method is illustrated with analysis of two families of KPIs proposed for the programme (Improving Access to Psychological Therapies, 2013). Section 4 presents some related work and concludes the paper.

2 DESIRED PROPERTIES OF KPIs AND REQUIREMENTS FOR A METHOD FOR MODELLING OF KPIs AND VALIDATION OF THEIR PROPERTIES

Kechenham and Winchell (Peter Kueng, 2000) defined six desired properties separating KPIs from other measures. These properties may be seen as requirements for KPI engineering. We now explore what those properties mean for KPI modelling:

1. *KPIs should be in a Quantifiable Form.* Quantification means deriving a number or a conclusion from a set of instances of selected concepts in the models. Any KPI can be presented as a quantification predicate of first-order logic (Andrews, 2002).
2. *A KPI needs to be Sensitive to Change.* Any variation in the KPI measure in the model should vary with changes of predefined factors of the process inputs and/or with changes of the states of instances of selected concepts in the models.
3. *A KPI should be Linear.* Linearity of a KPI means that performance changes in line with the value of the indicator via a linear relationship. As non-linear, e.g. power and exponentially related KPIs exist, we change this requirement to 'the value of a KPI must be able to be shown by a consistent mathematical relationship in its simplest form'. This implies that linear relationships are the best as they are more easily tested, but that other numerical relationships can be used provided the relationship can be defined clearly.
4. *A KPI should be Reliable.* Reliability means that the algorithms for KPI calculation should be free from semantic errors and correctly calculate performance both in routine circumstances, as well as in unexpected circumstances. The validation of this property demands a model of the semantic concepts of the professional terminology and a model of the relevant aspects of the underlying business process.
5. *A KPI should be Efficient.*
(a) The indicators should be intuitive, unambiguous and easy to understand in order to avoid wasted effort or errors in their use and application. This efficiency can be achieved by demonstrating the semantics of the KPI definition on a model to the users.
(b) A KPI must be cost-effective to produce. This implies that a KPI should be created in the simplest way from any constituent metrics or indicators and that their production should use the simplest possible calculations. The number of elements (e.g. numerical inputs, states from different processes) comprising the KPI family is a measure of efficiency. If this number can be reduced, then the KPI family is not efficient and can be improved further.
6. *A KPI should be Oriented to Improvement, not to Conformance to Plans.* There is a danger that KPIs can be used to manipulate numbers instead of showing the improvement. The improvement oriented KPIs imply changes necessary to ensure competitive business performance. This property relates the KPI definitions both to strategic and tactical goals and to the underlying business process.

Analysis of the set of desired properties of KPIs shows that they cannot be validated without abstract models of the relevant aspects of the underlying business process and the goals of the assessed organizations. The input information for modelling the underlying business process and the goals of the assessed organizations should be derived from the definitions of the KPIs and their intended use. The validation of properties needs execution of the abstract business model.

The method for KPI modelling and validation should:

(1) enable understanding the KPIs and their relationships;

(2) provide intuitive models of KPIs :
(a) at the level of numerical modelling in order to test desired properties 1,2,3;
(b) at the level of goal and conceptual modelling in order to test desired properties 4,5,6.

3 METHOD FOR MODELLING OF KPIs

The input of our method is a document that defines KPIs for a branch of organizations. The KPIs are already designed, and the relevant elements of the business process are described in the definition of KPIs using the professional terminology.

The goal of our method is validating the assumptions about the relevant aspects of the underlying business process and assessing the KPI properties.

3.1 Case Study

We illustrate our method with a case study of the KPIs officially used in the programme IAPT (Improving Access to Psychological Therapies, 2013). The goals of this medical programme are the monitoring of the coverage by therapies and effectiveness of therapies for depression or anxiety disorders. It is expected that ”3.2 million people will access IAPT; 2.6 million patients will complete a course of treatment and up to 1.3 million (50% of those treated) will move to measurable recovery. For common mental health conditions treated in IAPT services, it is expected that a minimum of 15% of those in need would willingly enter treatment if available.” The indicators need to measure a quarter on quarter improvement. The IAPT program defines KPIs and High level indicators (HIs). HIs are strategic indicators used to support change decisions, while KPIs are at the tactical level of management and control performance.

The program defines a family of the following indicators:

KPI1: Level of Need. It presents the number of people who have depression and/or anxiety disorders in the general adult population. The number presenting population is produced as a result of the Psychiatric Morbidity Survey.

KPI2: No longer collected.

KPI3a: The number of people who have been referred for psychological therapies during the reporting quarter.

KPI3b: The number of active referrals who have waited more than 28 days from referral to first treatment/first therapeutic session (at the end of the reporting quarter).

KPI4: The number of people who have entered psychological treatment, (i.e. had their first therapeutic session) during the reported quarter is related to the concept person.

HI1: Access Rate. It indicates the rate of people entering treatment from those who need treatment $HI1 = KPI4/KPI1$.

KPI5: The number of people completed treatment.

KPI6: The number of people moving to recovery. This number sums up those who completed treatment, who at initial assessment achieve ”caseness” and at the final session did not.

KPI6a: No longer collected.

KPI6b: The number of people who have completed treatment but were not at ”caseness” at initial assessment.

HI2: Recovery Rate. It is calculated using the formula $HI2 = KPI6/(KPI5 - KPI6b)$.

The indicators in this case study do not measure the duration of operations or localized metrics of the detailed operational process steps and so cannot be classified as operational.

The set of indicators identifies two relevant improvement aspects of the business processes of the assessed organizations, namely an aspect of access to the treatment and the aspect of assessment of patients and their treatment.

3.2 Method Steps

Our method of KPI modelling contains the following steps:
1. Relating KPIs to goals of processes.
2. Conceptual modelling of the KPIs and the relevant underlying processes.
3. Relating the KPIs to the business concepts.
4. Protocol modelling of the business concepts.
5. Deriving the KPIs from the protocol models.
6. Validating the KPI properties by using the executable protocol model, goal and conceptual models.

1. Relating KPIs to Goals of Processes. In this step we use ideas of the well-established group of Goal-Oriented approaches to Requirements Engineering (GORE) (Dardenne et al., 1993; Regev and Wegmann, 2011). The notion of a goal is used as a partial description of a system state being a result of an execution of the system. The authors of the GORE methods emphasize the similarity between goals, requirements and concerns and propose to combine them in a tree structure. Goals are refined by requirements and concerns.

Figure 1 shows three processes relevant for modelling of the IAPT KPIs: (1) *Survey of the Needs of Population*, (2) *Psychological Therapy* and (3) *Program of Improving Access to Psychological Therapies (IAPT)*. Each of processes has its own goals.

The goals of the Psychological Therapy are: *"A Referred Person has access to psychological therapies"* and *"A Referred Person after treatment has improved conditions"*. These goals are combined with the AND operator. The *Survey of the Needs of Population* has the goal *"Estimate the size of the Population of People needed psychological treatment."*

The *Program of IAPT* has the goals *"Measure access to psychological therapies"* and *"Measure effectiveness of treatment"*. The goals of the *Program of IAPT* are related to the goals of the *Psychological Therapy*. We indicate these relations in Figure 1 as *"Monitor Access"* and *"Monitor Effectiveness of treatment"*.

A distinctive feature of our approach is deriving goals of the KPI measurement and the underlying business processes from the document with the KPI definitions. This approach allows us to identify the aspects of the business process that were used in the design of the KPI definitions: Estimate the size of population of people needed phycological therapy; Guarantee that a refereed person has access to phycological therapy; Guarantee the good chance that a refereed person after the treatment has improved conditions.

2. Conceptual Modelling of the KPIs. As in other GORE approaches the goals of each process are refined to concepts. The conceptual modelling is a wide variety of methods. The comparison studies conclude that "abstracting from their graphical form, the core expressivity of all conceptual models proposed in literature is similar" (Golfarelli, 2009). So, we choose the class diagram as the most used and standardized in UML. In this step, our method has similarity with the KAOS method (Dardenne et al., 1993).

Figure 1 shows that for modelling of each process, we extract corresponding concepts with attributes. The concepts are often the subjects of the process or the result of the process. The concept *Referred Person* corresponds to the *Psychological Therapy*. A *Referred person* is the subject of *Psychological Therapy* mentioned in the goals.

The concept *Survey* is the result of the process *Survey of the Needs of Population*.

In the search of generic concepts for modeling of KPIs we decided to use the concept *Dashboard* for each goal of the measurement process. A dashboard in the business intelligence domain presents a collection of measures of different levels supporting a particular request. The concept *Dashboard* allows us to collect the KPIs as attributes of the dashboard concept.

Using this choice the *Program of IAPT* process is modeled with two concepts *Access Dashboard* and *Recovery Dashboard*. *Access Dashboard* corresponds to the goal *"Measure access to psychological therapies"*. All the tactical and strategic measures needed for access measurement: **KPI1**, **KPI3a**, **KPI3b KPI4** and **HI1**, are modeled as attributes of the *Access Dashboard*. *Recovery Dashboard* with its attributes corresponds to the goal *"Measure effectiveness of treatment"*.

3. Relating the KPIs to the Business Concepts. The attributes of the dashboard concepts designed for modeling of KPIs need to be derived from the concepts of the relevant business processes. From the KPI definition, we extracted two relevant business processes and two corresponding business concepts for calculation of the KPIs of the IAPT program: *Survey* and *Referred Person*. Each of these concepts has its own attributes.

Conceptual modelling forces us to think about the attributes needed and semantics for KPI calculation. The attributes of concepts are taken from the KPI definitions. For example, the definition of KPIs says that the monitoring takes place quarterly. This implies that the concepts of the underlying process need the attributes representing the date of their appearance. The *Survey* gets its attribute *DateOfSurvey* and *Reffered Person* gets its attribute *DateOfReferring*.

We use also a generic attribute *State* and identify its possible values from the KPI definitions. For example, the names of the states of the life cycle of the *Referred Person*: *Referred*, *Waited 28 days*, *Entered treatment* and *Completed treatment*.

4. Protocol Modelling of the Business Concepts. Until this point, our method just extended the domain of application of GORE methods (such as KAOS) to modelling of KPIs.

From this point, our method becomes semantically different from other GORE methods. We introduce an intermediate step, namely modelling all the concepts as protocol machines.

The GORE models use activity diagrams or UML state machines for modelling of concepts. Activity diagrams and UML state machines have asynchronous semantics (McNeile and Roubtsova, 2010). The interaction of concepts is initialized by sending a mes-

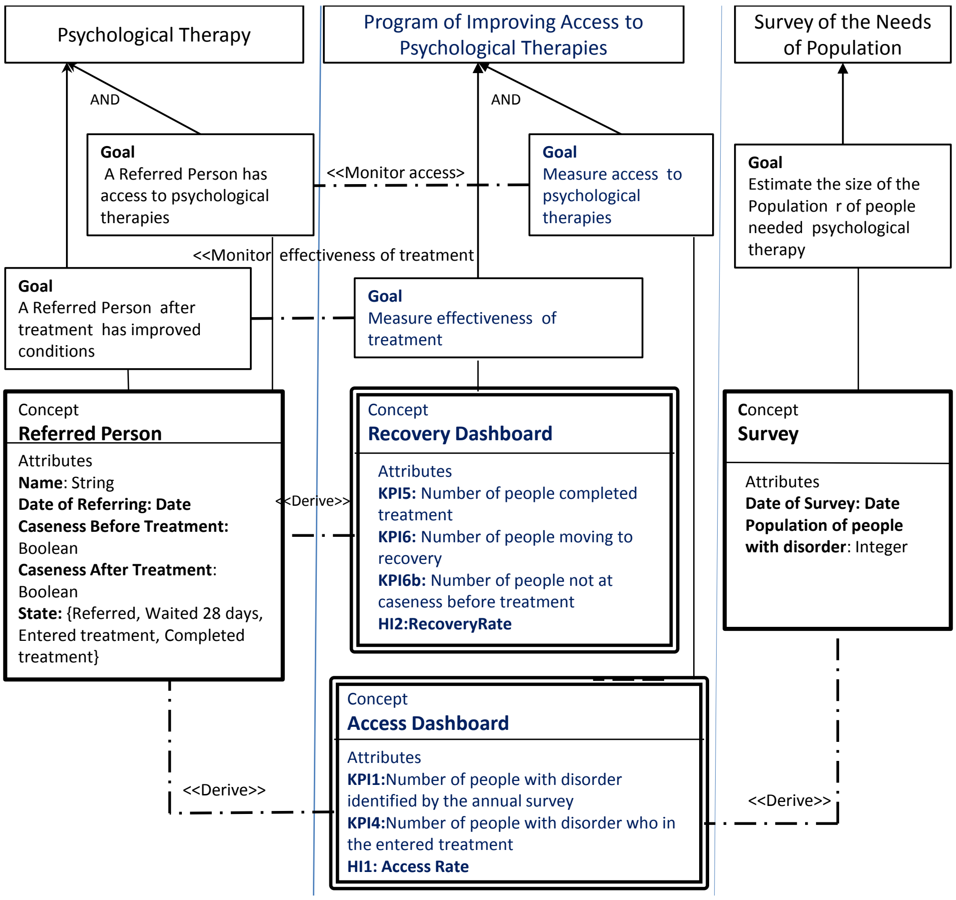


Figure 1: Goals and Concepts.

sage or by calling a method. These models suit for modelling of system implementation. Asynchronous interactions create many intermediate states that are not justified by the goals of the system (Gossler and Sifakis, 2003). Analysis of intermediate states may be relevant for validation of asynchronous implementation, but the KPIs are situated at the tactical and strategic level, at the level of visible states of the system.

We have chosen the protocol machines for the following reasons:

- Firstly, we want to minimize complexity of semantic modelling of business concepts. Protocol modelling uses a data-extended synchronous CSP-parallel composition (McNeile and Simons, 2006).CSP stands for Communicating Sequential Processes. The initial form of this composition operator was proposed in (Hoare, 1985). The initial operator was extended for machines with data by (McNeile and Simons, 2006). Because of the synchronous composition of its parts a protocol model presents only quiescent states of the system. This allows us to model only visible state and essentially decrease the state space of the model. All states can be justified by the system goals and subgoals. Modelling and reasoning can be focused on the business semantics.

- Secondly, we need to model KPIs that count entities in a specified state. The semantics of derived states is needed for KPI modelling as KPIs must reflect the state of business processes from which the measurements are obtained to drive the KPIs. Activity diagrams and UML state machines do not have semantics allowing one concept to derive its states from the state of other concepts (OMG, 2003).

- Thirdly, we should be able to model crosscutting sub-processes as KPIs may collect information from different elements of the system. Crosscutting sub-processes are repeated parts of a business process that cannot be decomposed using hierarchy or sequential decomposition. Activity diagrams and UML state machines cannot easily specify the crosscutting sub-processes as the composition techniques of the diagrams are restricted with sequential and hierarchical composition (Gossler and Sifakis, 2003). Separating crosscutting sub-processes can be used for efficient modelling of KPIs. Combining different measures at different abstraction level presented with crosscutting parts simplifies the KPI models. Protocol modelling supports the CSP parallel composition of crosscutting sub-processes in the business process that simplifies the KPI modelling.

A protocol machine is an object life-cycle model. It is presented as a state transition structure with a data storage that defines the ability of a system to interact with its environment by accepting or refusing events from the environment. A protocol machine can be seen as an object that exists even without its creation in its initial state. An object goes into its active state accepting a creating event. All protocol machines are composed with the data-extended synchronous CSP-parallel composition.

The Protocol Model of the Program for IAPT. Protocol machines used for modelling of KPIs are graphically shown in Figure 2. Each concept from Figure 1 has the corresponding protocol machine at Figure 2 with the same name. The boxes in Figure 2 represent protocol machines, where the attributes of concepts are shown in bubbles. The graphical presentation does not provide all the elements of the model. The complete protocol model of the program for IAPT is presented with its metacode and small java functions and can be found in (Roubtsova, 2013). For instance, the metacode of an instance of the `Survey` concept is a protocol machine `Survey`.

```
OBJECT Survey
NAME SurveyName
    ATTRIBUTES  SurveyName: String,
                Population:Integer,
                 DateOfSurvey:Date
    STATES created
    TRANSITIONS @new*CreateSurvey=created

EVENT CreateSurvey
    ATTRIBUTES  Suvey:Survey,
                SurveyName:String,
                Population:Integer,
                DateOfSurvey:Date
```

It is assumed that the survey is periodic (the period is not given), and we add the tacit attribute to the `Survey`: `DateOfSurvey:Date`. Only the `Survey` instance closest to the date of KPI monitoring is used for KPIs calculation.

Each instance of the `Survey` is created by accepting an event `Create Survey`. Only the `Survey` in state ”`created`” can provide the values of its attributes of the `LevelOfNeed` and `Population` for performance indicators. The acceptance of an event `Create Survey` brings with its attribute `Population` the number of people who have depression and(or) anxiety disorders and with its attribute `DateOfSurvey` the value of the attribute of the protocol machine `Survey`.

By accepting or refusing this event the protocol model communicates with the environment. This communication is simulated in Modelscope tool supporting Protocol Modelling (McNeile and Simons, 2005).

A protocol machine may also describe a part of the life cycle of an object. The metacode is started with the key word BEHAVIOURFor example, the concept *Refereed Person* is presented as the protocol machine `Referred Person`. This protocol machine includes behaviour `Treatment`. Both protocol machines `Referred Person` and `Treatment` are synchronized with an event `EnterTreatmentAndAssess`. The separation of the concept `Treatment` simplifies the quantification on the states of the `Referred Person` used for KPI definition.

```
OBJECT ReferredPerson
    NAME PersonName
    INCLUDES Treatment
ATTRIBUTES PersonName: String,
                DateOfReferring:Date,
STATES referred, 28DaysWaited, left,
enteredTreatmentAndInitiallyAssessed,
      completedTreatmentAndAssessed
    TRANSITIONS @new*Refer =referred,
                referred*Decline=left,
                left*Return=referred,
                referred*Wait=28DaysWaited,
                28DaysWaited*Leave=left,
    28DaysWaited*EnterTreatmentAndAssess=
                28DaysWaited

BEHAVIOUR Treatment
    ATTRIBUTES CasenessBefore:Boolean,
               CasenessAfter:Boolean
    STATES enteredTreatmentAndInitiallyAssessed,
   completedTreatmentAndAssessed,left
    TRANSITIONS
    @new*EnterTreatmentAndAssess
    =enteredTreatmentAndInitiallyAssessed,
enteredTreatmentAndInitiallyAssessed*Leave=left,
    enteredTreatmentAndInitiallyAssessed*
    CompleteTreatmentAndAssess
        =completedTreatmentAndAssessed

EVENT Refer
ATTRIBUTES  ReferredPerson:ReferredPerson,
            PersonName:String,
                DateOfReferring:Date
EVENT Decline
    ATTRIBUTES  ReferredPerson:ReferredPerson
EVENT Return
    ATTRIBUTES  ReferredPerson:ReferredPerson
EVENT Wait
    ATTRIBUTES  ReferredPerson:ReferredPerson
EVENT Leave
```

```
    ATTRIBUTES  ReferredPerson:ReferredPerson
EVENT EnterTreatmentAndAssess
    ATTRIBUTES ReferredPerson:ReferredPerson,
               CasenessBefore:Boolean
EVENT CompleteTreatmentAndAssess
    ATTRIBUTES ReferredPerson:ReferredPerson,
               CasenessAfter:Boolean
```

Attributes of the protocol machines `Referred Person` and `Treatment` are shown in bubbles in Figure 2. Attributes `CasenessBefore:Boolean` and `CasenesAfter:Boolean` model the procedure of assessment of the patient's conditions. Each transition is labelled with an external event. As events are the structures that carry data, they allow the model to updatethe attributes in the life cycle of an instance of the `Referred Person`. The value of the `DateOfReferring` is entered with event `Refer`. `CasenessBefore` is inserted with event `EnterTreatmentAndAssess`. `CasenessAfter` is entered with event `CompleteTreatmentAndAssess`.

5. Deriving the KPIs from the Protocol Models of Concepts. An instance of a dashboard protocol machine models a KPI report request. The metacode of the `Access Dashboard` is shown below.

```
OBJECT AccessDashboard
NAME  DashboardName
ATTRIBUTES,
    DashboardName:String,
        StartOfReportingQuarter:Date,
        !LevelOfNeed:Integer,
        !NumberReferredPersons:Integer,
        !NumberReferredPersonsWaited:Integer,
        !NumberOfEnteredTreatment:Integer,
        !AccessRate: Integer,
STATES created
TRANSITIONS
        @new*CreateAccessDashboards=created
```

The protocol machines `AccessDashbord` and `RecoveryDashboard` present KPIs monitoring the access and recovery by selecting and counting the instances of the protocol machines. Derived attributes of dashboard protocol machines marked by the exclamation symbol "!" represent individual KPIs. The event `CreateAccessDashboard` is used to insert the value of the starting date of the reporting quarter into the attribute `StartOfReportingQuarter:Date`.

Each dashboard protocol machine reads the state of protocol machines `Survey` and `Referred Person` and derives the values of own attributes presenting KPIs.

Each derived attribute presenting a KPI has a corresponding derivation function. The functions are stored in the java files extending behaviour of dashboard protocol machines. For example, the attribute **KPI1**: `LevelOfNeed` is calculated using its request function presented below.

```
public class AccessDashboard extends Behaviour{
//KPI 1 Level of Need

public int  getLevelOfNeed() {
int LevelOfNeed=0;

// choose the date three years ago
Calendar cal = Calendar.getInstance();
cal.add(Calendar.YEAR, -100);
Date dd = cal.getTime();

Date Dashd =
this. getDate("StartOfReportingQuarter");

Instance[] existingSurvey =
selectInState("Survey", "@any");

for
(int i = 0; i < existingSurvey.length; i++) {
    Date SD=
    existingSurvey[i].getDate("DateOfSurvey");
if
(SD.compareTo(dd)>0 && SD.compareTo(Dashd)<0)
{LevelOfNeed=
existingSurvey[i].getInteger("Population");
dd=SD;
}}
return LevelOfNeed;
}}
```

`AccessDashboard` reads the state of protocol machine `Survey` (but does not change it). It finds the Survey with the closest date and takes the value of the `Population` of this `Survey` and assigns it to own attribute `LevelofNeed`. The search function `selectInState("Survey", "@any")` selects the set of surveys to choose the latest survey from this set.

Protocol modelling has predefined select functions useful for definition of tactical KPIs:
- *selectInState ("BehaviourName","State")* returns an array of instances, all of which include the specified behaviour;
- *selectByRef("BehaviourName","AttributeName")* returns an array of instances, all of which include the specified behaviour (or object) and have the specified attribute.

6. Validating the KPI Properties by using the Executable Protocol Model, Goal and Conceptual Models. At the moment, when the KPIs are protocol modeled as valid numeric algorithms they can be analyzed and tested. The algorithms for KPIs are presented in java files and implement exceptions, e.g. for the division by zero for `Access Rate` and `Recovery Rate`.

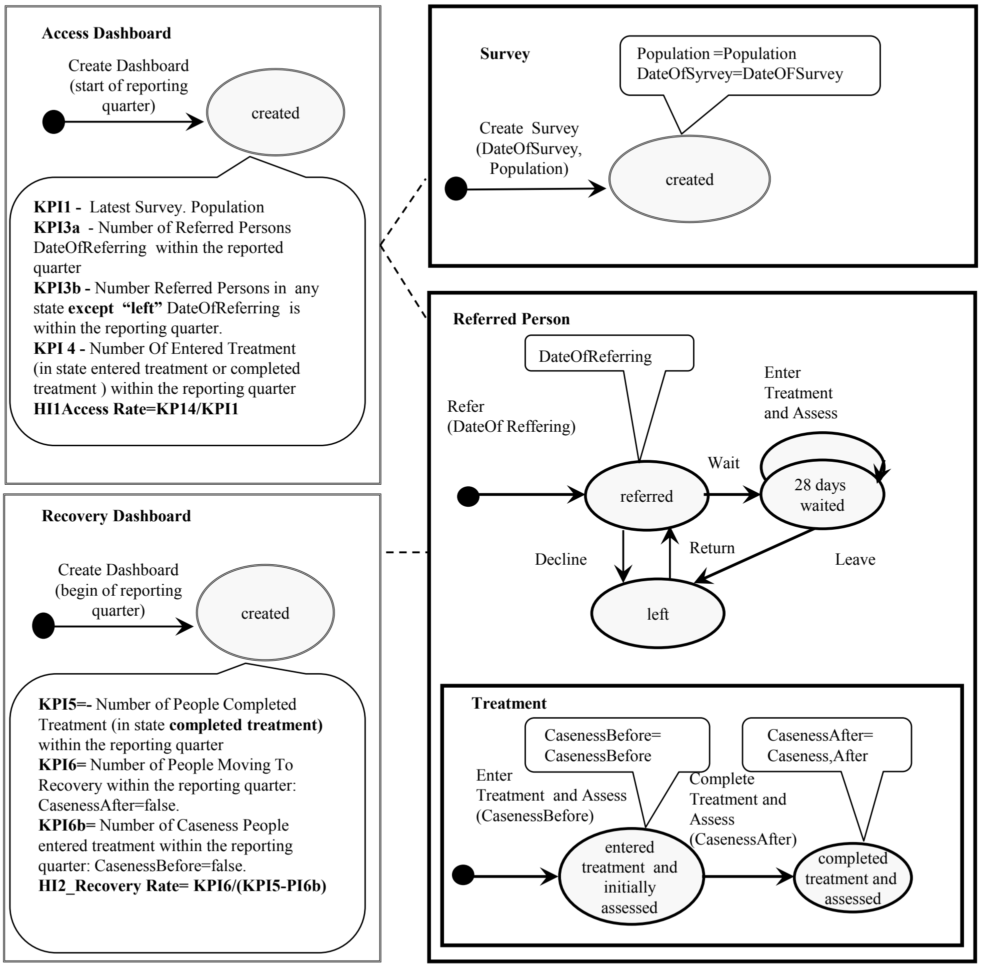


Figure 2: Protocol Model of the IAPT KPIs.

Quantifiability. Quantification means deriving a number or a conclusion from a set of instances of selected concepts in the models. As we apply the select functions for KPI derivation, this means that quantification is already built into the KPI derivation procedure.

Sensibility to the Change. In order to test the sensitivity to change for any KPI, the protocol model is populated with instances.

Let us assume that we want to test how val the KPI1:Level of Need is changed when the new instance of `Survey` appears. We create two instances of the protocol machine `Survey` with different `DateOfSurvey` and one instance of the `AccessDashboard` with `StartOfReportingQuarter`. We need to test that the closest instance of the `Survey` will be chosen to update the KPI1. We do not need to create a database to test the KPI definition in protocol modeling. We use only two instances of the protocol machine `Survey` needed for validation of the sensitivity to the change of the `DateOfSurvey`.

Linearity. In order to test linearity of a KPI to a number of instances, we need a model with N instances and a model with $N+1$ instances. The tests are collected during the execution. Because the protocol model has only the quiescent states, the KPI-attributes of the dashboard, are derived from other protocol machines at the same moment. For example, all KPIs of `AccessDashboard` are derived at the `StartOfReportingQuarter`.

Reliability of the Business Process used for KPI Definitions. Execution may question semantic reliability of KPIs because of incompleteness of the business process used in the KPI definition.

For example, the procedure of testing `Caseness` is not specified in document of the IAPT (Improving Access to Psychological Therapies, 2013). The strategic *HI:Recovery Rate* depends on the quality of the procedure of testing `Caseness` both before and after treatment. The *Recovery Rate* is increased if more of healthy people with assessed as "false" will enter the treatment. The *Recovery Rate* is increased if more of sick people are wrongly assessed as "false" will leave the treatment. In other words, the quality of testing is the point of attention for management of the business process using this KPI family. In order to improve reliability of the `HI: Recovery Rate`, the procedure of the `Caseness` testing should be specified.

Efficiency of KPI Sets. It is more difficult to validate efficiency as there are usually several different ways to collect data from the model for a KPI calculation. The KPI is efficient in organization if it is simple and well understood. In this way, the efficiency is related with semantic reliability.

In general, the KPIs should not duplicate each other. As we analyse the working programme, the duplications were already avoided. In the IAPT programme, the KPI2 and KPI6a duplicate other KPIs. They were found superfluous already by organizations trying to apply the set of the IAPT KPIs. The validation could be done on the model.

Improvement Orientation of KPIs. The most important property of KPIs is improvement orientation. There is a danger of replacing the improvement orientation of KPIs with the plan orientation. In this case, the KPIs may be used for manipulating numbers. The value of an improvement oriented KPI cannot be manipulated in the attempt to meet its planned value.

Our case study presents examples of both an improvement oriented KPI and a possibly plan oriented KPI.

The HI $AccessRate = (NumberOfEnteredTreatment/LevelOfNeed)$ is an example of the improvement oriented KPIs. It corresponds to the goal: *"A Referred Person has access to psychological therapies."* The improvement means positive growth of the ratio of treated people to the people needing treatment. Modelling shows that the numerator and denominator of the KPI are objective values that grow through the model execution and cannot be manipulated in the defined process. The `LevelOfNeed` comes from an independent process Survey. The `NumberOfEnteredTreatment` is a summation of individually `Referred Persons`, which are independent of the treatment providing

The HI $RecoveryRate = NumberofPeopleMovingToRecovery/(NumberofPeopleCompletedTreatment - NumberOfCasenessPeopleBeforeTreatment)$ may become plan oriented and open to manipulations. For validation of the improvement orientation of this indicator, we use both the goals associated with KPIs and the model of the underlying process.

The KPI corresponds to the goal *"A Referred Person after treatment has improved conditions"*. The improvement corresponds to the growth of the `Recovery Rate`, but the growth may be manipulated by the procedure of the assessment `Caseness` both before and after treatment. If this procedure is independent of the process of treatment and well defined/specified to avoid manipulations, the `Recovery Rate` is improvement oriented. If the procedure of `Caseness` assessment belongs to the treatment process, and this treatment process gets funding on the basis of this KPI, then the value of `Recovery Rate` can be manipulated to meet the planned values by assessing healthy people as sick before the treatment and sending them for the treatment or by assessing sick people as healthy after the treatment.

4 RELATED WORK AND CONCLUSIONS

The KPI specific modelling techniques described by (Strecker et al., 2012; Frank et al., 2009; Popova and Sharpanskykh, 2010), are based on conceptual modelling and propose metamodels for KPI design. They are aimed to integrate the enterprise models with a model of performance measurement systems and use the integrated model as a basis for further analysis at different organizational levels of abstraction. The methods do not have means for execution of process models and validating the properties of KPIs.

The aim of our method is different as we take a designed set of KPIs as an input from a document or a standard and then model and analyse the abstract business process derived from this KPI set and the properties of KPIs. We relate KPIs only to the abstract business processes derived from the KPI definitions and therefore, simplify the analysis.

We don't restrict our modelling techniques with conceptual modelling. Our method combines elements of goal modelling, conceptual modelling and protocol modelling. The need of a practical combination of these methods to validate KPI properties led us to the choice of the synchronous protocol modelling technique. As a protocol model is a combination of process model and data model, it contains useful procedures for communication of protocol machines with data and advanced procedures for derivation of states of one protocol machine from the states of others. These advanced protocol modelling operators are supported with the Modelscope tool (McNeile and Simons, 2005). The model is executed and tested.

The combination of techniques proposed in this paper supports the requirements for the method for KPI modelling as it presents a) an abstraction of the available conceptual models enabling modelling of KPIs and understanding their relationships; b) intuitive models of KPIs of different levels. Our method enables model execution and therefore, validation of the desired properties of KPIs. The goal model and the executable protocol model support the validation of the properties of KPIs including semantic reliability and improvement orientation. The validated processes and KPIs can be used for implementation modelling.

In the future work, we plan to further develop techniques to test semantic reliability and efficiency of a KPI and adapt our method for design and analysis of tactical, strategic and complex KPIs (Robert Kaplan, and David Norton, 2001) used in industry.

We also plan to integrate our method with the methods for system implementation modelling. The models of implementation are usually asynchronous. All asynchronous deviations of their behaviour from the visible system behaviour presented with protocol models should be required or accepted by users. In such a way using protocol models may contribute to reliability of requirements and models of implementations.

REFERENCES

Andrews, P. (2002). *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof*. Berlin: Kluwer.

Berler, A., Pavlopoulos, S., and Koutsouris, D. (2005). Using key performance indicators as knowledge-management tools at a regional health-care authority level. *IEEE Transactions on Information Technology in Biomedicine*, 9(2):184–192.

Dardenne, A., van Lamsweerde, A., and Fickas, S. (1993). Goal-directed requirements acquisition. *Sci. Comput. Program.*, 20(1-2):3–50.

Frank, U., Heise, D., and Kattenstroth, H. (2009). Use of a domain specific modeling language for realizing versatile dashboards. *In: Tolvanen JP, Rossi M, Gray J, Sprinkle J (eds) Proceedings of the 9th OOPSLA workshop on domain-specific modeling (DSM)*.

Garengo, P., Biazzo, S., and Bititci, U. S. (2005). Performance measurement systems in SMEs: A review for a research agenda. *International Journal of Management Reviews*, 7:25–47.

Golfarelli, M. (2009). From User Requirements to Conceptual Design in Data Warehouse Design - a Survey. *In Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction. L. Bellatreche (Ed.)*, pages 1–14.

Gossler, G. and Sifakis, J. (2003). Composition for Component Based Modeling. *Springer,LNCS*, 2852):443–46.

Hoare, C. (1985). *Communicating Sequential Processes*. Prentice-Hall International.

IAB Europe (2013). *http://www.iabeurope.eu/news/launch-of-a-committee-on-standards-and-kpis-for-brand-and-audience-campaigns.aspx*.

Improving Access to Psychological Therapies (2013). IAPT Key Performance Indicator (KPI). Technical Guidance for Adult IAPT Services. *http://www.iapt.nhs.uk/silo/files/iapt-kpi-technical-guidance-201213-v20-.pdf*.

KPIStandard (2013). *http://kpistandard.com/html/EN/Sectors*.

McNeile, A. and Roubtsova, E. (2010). Aspect-Oriented Development Using Protocol Modeling. *LNCS 6210*, pages 115–150.

McNeile, A. and Simons, N. (2005). http://www.metamaxim.com/.

McNeile, A. and Simons, N. (2006). Protocol Modelling. A Modelling Approach that Supports Reusable Behavioural Abstractions. *Software and System Modeling*, 5(1):91–107.

Neely, A., Mills, J., Platts, K., Richards, H., Gregory, M., Bourne, M., and Kennerley, M. (2000). Performance measurement system design: developing and testing a process-based approach. *International Journal of Operations and Production Management*, 20(10):1119 – 11452.

OMG (2003). *Unified Modeling Language: Superstructure version 2.1.1 formal/2007-02-03*.

Parmenter, D. (2010). *Key Performance Indicators, Developing, Implementing and Using Winning KPIs*. John Wiley & Sons, New Jersey.

Peter Kueng (2000). Process performance measurement system - a tool to support process-based organizations. *TOTAL QUALITY MANAGEMENT*, 11(1):67–85.

Popova, V. and Sharpanskykh, A. (2010). Modeling organizational performance indicators. *Information systems*, 35(4):505–527.

Regev, G. and Wegmann, A. (2011). Revisiting Goal-Oriented Requirements Engineering with a Regulation View. *Springer. LNBIP*, 109):56–69.

Robert Kaplan, and David Norton (2001). Transforming the Balanced Scorecard from Performance Measurement to strategic management: Part I. *Accounting Horizons*, pages 87–104.

Roubtsova, E. (2013). Protocol Model of the KPIs from ”The programme ”Improving Access to Psychological Therapies. *http://www.open.ou.nl/elr/IAPT.zip*.

Strecker, S., Frank, U., Heise, D., and Kattenstroth, H. (2012). MetricM: A modelling method in support of the reflective design and use of performance measurement systems. *Springer, Information Systems and e-Business Management*, 10:241–276.

Value Map
A Diagnostic Framework to Improve Value Creation and Capture in Service Systems

Arash Golnam
School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne , Lausanne, Switzerland
arash.golnam@epfl.ch

Vijay Viswanathan
Webdoc, Rue de Genève 17 Lausanne, 1003, Switzerland, Switzerland
vviswanathan@gmail.com

Christa I. Moser
Business School Lausanne, Route de la Maladière 21, 1022 Chavannes, Switzerland
christa.moser@bsl-lausanne.ch

Paavo Ritala
School of Business, Lappeenranta University of Technology, Lappeenranta, Finland
paavo.ritala@lut.fi

Alain Wegmann
School of Computer and Communication Sciences, Ecole Polytechnique Fédérale de Lausanne , Lausanne, Switzerland
alain.wegmann@epfl.ch

Keywords: Modeling, Problem Structuring Method (PSM), SEAM, Service Systems, Value Creation and Capture, Value Map.

Abstract: In this paper, we introduce a problem structuring method (PSM) called "Value Map". Value Map is an extension to the Supplier Adopter Relationship Diagram in the Systemic Enterprise Architecture Method (SEAM). Value Map assists in understanding, analysis and design of value creation and capture in service systems. To develop the Value Map, we reviewed the literature that examine value creation and capture particularly in the marketing and microeconomics domains. The literature review helped us to discover and explore the relationships among the important concepts relevant to the processes of value creation and capture. Having identified these concepts and their relationships, we graphically represented them in the form of a conceptual model. The conceptual model provided insights into the structure and the dynamics of value creation and capture and served as a reference point for developing the notational elements and the modeling constructs in the Value Map. We illustrate the applicability of the Value Map by modeling value creation and capture in the service system of a social networking company called Webdoc.

1 INTRODUCTION

In the theories of economic exchange, value was traditionally viewed only from the perspective of monetary transactions between the customers and the organization. Value was perceived to be rooted in goods that were produced by the organization. Once distributed to customers, the value produced was destroyed, or consumed. In the marketing literature, this perspective is broadly referred to as the Goods-Dominant (G-D) logic, which was prevalent pre-1900s. From the standpoint of the G-D logic, customers played a negligible role in the value creation process. In other words, the organization created what was perceived as valuable to the customer (Prahalad and Ramaswamy, 2004) without the involvement of the customer.

According to Vargo et al. (2010, 136), the underlying objective of the G-D logic is to "maximize operational efficiency and reduce firm costs in order to increase financial profits". Moreover, G-D logic primarily focuses on *operand resources* (i.e., those resources that are tangible; physical goods) that are manifested in products.

In a paradigm shift, the economic exchange model has been augmented and extended to include *customers* as a fundamental tenet of the value creation process. This shift has led to the emergence of the Service-Dominant (S-D) logic. The G-D and the S-D logic differ in a number of important ways, (see Table 1).

Table 1: G-D Logic vs. S-D Logic.

	G-D Logic	S-D Logic
Focus	Operand resources; creating goods to be sold	Operant resources; intangible resources (i.e., knowledge and skills)
Goods	The product of value to be exchanged	Seen as intermediaries in service delivery
Service	Intangible output of a good	Service is the foundation of all exchange
Value	Created within organizations	Co-created by organizations and customers

The focus in the S-D logic is on intangibles, competencies, dynamic exchange processes and relationships that are broadly referred to as *operant resources*. Operant resources have an influence on other resources to create benefit through the service (Vargo et al., 2010).

The concept of a good in the G-D logic is the product of value to be exchanged. While, in the S-D logic, a good is merely seen as an intermediary in the delivery of service, broadly viewed as delivery mechanisms for services (Vargo and Lusch, 2008).

Furthermore, in the S-D logic, the concept of service is extended beyond a "particular" kind of intangible good (i.e., knowledge and skills) or an intangible output of a good. Instead, service is deemed as the foundation of all exchange (i.e., service exchanged for service) (Vargo and Lusch, 2004a).

Finally, the S-D perspective conceptualizes a firm's offerings not as an output, but as an input for the customer's value-creation process. Thereby, instead of viewing value as being created within companies, value is increasingly viewed as being co-created between companies, customers, and other actors within a *service system*.

Service systems are the arrangement of resources, including people, information, and technology (Vargo et al., 2009). In service systems, value is perceived as being created in collaboration with the customer (Sphorer and Maglio, 2008). Grönroos, (1979, 2006, 2008), Ballantyne and Varey (2006), and Gummesson (2007) argue that in the S-D logic, the supplier is not the sole creator of value, but that value emerges when the customer is involved in the process. Thus, from the S-D standpoint, customers are the eventual locus and the determining party of the value that is created (Sandström and Kristensson, 2008). Lusch and Vargo (2006) suggest that the customer's collaborative role in value creation is what is known as *co-creation of value*.

Moreover, the S-D logic emphasizes on the subjective and experiential nature of value and thus asserts that value is "uniquely and phenomenologically determined by the beneficiary" (Vargo and Lusch, 2008). Based on this perspective, a distinction is made between *value-in-use* and *value-in-exchange*. Value-in-use refers to the specific qualities of the service. These qualities are perceived by users in relation to their needs (i.e., speed or quality of performance, aesthetics, or performance features). Value-in-exchange can be defined as the "monetary amount realized at a certain point in time in exchange" (Lepak et al., 2007).

After value has been created, it is important for the organization to capture this value. Lepak et al. (2007) explain that some value may be lost or in some cases, shared with other stakeholders. Value capture, also termed value retention or value appropriation, deals with the amount of exchange value the customer has kept and retained by the organization in the form of profit (Bowman and Ambrosini, 2000). From a non-monetary perspective, value capture can be described as the degree to which service quality goals have been met or exceeded (Parasuraman et al., 1985).

Once value has been (co)-created, the viability of the service system depends on its ability to capture the created value. In other words, the service provider sustains its existence with the value it retains (Ritala et al., 2009). Thus, it can be asserted that sustainable value (co)-creation and capture is an imperative for viability of service systems.

In the service science literature, a number of modeling frameworks provide conceptual tools to support the design of service offerings (see for instance Gordijn and Akkermans, 2003; Weigand et al., 2009; Pijpers and Gordijn, 2007; Yu, 1997; Weigand, 2009). Such modeling frameworks, however, mainly address the design and analysis of value from the customers' perspective and do not sufficiently address service providers' value capture in the service value equation. In general, the same gap can be broadly identified in the service literature, where value (co)-creation has often been emphasized over value capture.

Moreover, there are nonlinearities and feedback structures inherent in the interplay between value creation and capture in service systems. For instance, a slight increase in price, results in the loss of a huge proportion of the market, or, a new service feature can boost the customer base of a service provider. While presenting both conceptual and practical challenges for service providers and service science researchers, this systemic interconnectedness has been glossed over in the service science research.

To tackle the above mentioned research gaps, in this study, we introduce the Value Map; a framework for modeling value in service systems that takes into account both value creation (for and with customers) and value capture (by service providers). The Value Map can be broadly referred to as a Problem Structuring Method (PSM) (Mingers and Rosenhead, 2004; Rosenhead, 1996; Rosenhead and Mingers, 2001) that aims to provide conceptual and practical assistance in analyzing, reconfiguring and designing value in service systems. The modeling constructs and notational elements in the Value Map are derived from a literature review we conducted to gain a new perspective into the structure and the dynamics of value creation and capture.

This paper is organized in the following way. In Section 2, we elaborate on the structure and the results of the literature review we conducted to discover the important concepts relevant to value creation and capture. In order to gain a better understanding, the concepts and their relationships were formalized in 10 algebraic functions and were graphically represented in form of a conceptual model. In Section 3, we introduce the value map and its modeling constructs and notational elements. In Section 4, we present the results of the application of the Value Map to model and improve value creation and capture in a social networking platform called Webdoc. Section 5 includes the related work. In this section we briefly report on the results of an empirical study we conducted to assess the usefulness of the Value Map and to compare it to an established method for presenting business models. Finally, in Section 6 we present our conclusions, limitations of research and our future work.

2 THE CONCEPTUAL MODEL

In this section, we present the structure and the results of the literature review we conducted on the theoretical frameworks that examine value creation and capture. A literature review can be conducted for a variety of purposes see (Hart, 1999, 27). In this paper, the literature review will help us discover the important concepts relevant to value creation and capture and explore the relationships among these concepts in order to gain a new perspective into the structure and the dynamics of value creation and capture. Thus, the literature review helps us understand the "what" (i.e., the concepts), the "how", (i.e., their relationships) and the "why" (i.e., the rationale behind the selection of the concepts and the perceived relationships among them). According to Whetten (1989), the "what", "why" and the "how" are the three tenets of a theoretical contribution.

The correct selection of the published materials is a vital element of a literature review. We followed Baker (2000) and developed a number of criteria for selection of the work to be included in the literature review. The articles we included in the literature review addressed value creation and capture simultaneously, and were indexed by Institute for Scientific Information (ISI). These two criteria led us to a total of around 30 articles. We then derived the key concepts discussed in each article. The concepts were then analyzed and divided into three categories: customer value, customer value creation process, and service provider value capture. Next, for each category, we developed a number of functions that embody algebraic expressions explaining the relationships between the concepts (see Table 2). Having identified the concepts and their relationships, we graphically represented them in form of a conceptual model made up of boxes (i.e., the concepts) and arrows (i.e., their relationships), (see Figure 1). According to Whetten (1989, 491), "such visual representations often clarify the author's thinking and increase the reader's comprehension". As illustrated in Figure 1, we have marked the three categories of concepts in the conceptual model.

3 THE VALUE MAP

Figure 2 represents the actors and their properties in a service system. We refer to this representation as the Service System Model. As illustrated, the *Service System* is composed of a *Service Provider*

Table 2: The Algebraic Functions capturing relationships between Customer Value, Customer Value Creation Process, and Service Provider Value Capture concepts.

Customer Value Conceptualizations	1	Net perceived customer value (NPCV) = (perceived service benefits) – (perceived service costs) *NPCV equals the benefits minus the costs of receiving the service.* (Kotler, 2000; Day, 1990; Huber, 2001)
	2	Perceived benefits of the service offering = (perceived functional benefits) + (perceived emotional benefits) *The sum of the functional and emotional benefits constitutes the perceived benefits of the service offering.* (Kotler, 2000; Grönroos, 2000)
	3	Perceived costs of the service offering = (Perceived non-monetary costs) + (Perceived monetary costs) *The costs incurred to the customer who receives the service are divided into two categories: monetary cost and non-monetary costs that can include time, energy, and psychic costs.* (Kotler, 2000; Bowman and Ambrosini, 2000)
	4	Relative NPCV of the service offering = (NPCV of the service provider's value network offering) – (NPCV of the competing value network's service offering) *Relative net perceived customer value is the net perceived value created by a service provider's offering in relation to the competing offerings.* (Bowman and Ambrosini, 2000)
Customer Value Creation Process Conceptualizations	5	Service components $\subset$ Resources and capabilities (of the service provider and its value network) *Service components are a subset of the resources and capabilities of the service provider and its value network that are manifested in the service.* (Kothandaraman and Wilson, 2001; Moller and Svahn, 2006; Bowman and Ambrosini, 2000)
	6	Service components (of service provider and its value network) $\Rightarrow$ Service features $\Rightarrow$ Service value attributes (of service customer) *Service components create some emergent properties for the service, which are noticed by the customer. We refer to these emergent properties of the service as service features. Service features impact the perceived customer value through various value attributes.* (Pynnonen et al., 2011)
Service Provider Value Capture Conceptualizations	7	NPCV of the service offering $\propto$ Service providers benefits *The customer's relative perception of value determines the actions the customers undertake, which result in generating more or less benefits for the service provider.* (Bowman and Ambrosini, 2000)
	8	Net captured value (NCV) of the service provider = (Value captured by the service provider) – (Cost of the service components) *The NCV is the value captured by the service provider minus the costs of the service components.* (Bowman and Ambrosini, 2000)
	9	(Non-)monetary benefits for the service provider $\propto$ Value captured by the service provider *The (non)monetary benefits created by the customer for the service provider are proportional to the value captured by the service provider.* (Nelson and Winter, 1982; Allee, 2008; Ulaga, 2003)
	10	Costs of the service components = (Organizing costs i.e. internal costs of the service provider) + (Opportunity costs i.e. external costs of the suppliers in service provider value network) *The costs of the service components equal the sum of the organizing costs of the service provider, and the external opportunity costs of the suppliers in the value network.* (Masten et al., 1991; Blomqvist et al., 2002)

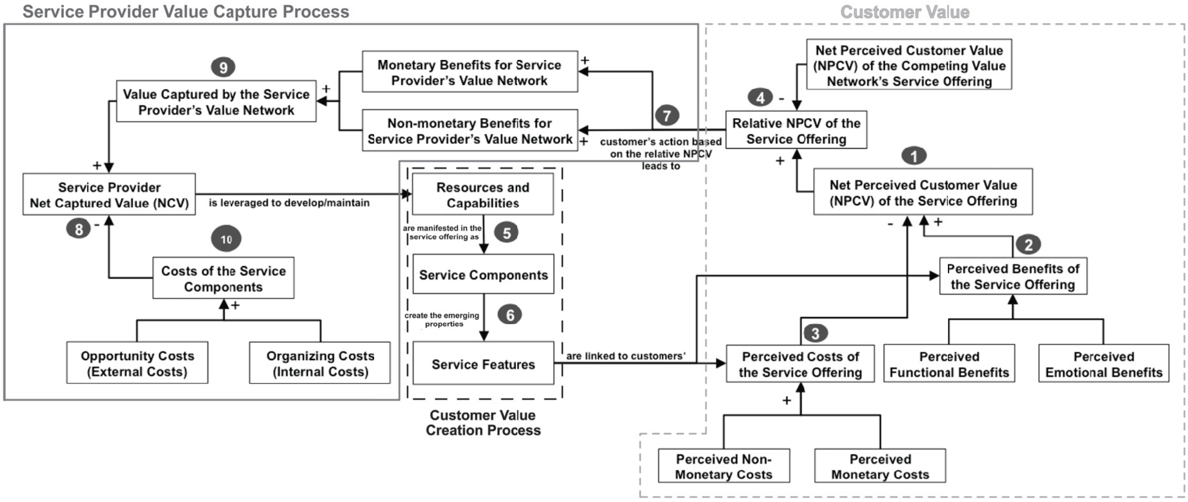


Figure 1: Graphical representation of value creation and capture concepts and their relationships.

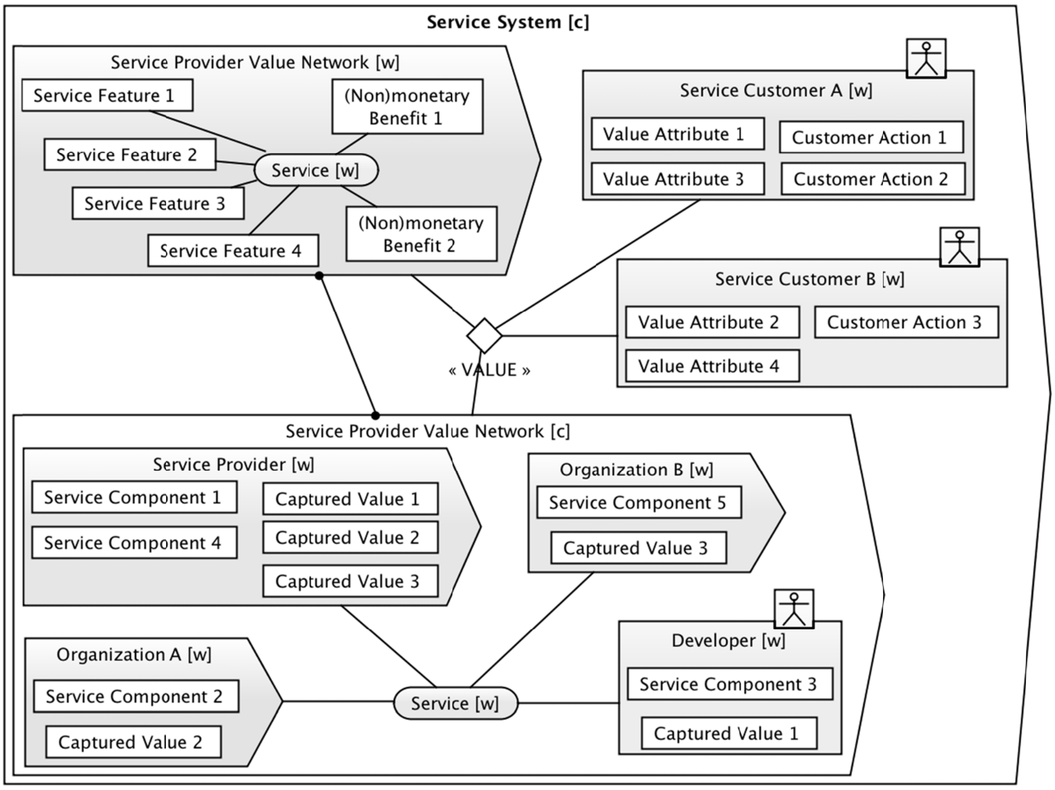


Figure 2: Service system model.

Value Network and *Service Customer A and B*. The *Service Provider Value Network* can be represented as a black-box or a white-box denoted respectively by grey and white colors. In Figure 2, [w], [c] denote whole (black- box) and composite (white-box) representations of the systems and entities. When represented as a black-box we model the *Service,* the *Service Features*, and the *(Non)monetary Benefits* for the *Service Provider Value Network* as its emergent properties. The white-box view of the *Service Provider Value Network* provides insight into the configuration of the value network. Thus, we will be able to view the organizations or the people who compose the value network and their

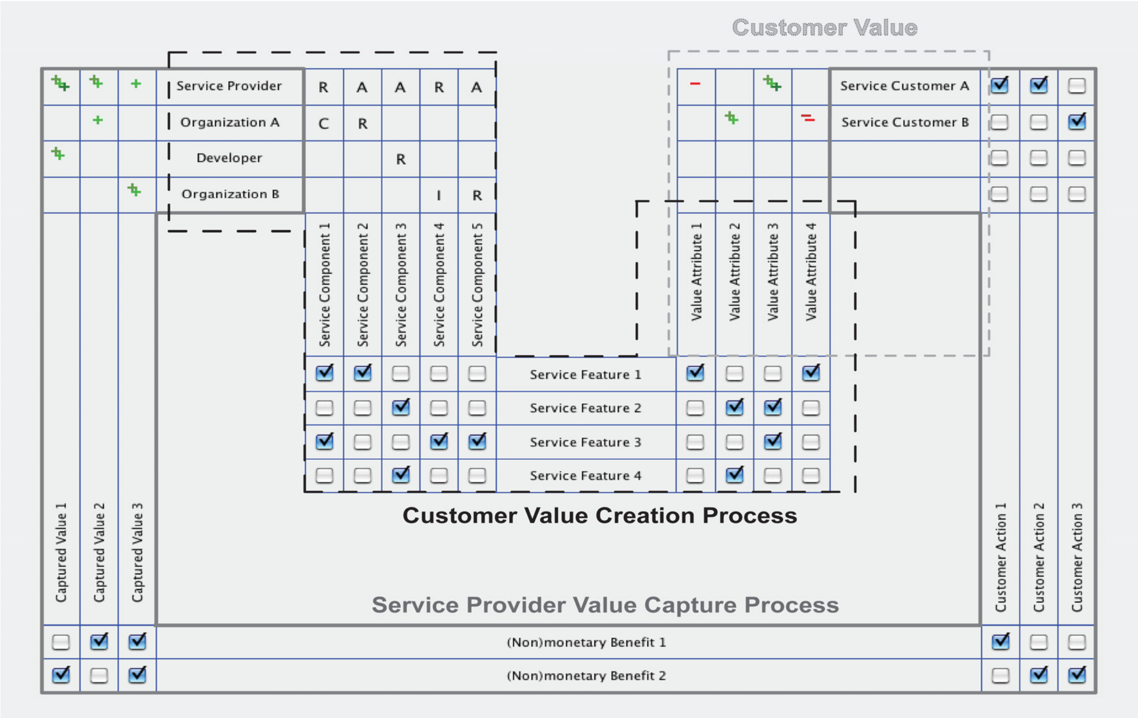


Figure 3: The value map.

contribution to the service in terms of the *Service Components* they provide. We can also see the *Value Captured* by each of the entities in the value network. As illustrated in Figure 2, the *Service Provider* collaborates with *Organizations A and B* and the *Developer* to create the *Service*. This collaboration is captured in terms of the *Service Components* each of these entities provides. Finally, in the Service System Model we represent the *Service Customers* by modeling the *Attributes* that impact their perception of the service value and the *Actions* the customers take on the basis.

A generic Value Map is illustrated in Figure 3. As marked in Figure 3, the Value Map embodies customer value, customer value creation and service provider value capture processes (i.e., the three categories of concepts presented in Section 2) in a service system. This is achieved by making the relationships between the actors (i.e., service provider, organizations in the value network, service customer, etc.) and the properties (service components, service features, value attributes, customer actions, etc.) presented in the Service System Model explict. In Table 3, we explain the relationships and their notation in the Value Map.

To map the *Service Provider* and the other entities in the *Service Provider Value Network* to the *Service Components* we use the RACI (Responsible, Accountable, Consulted, Informed) Matrix. As illustrated in Figure 3, the *Service Provider* is responsible for *Service Component 1* and consults *Organization A*. This consultation may reduce the risk of incompatibility between the *Service Components 1 and 2* or ensure the existence of a contingency plan in case an unanticipated scenario arises in the value creation process.

Table 3: Relationships and mappings in the Value Map.

Relationship	Mapping Notation
- Entities in the value network - Service Components	RACI Matrix
- Service Components - Service Features	☑
- Service Features -Value Attributes	☑
-Value Attributes - Net Perceived Customer Value	+++ Strong Positive --- Strong Negative
- Customer - Customer Actions	☑
- Customer Actions - (Non)monetary Benefits	☑
- (Non)monetary Benefits - Captured Value	☑
- Captured Value - Service Provider's Net Captured Value	☑

The *Service Provider* also contributes to the service through *Service Component 4*. Here, the *Service Provider* makes sure that *Organization B* is kept informed about the progress. The *Service Component 4* provided by *Organization B* may be affected by *Service Component 5* which is provided by the *Service Provider*. Note that these two service componets create the *Service Feature 3*. This sheds light on why *Organization B* needs to be kept up-to-date. In principle, the *Service Provider* is accountable for correct and thorough provisioning of the service components for which other entities are responsible.

As discussed in the previous section, *Service Components* create the *Service Features* that impact the net perceived customer value (NPCV) through the *Value Attributes*. Based on his or her perception of the value of the service offering, the customer takes *Actions*. These *Actions* generate the *(Non)monetary Benefits* for the *Service Provider Value Network*. These benefits are directly linked to the *Value Captured* by each of the entities in the value network. In Figure 3, the *Service Provider* and *Organization A* provide *Service Components 1 and 2* respectively. These two components will create the *Service Feature 1* that negatively impacts the NPCV for *Service Customer A and B* through *Value Attributes 1 and 4*. As shown, the impact is stronger for the *Service Customer B*. Similarly, the *Service Provider* and *Organization B* provide *Service Components 4 and 5* respectively, thereby creating *Service Feature 3*. This service feature has a strong positive impact on the *Service Customer A*'s perception of the service value as captured in Value *Attribute 3*. *Service Customer A* takes *Customer Actions 1 and 2* that contribute to the *(Non)monetary Benefits 1 and 2* thereby realizing and contributing to *Captured Value 1 - 3* for the entities in the *Service Provider Value Network.* As shown, Captured Value 1 has a strong and medium positive impact on the net captured value of the Service Provider and Developer respectively. Other sections of the Value Map can be interpreted the same way.

4 MODELING VALUE CREATION AND CAPTURE IN WEBDOC

In this section, we report on the application of the Value Map as a diagnostic tool to improve value creation and capture in the service system of Webdoc. First we present some information about Webdoc and the motivations underlying the project in which the Value Map was applied. Next, we discuss how customer value attributes were surfaced by means of the data capture and user intelligence tools. Then, we model the creation and capture of value in Webdoc using the Value Map. Finally, we present some strategy implications based on the findings from our modeling process.

4.1 Webdoc

Webdoc is an Internet startup founded in Lausanne in 2009. It currently has offices in Lausanne (headquarters: management, engineering, design, and product), London (business development), Lima (community engagement and support), and San Francisco (business development). Webdoc provides a social network platform on which users can express themselves in a richer, more interactive way than traditional social networks. Specifically, it provides a channel in which existing web content, be it video, audio, images, or text, can be combined with content created using the proprietary rich editor, in a way that requires no technical skills and is easy to share and distribute. These creations are referred to as "webdocs" and can be embedded on any third-party site, including other websites and social networks. Additionally, all webdocs created can be showcased in their relevant category of interest on the Webdoc destination site. The creators have the option to make their webdocs completely private (only users granted explicit permission can view) or public but unlisted (meaning the webdoc will not be featured on the Webdoc site). The service is free to all users with no advertising, currently available in 5 languages (English, French, Spanish, Portuguese, and Russian), and accessible through a variety of platforms including desktop web browsers, mobile device browsers, and native mobile applications.

As the company and user base has grown tremendously in the past 12 months, there has been an increasing need for establishing a better understanding of and improving perceived customer value. The analysis, conception, and subsequent improvement of the value perceived by the customers feed into vital functions of the service and company, including product development, overall strategy, valorization of the company for current and future investment rounds, and optimization of the service. These needs are what triggered the work that has led to the culmination of this project. In the next sections we explain how the value attributes were surfaced and how Value Map improved value creation and capture in Webdoc's service system.

4.2 Surfacing Customer Value Attributes

One of the main challenges in modeling value creation and capture in service systems is surfacing the customer value attributes. This is considered as an important initial step to gain insights into the customers' perceived benefits and costs of adopting the services offered by a service provider. In the context of the project conducted at Webdoc, this step was further sub-divided into two distinct but strongly interconnected fields: data capture and user intelligence.

4.2.1 Data Capture

Broadly speaking, information on customers' perceptions of value and their relative importance can be gathered through direct interaction with customers or customer surveys. Revealed preference methodologies (Carson et al., 1996) are also used to understand customers' needs and preferences based on their behavior. However, for Internet-based services, the channels through which the service provider can understand its users are very different than those of a traditional service. The overwhelming difference is the radically new interaction paradigm through which service providers and service adopters communicate. For traditional service providers, a wealth of customer data, such as customer demography, is gathered without any explicit effort, simply by the customer's physical presence. On the other hand, for an Internet firm like Webdoc, sophisticated measures need to be put in place to understand even the most fundamental characteristics of its users, such as location, language, gender, and age. Without the application of data capture tools it would almost be impossible to answer basic questions such as "Who are the service customers?" "How frequently do they use the service?" "How do service customers access the service?" "How much do they use the service for?" To answer such questions, a number of service providers offer web analytics packages. These are third party, off-the-shelf solutions that can be customized to varying degrees, and are provided for a cost ranging from free to tens of thousands of dollars a month. There also exists the possibility for every Internet company to custom-build its own web analytics and data capture solution. In the context of this project, the latter was the first solution considered, but was quickly discarded due to its infeasibility.

4.2.2 User Intelligence

Data capture contributes to the decision processes in Internet-based services by providing macro-level information. User intelligence tools, however, provide a much more nuanced perspective at the micro level, which sacrifices on breadth of data for depth. The fundamental motivation of the application of user intelligence was the need for product development insight. While numeric metrics such as overall visitors, logged in users, views of a particular page, and so on are certainly invaluable, they are more useful in measuring the effectiveness (or ineffectiveness) of a feature post-change than they are in suggesting what changes might be needed in the first place. Thus, user intelligence provides data that is more prescriptive. This data is complementary to the descriptive data derived from the data capture tools.

User intelligence applications offer various analytical and intelligence tools such as *heat maps* and *user recordings*. Heat maps are screenshots of the website showing the spatial distribution of clicks over the screen space that offer important product insight, as they show what links and content garner the highest level of attention from the audience. User recordings are an attempt to recreate individual user sessions by aggregating mouse movement, keyboard activity, scrolling and navigation, and clicks into a video.

Some advanced user intelligence applications provide the possibility of creating a test environment in which a random sample of participants execute tasks that are predefined based on the demographic and technical requirements. Upon completion of the tasks, a questionnaire is automatically generated, which is filled out by the participant. The key aspect is that while performing the tasks, the entire user screen is recorded, along with an audio stream for the live commentary of the participants. The application of data capture and user intelligence tools provided invaluable assistance in surfacing the customer value attributes.

4.3 Modeling Value Creation and Capture in Webdoc

In this section, we apply the Value Map to represent value creation and capture in Webdoc's service system. To this end, first we shed some light on how Webdoc can capture value as a service provider. Next, we analyse the value for Webdoc's customers. Finally, we show how the Value Map resulted in improving value creation and capture in Webdoc.

4.3.1 Value Capture by Webdoc

Internet-based companies, in particular, social networking platforms such as Webdoc follow a free business model (Osterwalder and Pigneur, 2010). This means these companies do not charge the customers for service they offer. Thus, to sustain their existence these service providers rely solely on the non-monetary benefits from their customers. These non-monetary benefits in the case of Webdoc include: number of users, volume of activity per user, and time spent on the platform per user. Such non-monetary benefits can result in value capture for Webdoc by:

- *Increasing the valuation of the company in case of an initial public offering (IPO) or acquisition*. As an example, Instagram, the online photo sharing service provider, was acquired by Facebook in April 2012 for $300 million in cash and 23 million shares of common stock. The deal was worth $1 billion at the time. Before the acquisition, Instagram announced that more than 5 billion photos had been shared through its mobile apps (Indvik, 2012).
- *Securing funding by venture capitalists (VCs).* Most start-ups rely on funds from external sources such as VCs (Bhide, 2000). The non-monetary benefits listed above are among the determining factors for VCs to make a decision to invest or to continue investing in a start-up company like Webdoc.
- *Monetization through advertisement.* Another possibility for Internet-based services is generating revenues by authorizing the presence of advertisements on their webpages. Advertisement-based monetization is one of the main revenue streams for internet-based service providers. The number of visitors, their activity volume and the time they spend on a website are the main criteria for businesses or individuals to choose a website on which they place their advertisements.

4.3.2 Value for Webdoc's Customers

To improve the NPCV, first an understanding of different customer categories of Webdoc needs to be established. Two main categories of customers are identified: first-time and return visitors. When a first-time visitor uses Webdoc's services again, he becomes a return visitor.

The return visitors are divided into two main categories: content creators and content consumers.

- *Content creators.* This category of customers creates or curates the content on Webdoc. Curation is the process of sorting content created by others on the web and presenting it in a meaningful and organized way.

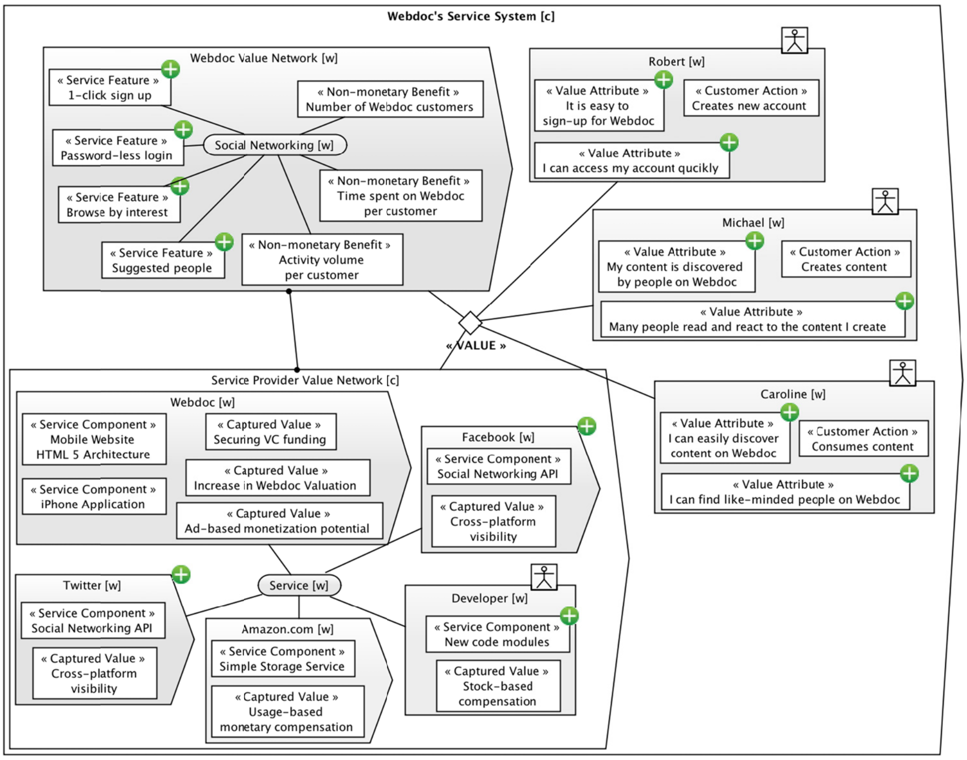


Figure 4: Webdoc's service system, new actors and properties.

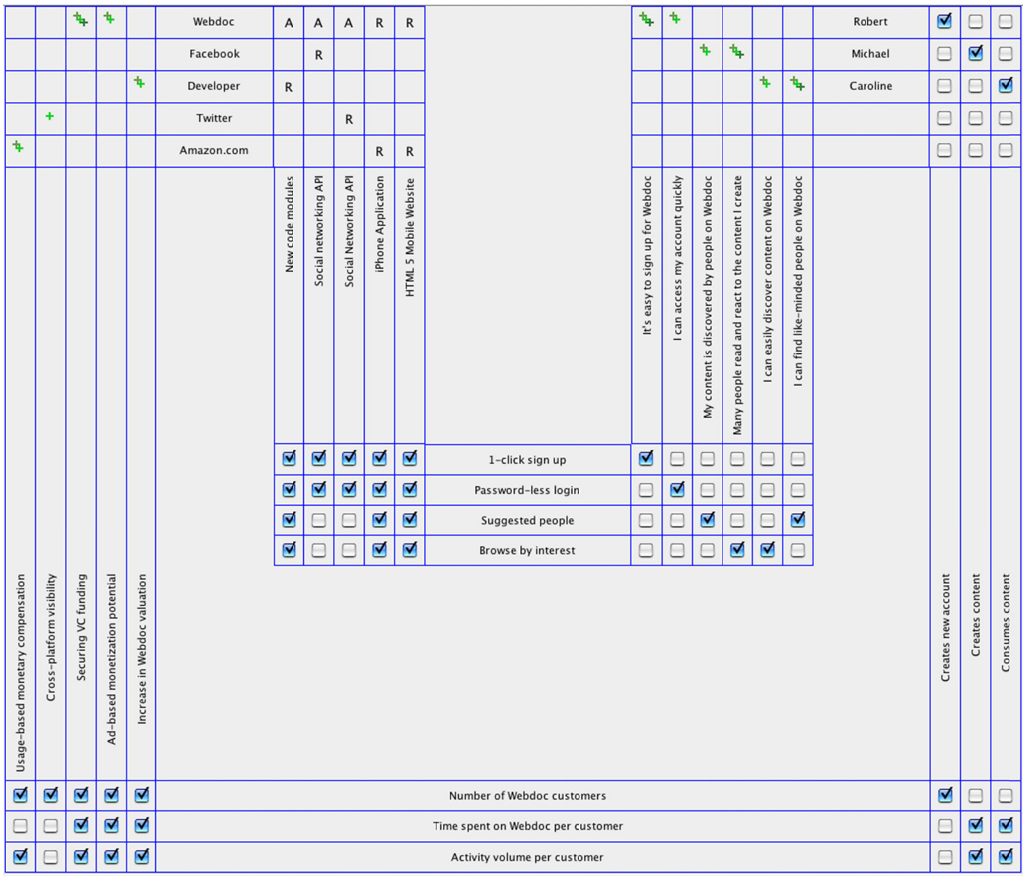


Figure 5: The Value Map: capturing value creation and capture in Webdoc's service system.

- *Content consumers.* These customers consume and react to the contents created by others. The reaction takes place in form of *sharing, liking,* or *commenting on* the contents.

It should be stated that these customer categories are not mutually exclusive.

As discussed in Section 2, the NPCV is impacted by the benefits and the costs of the service. As Webdoc offers its service to the customers free of charge, it would have been intuitive to focus on the service benefits to improve the NPCV. However, the findings from the data capture and user intelligence step revealed a number of improvement opportunities concerning the non-monetary costs of the service. As outlined in Table 2, these non-monetary costs include but are not limited to time, energy, and psychic aspects of adopting a service. Time cost is the aggregate of the durations the service customer needs to invest in order to be able to use the service. Energy cost is the sum of the effort that needs to be spent. Psychic cost is the most abstract of all – the cognitive stress undergone by the customer in using the service.

Data gathering and user intelligence aided us in mapping these three categories of non-monetary costs onto their corresponding value attributes (i.e., non-monetary costs of the service):

- Filling out the sign up form and discovering content and people increased the time and energy costs of the Webdoc's customers.
- Remembering passwords incurred Webdoc customers with high psychic costs.

4.3.3 Improving Value Creation and Capture in Webdoc

Figure 4 shows Webdoc's service system. The new actors and their properties are marked with the plus sign. In the Value Map in Figure 5, we illustrate how these novelties result in improving value creation and capture in Webdoc service system. We explain these changes in the following sections.

4.3.3.1 Reducing Clicks to Signup

Initially, an unregistered visitor needed to search for the sign up button, click it, and then fill out a form to complete the process. To save the visitor's time, the application programming interfaces (APIs) from social networking websites, Facebook and Twitter were integrated in the home page of Webdoc as shown in Figure 6. This way, the first-time visitors could sign up with one click without filling out the sign up form. The return visitors could also use their Facebook or Twitter credentials to connect to Webdoc.

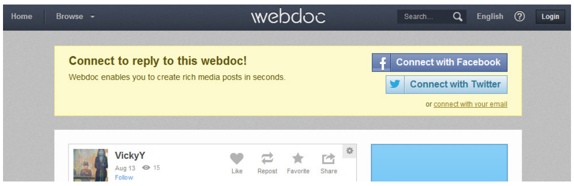


Figure 6: Reducing clicks to sign up.

As illustrated in the Value Map in Figure 5, Twitter and Facebook provide Webdoc with the APIs as the service components. These APIs along with the *New code modules* provided by the *Developer* result in the service features *1-click sign up* and *Password-less login*. These two new features create the following two value attributes for Robert who is a first-time visitor: *It's easy to sign up for Webdoc* and *I can access my account quickly*. These features along with the rest of the benefits of Webdoc convince Robert to *create a new account* (i.e., customer action) thereby increasing *number of Webdoc customers* increases. The rise in the number of customers contributes to *Increase in Webdoc valuation, Ad-based monetization potential* and *Securing VC funding*. As stated in Section 4.3.1, these are the main ways Webdoc can capture value. We can also see that Ad-based monetization potential is not as important as the other two value attributes. The number of Webdoc customers also gives Twitter and Facebook *Cross-platform visibility,* which can contribute to their web presence.

The two new service features improved the NPCV by reducing the time and energy costs associated with filling out the sign up form as well as the psychic costs of the remembering passwords. Introduction of these features increased the number of new accounts created on Webdoc. Moreover, nearly two months after their implementation, over 80% of the users were logging in to Webdoc using their Twitter and Facebook accounts.

4.3.3.2 Welcome Workflow

To facilitate discovering content and content creators, a welcome workflow was designed, see Figure 7.

As shown in the Value Map, the welcome workflow, captured in the *New code modules* service component, resulted in the creation of two new service features: *Browse by interest* and *Suggested people*.

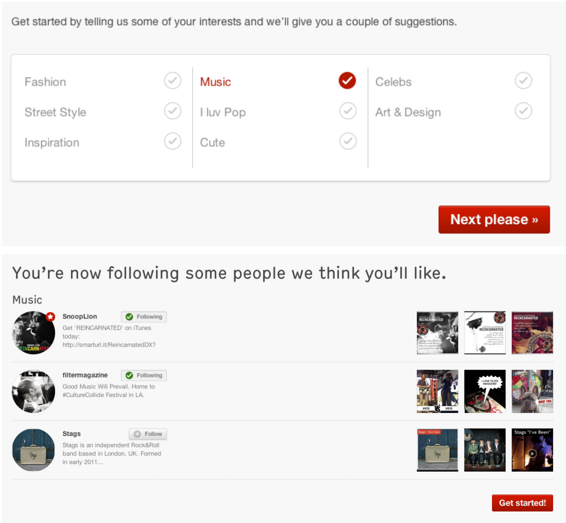


Figure 7: Welcome workflow.

Caroline is a content consumer. The two features help her in *finding like-minded people* and *discovering content on Webdoc*. Michael, a customer who creates content on Webdoc, benefits from these two features as *his content is discovered by people on Webdoc* and *many people read and react to the content he creates.* These value attributes form a self-reinforcing positive feedback loop. Michael creates content, which is discovered by Caroline. Caroline consumes Michael's content and reacts to it by commenting or liking or reposting his content. This motivates Michael to create or curate even more content. This virtuous cycle increases *time spent on Webdoc per customer* and *activity volume per customer.* These two non-monetary benefits contribute to Webdoc's value capture the same way as the *number of Webdoc customers.* When Webdoc's valuation increases some value is also captured by the developer who receives stock-based compensation. Finally, operating on the basis of a pay-per use pricing mechanism, Amazon.com also captures some value when the number of the customers and the activity per customer increase.

The new features created by the welcome workflow reduce the energy and time costs pertinent to discovering people and content. The introduction of these two features contributed to the 250% increase in the log in rate of the Super Users, those customers of Webdoc who visit the website at least three times a week.

4.4 Strategy Implications for Webdoc

Over the past few months, Webdoc has improved the value its customers perceive from the services it offers, by reducing the non-monetary costs associated with its services. These improvements have resulted in an increase in the number of customers, the activity and the time spent by each customers. However, similar to any growth pattern there are limits to this growth.

We suggest that Webdoc should also become a platform to promote the work of the artists who are not famous. A young Sci-Fi writer, an unknown musician or a semi-professional painter can be the potential new actors in Webdoc's service system. These people should not be merely seen as customers. In fact, they should be taken into account as parts of the Webdoc's value network. Expanding the value network results in the creation of a bigger pie for all the organizations and people involved and results in creating more value for customers.

5 RELATED WORK

The Value Map is an extension to the SAR (Supplier Adopter Relationship) Diagram in (Golnam et. al, 2010; 2011; Wegmann et al., 2007). The Service System Model is based on the System Diagram (Rychkova et al., 2007). The SAR and the System Diagram are parts of the Systemic Enterprise Architecture Methodology (SEAM) (Wegmann, 2003).

SEAM was designed from the ground up with general systems principles and serves to analyze and to assist in the design of business and engineering strategies. Developed at Ecole Polytechnique Fédérale de Lausanne (EPFL), SEAM has been used for teaching (Wegmann et al., 2007) and consulting (Wegmann et al., 2005).

In developing the Value Map, we are also inspired by the House of Quality (Clausing and Hauser, 1988), a quality improvement method, derived from Quality Function Deployment (QFD). We integrate the Strategy Canvas (Kim and Mauborgne, 2005) as a part of the Value Map. Strategy Canvas is a diagnostic framework for strategy development. It enables an organization to visualize the competitive factors and the current state of play of those factors within a market place and to compare the organization's offering with those of the industry in general.

Business Model Canvas (Osterwalder and Pigneur, 2010) is a strategic management tool, that assists in the development of new, and improvement of existing business models. It is widely recognized as one of the most established methods for business model design and innovation. The canvas represents value creation and capture in business models by nine building blocks: key partners, key activities, key resources, value propositions, customer relationships, channels, and customer segments. Business Model Canvas is one of the most established methods in the academia and industry for business model design, development and improvement.

To evaluate the usefulness of the Value Map we conducted an empirical study in which we also compared the Value Map with Business Model Canvas. The study was conducted in form of three workshops with 14 senior managers and executives from a range of industries in Iran. In the first workshop we presented the theoretical and conceptual discussions underlying problem structuring and business modeling. Next, we familiarized the participants with Business Model Canvas and presented an example illustrating its application in business modeling. In the second workshop, we introduced the Value Map. At the end of the second workshop, the participants filled out a survey questionnaire. The questions were divided into three categories: the importance of value creation and capture in business models, the potential merits of modeling value creation and capture with the Value Map and comparison of the Value Map with the Business Model Canvas.

The results reflected that the participants believed that Value Map helps business practitioners understand and analyze customer value, customer value creation, and the value capture processes. Based on the results, this is achieved by creating a common language that enables the representation and the discussion of the as-is and to-be situation of value creation and capture in an organization's business model. The results in general suggest that Value Map is a useful visualization tool that contributes to managerial decision-making processes of business practitioners in the choice situations that entail value creation and capture in an organization's business model. We learned that the Value Map complements and augments the Business Model Canvas by aiding the business practitioners in representing the necessary building blocks of business model of an organization and their inter-relations and interconnectedness.

A week after the second workshop, we held the third workshop with the participants to debrief them on the application, the potential merits and the improvement opportunities with respect to the Value Map. Based on the discussions, we drew the conclusion that the Strategy Canvas can be used as an input to the Value Map in designing the value creation and capture processes in a business model. These discussions also revealed a number of improvement opportunities, such as quantification of the qualitative concepts, improving the graphical representation of the Value Map.

Some of the improvement opportunities mentioned by the participants are already taken into account in the instantiation of the Value Map in the www.tradeyourmind.com online platform. For instance, the inclusion of the quantitative models that can generate numerical analyses of various value creation and capture strategies is part of the platform. The step-by-step model generation wizard embedded in the www.tradeyourmind.com platform also facilitates the development and the presentation of the Value Map. We will try to address the remaining points in our future work.

6 CONCLUSIONS

In this paper, we introduced the Value Map as a problem structuring method (PSM) that aids in conceptualization and representation of value creation and capture in service systems. The Value Map is grounded in the theoretical insights from economics, management science and (services) marketing literature, drawing principally upon work from the past two decades on value creation and capture, including theories, frameworks, constructs, and other models. We illustrated the usability and applicability of our framework by modeling value creation and capture in Webdoc's service system. We also briefly presented the results of a survey conducted to assess the usefulness of Value Map and compare it with Business Model Canvas.

This research suffers from a number of limitations. We used data synthesized in a single case study to illustrate the applicability of the Value Map. Despite the fact that the data for the case study was gathered from a project we conducted in a company, we believe we need to apply the Value Map in several other contexts to be able to fully assess the practical relevance of its representations. Thus, in our future work we will focus on applying the Value Map in a number prospective business cases. This will definitely result in a better evaluation of the applicability of the Value Map.

The second limitation of this research concerns the empirical study we conducted to evaluate the usefulness of the Value Map. The fact that all the participants in the survey were from Iran and the relatively small sample size limit the generalizability of the findings of our research. To tackle this limitation, the same study should be conducted among executives and managers from different countries.

Lastly, the articles based on which the conceptualizations underlying the Value Map were developed are not exhaustive. Despite the fact that we synthesized over 30 well-cited articles on value creation and capture that were to the best of our knowledge seminal to the field, some relevant work still may not have been included in the review of the literature. Inclusion of such articles can bring in new modeling constructs or fine-tune and improve the existing constructs in the Value Map. Refining our conceptualizations based on the existing work that has not been included in the study will also be a part of our future work.

REFERENCES

Allee, V., 2008. Value network analysis and value conversion of tangible and intangible assets. *Journal of Intellectual Capital, 9* (1), 5-24.

Baker, M. J., 2000. Writing a literature review. *The Marketing Review, 1* (2), 219-247.

(n.d.).

Ballantyne, D., Varey, R. J., 2006. Creating value-in-use through marketing interaction: the exchange logic of relating, communicating and knowing. *Marketing Theory, 6* (3), 335-48.

Beer, S., 1995. *The heart of enterprise.* New York: John Wiley & Sons.

Bhide, A., 2000. *The origin and evolution of new businesses*: Oxford University Press on Demand.

Blomqvist, K., Kylaheiko, K., Virolainen, V. M., 2002. Filling a gap in traditional transaction cost economics: Towards transaction benefits-based analysis. *International Journal of Production Economics, 79* (1), 1-14.

Bowman, C., Ambrosini, V., 2000. Value Creation Versus Value Capture: Towards a Coherent Definition of Value in Strategy. *British Journal of Management, 11* (1), 1-15.

Carson, R. T., Flores, N. E., Martin, K. M., Wright, J. L., 1996. Contingent valuation and revealed preference methodologies: comparing the estimates for quasi-public goods. *Land Economics*, 80-99.

Clausing, D., Hauser, J. R., 1988. The house of quality. *Harvard Business Review,* 66, 63-73.

Golnam, A., Regev, G., Ramboz, J., Laparde, P., Wegmann, A, 2010. Systemic Service Design: Alaigning Value with Implementation. *In:* Snene, M., Ralyté, J. & Morin, J.-H., eds. 1st International Conference on Exploring Services Sciences, Geneva, Switzerland. Springer, 150-164.

Gordijn, J., Akkermans, J., 2003. Value-based requirements engineering: Exploring innovative e-commerce ideas. *Requirements Engineering, 8* (2), 114-134.

Grönroos, C., 2006. Adopting a service logic for marketing. *Marketing Theory , 6* (3), 317-33.

Grönroos, C., 2008. Service logic revisited: who creates value? And who co-creates? *European Business Review, 20* (4), 198-314.

Grönroos, C., 2000. Service Management and Marketing. *European Journal of Marketing, 15*, 3-31.

Grönroos, C., 1979. *Service marketing: a study of the marketing function in service firms (In Swedish with an English summary).* Diss. Helsinki and Stockhold, Hanken Swedish School of Economics, Marketing Technique Centre and Akademilitteratur, Helsinki.

Grönroos, C., Ravald, A., 2011. Service as business logic: implications for value creation and marketing. *Journal of Service Management, 22* (1), 5-22.

Gummesson, E., 2007. Exit services marketing - enter service marketing. *Journal of Customer Behaviour, 6* (2), 113-41.

Hart, C., 1999. *Doing a literature review: Releasing the social science research imagination*: Sage Publications Limited.

Indvik, L., 2012. Facebook Completes $730 Million Instagram Acquisition Retrieved May, 2013, from http://mashable.com/2012/09/06/facebook-instagram-acquisition-complete/

Kim, W. C., Mauborgne, R. E., 2005. *Blue Ocean Strategy: How to Create Uncontested Market Space and Make Competition Irrelevant*, Harvard Business Press.

Kothandaraman, P., Wilson, D. T., 2001. The Future of Competition: Value-Creating Networks. *Industrial Marketing Management, 30* (4), 379-389.

Kotler, P., 2000. *Marketing management, millennium ed.*

Lepak, D. P., Smith, K. G., Taylor, M. S., 2007. Value Creation and Value Capture: A Multilevel Perspective. *Academy of Management Review, 32* (1), 180-194.

Lusch, R. F., Vargo, S. L., 2006. *The Service-Dominant Logic of Marketing: Dialog, Debate, and Directions.* Armonk, New York, USA: M. E Sharpe.

Masten, S. E., Meehan, J. W., Snyder, E. A., 1991. The costs of organization. *Journal of Law, Economics, and Organization, 7* (1), 1.

Mingers, J., Rosenhead, J., 2004. Problem structuring methods in action. *European Journal of Operational Research, 152* (3), 530 - 554.

Moller, K., Svahn, S., 2006. Role of Knowledge in Value Creation in Business Nets*. *Journal of Management Studies, 43* (5), 985-1007.

Nelson, R. R., Winter, S. G., 1982. *An evolutionary theory of economic change*: Belknap press.

Osterwalder, A., Pigneur, Y., 2010. *Business model generation: A handbook for visionaries, game changers, and challengers*, New York: Wiley.

Parasuraman, A., Zeithaml, V. A., Berry, L., 1985. A Conceptual Model of Service Quality and Its Implications for Future Research. *Journal of Marketing, 49*, 41-50.

Pijpers, V., Gordijn, J., 2007. e 3 forces: Understanding Strategies of Networked e 3 value Constellations by Analyzing Environmental Forces *Advanced Information Systems Engineering*, 188-202.

Prahalad, C., Ramaswamy, V., 2004. Co-creation Experiences: The Next Practice in Value Creation. *Journal of Interactive Marketing, 18* (3).

Pynnonen, M., Ritala, P., Hallikas, J., 2011. The new meaning of customer value: a systemic perspective. *Journal of Business Strategy, 32* (1), 51-57.

Ritala, P., Hurmelinna-Laukkanen, P., 2009. What's in it for me? Creating and appropriating value in innovation-related coopetition. *Technovation, 29* (12), 819-828.

Rosenhead, J., 1999. What's the problem? An introduction to problem structuring methods. *Interfaces*, 26, 117-131.

Rosenhead, J., Mingers, J., 2001. *Rational analysis for a problematic world revisited.* Chichester: John Wiley and Sons.

Rychkova, I., Regev, G., Le, L.-S., Wegmann, A., 2007. *From Business To IT with SEAM: the J2EE Pet Store Example.* Paper presented at the Enterprise Distributed Object Computing Conference, 2007. EDOC 2007. 11th IEEE International.

Sandström, S. E., Kristensson, P., 2008. Value in use through service experience. *Managing Service Quality , 18* (2), 112-126.

Sphorer, J., Maglio, P., 2008. The Emergence of Service Science: Toward systematic service innovations to accelerate co-creation of value. *Production and operations management*.

Storbacka, K., Lehtinen, J., 2001. Customer Relationship Management: Creating Competitive Advantage Through Win-win Relationship Strategies.

Ulaga, W., 2003. Capturing value creation in business relationships: a customer perspective. *Industrial Marketing Management, 32* (8), 677-693.

Vargo, S. L., Lusch, R. F., 2004a. Evolving to a New Dominant Logic of Marketing. *Journal of Marketing , 68* (1), 1-17.

Vargo, S. L., Lusch, R. F., 2008. Service-Dominant Logic: Continuing the Evolution. *Journal of the Academy of Marketing Science, 36* (1), 1-10.

Vargo, S. L., Lusch, R. F., Akaka, M. A., 2010. Advancing Service Science with Service-Dominant Logic - Clarifications and Conceptual Development. *Handbook of Service Science*, 133-156.

Vargo, S. L., Maglio, P. P., Akaka, M. A., 2009. On Value and Value Co-creation: A Service Systems and Service Logic Perspective. *European Management Journal, 26*, 145-152.

Wegmann, A., 2003. *On the systemic enterprise architecture methodology (SEAM).* Paper presented at

the International Conference on Enterprise Information Systems, Angers, France.
Wegmann, A., Regev, G., Loison, B., 2005. Business and IT alignment with SEAM. In *Proceedings of the 1st International Workshop on Requirements Engineering for Business Need, and IT Alignment*, Paris, France.
Wegmann, A., Julia, P., Regev, G., Perroud, O., Rychkova, I., 2007. *Early Requirements and Business-IT Alignment with SEAM for Business*. Paper presented at the 15th IEEE International Requirements Engineering Conference (RE'07), Delhi, India.
Weigand, H., 2009. Value EncountersñModeling and Analyzing Co-creation of Value. *Software Services for e-Business and e-Society*, 51-64.
Weigand, H., Johannesson, P., Andersson, B., Bergholtz, M., 2009. *Value-based service modeling and design: Toward a unified view of services*.
Whetten, D. A., 1989. What constitutes a theoretical contribution? *Academy of Management Review, 14* (4), 490-495.
Yu, E. S. K., 1997. *Towards modelling and reasoning support for early-phase requirements engineering*. Paper presented at the The Third IEEE International Symposium on Requirements Engineering.

XChor

Choreography Language for Integration of Variable Orchestration Specifications

Selma Suloglu[1], Bedir Tekinerdogan[2] and H. Ali Doğru [1]
[1]*Computer Engineering Department, Middle East Technical University, Dumlupınar Boulevard, Ankara, Turkey*
[2]*Computer Engineering Department, Bilkent University, Ankara, Turkey*
{Selma, dogru}@ceng.metu.edu.tr, bedir@cs.bilkent.edu.tr

Keywords: Service-Oriented Architecture, Choreography, Choreography Languages, Variability Management, Variability Metamodel.

Abstract: In this paper, we propose and develop a new choreography language XChor which can be used to support variability in choreography specifications and integrate these with variability of orchestration specifications. We describe the metamodel of XChor and illustrate the adoption of the language by specifying user verification choreography in the adaptable security system. Orchestration and choreography models are mechanisms to realize service composition and coordination while some of them support variation to deal with reuse challenge. Several approaches have been introduced to support variability in orchestration and choreography languages. Unfortunately, variability is not explicitly addressed in current choreography languages. As such, it is hard to provide a consistent configuration of service composition within and across business organizations.

1 INTRODUCTION

Several organizations develop, share and reuse business processes by establishing collaboration with other organizations in order to fulfill different stakeholder needs. Being agile is an important challenge in business process integration context which requires a dynamic environment. Service-oriented architecture (SOA) is a promising approach to realize such environments by designing and developing distributed systems (Erl, 2005). SOA aims to facilitate reuse of services and incorporates service consumers and service providers. A service is self-contained, and can be independently deployed in a distributed component.

Building enterprise solutions to realize business processes typically requires the composition of multiple existing enterprise services. Composite services can be further recursively composed with other services to derive higher level solutions. Two different types of service compositions are defined: 1: service choreography where the interaction protocol between several partner services is defined from a global perspective. 2: service orchestration, where the interaction logic is specified from the local point of view of one single participant, called the orchestrator.

Reuse in SOA can be achieved by managing variability in different granularity levels, namely choreography, orchestration and atomic services. Assuming that all granularity levels can be treated as services, variability can come from (i) their interfaces (functions and parameters), (ii) connectors (the way they interact) and (iii) composition (the way they are gathered in order to achieve a goal). Interface variability requires a configuration mechanism specifying when and how to change its functions and parameters. Connector variability needs a relation mechanism to indicate when and which connector is used between two services. Composition variability necessiates a tailoring mechanism to define in which order and how services are interacting with each other. Services offer different functionalities regarding their variability bindings. Therefore, it is the composition's responsibility to provide a consistent variability binding between interacting services. This requires a mechanism to establish variability associations which determines when and how interacting services bind to specific variants. In other words, composition is responsible for handling consistent variability binding of interacting services and providing a configuration infrastructure.

Addressing and fulfilling all these variability needs to provide seamless integration of services. To cope with such challenges several approaches have been introduced. However, explicit introduction of variability integrated with choreography languages is not addressed. Specification of consistent variability binding and configuration of interacting services are not considered in the choreography language level. Moreover, there is a lack of support to reuse existing choreographies.

In this article, we first analyse and discuss existing orchestration and choreography languages with respect to variability management. We identify the problems and the requirements for variability in choreography languages. To support interface and composition variability in choreography specifications we developed a new domain specific language called XChor.

The remainder of the paper is organized as follows. Section 2 firstly describes variability management in existing choreography and orchestration languages. Then the requirements for managing variability in choreography languages are defined and problems in existing languages are stated. Section 3 introduces the metamodel developed by authors for supporting variability in choreography specifications and integrating these with variability of orchestration specifications. Section 4 describes the XChor language and demonstrates its usage through an example. Finally section 5 provides the conclusions.

2 VARIABILITY MANAGEMENT IN EXISTING ORCHESTRATION AND CHOREOGRAPHY LANGUAGES

Obviously for small systems we could handle orchestration specifications using traditional approaches such as interaction diagrams. Variable parts and their relations can be modeled and implemented by data and through 'if' control structures. However, for integration of large scale systems soon the traditional approaches are less expressive and not tractable. Therefore, to cope with variability in choreography, orchestration, and atomic services various language approaches have been introduced. We have listed the popular approaches in Table 1. We evaluate these approaches with respect to the following criterias:

- *Composition Approach:* Defines whether the language supports choreography and/or orchestration. Orch is the abbreviation of orchestration and Chor is that of choreography.
- *Variability Support:* Defines whether the language supports variability. 'Yes' indicates that the language provides explicit language mechanisms for variability. 'Implicit' indicates that although the language does not provide explicit mechanisms, variability is supported implicitly. 'No' means that there is no variability support.
- *Tool Support:* Availability of tools.
- *Modelling Approach:* Defines the adopted modelling approach which can be either based on interaction or interconnection. Modeling based on interaction represents definition of one building block (document or specification) for the whole system, whereas interconnection suggests modeling control flow logic per participant. Intera is the abbreviation of interaction and Interc is that of interconnection.

BPEL (OASIS 2007), VxBPEL (Koning et al., 2009), Jolie (Montesi et al., 2007) and Jorba (Lanese et al., 2010) purely target orchestration as the composition approach and interconnection as the modelling approach. Among them VxBPEL has an explicit variability model. On the other hand, Jorba, a rule-based approach to dynamic adaptation implemented on top of the Jolie language, provides a mechanism without explicit specification of variability.

WSMO (Fensel et al., 2007), BPMN (OMG 2011) and Reo (Arbab, 2004) target orchestration and choreography specification. While WSMO provides an interconnection model, Reo and BPMN include interaction and interconnection models. Among them, Reo offers variability support by hyper-graph transformation.

BPEL abstract processes, WS-CDL (W3C, 2005), Let's Dance (Zaha et al., 2006), MAP (Barker et al., 2009), BPEL4Chor (Decker et al., 2007), and an extension of it – BPEL^{gold} (Kopp et al., 2010) all target choreography for service composition. An interaction modelling approach is followed by WS-CDL, Let's Dance and MAP, whereas interaction model is applied in BPEL abstract processes, BPEL4Chor and BPELgold. Moreover, MAP supports an interaction model by separating choreography definition to peers related with services.

According to Table 1, the languages supporting variability are VxBPEL, Jorba and Reo. VxBPEL language seems to be the only language which provides explicit support for variability mechanisms based on ConIPF Variability Modeling. Framework (COVAMOF) (Sinnema, Deelstra, Nijhuis, Bosch, 2004.). The approach extends BPEL with variability constructs, such as <<VariationPoint>> and <<Variant>>. However, the language does not support variability of choreography. In parallel, there is no mechanism to inspect the global view of variability when more than one VxBPEL orchestration interacts.

Table 1: Comparison of existing orchestration and choreography languages.

	Composition Approach	Variability Support	Tool Support	Modelling Approach
BPEL 2.0	Orch	No	Yes	Interc
VxBPEL	Orch	Yes	No	Interc
Jolie	Orch	No	Yes	Interc
Jorba	Orch	Implicit	Yes	Interc
Reo	Orch Chor	Implicit	Yes	Interc Intera
WSMO	Orch Chor	No	Yes	Interc
BPMN 2.0	Orch Chor	No	Yes	Interc Intera
WS-CDL	Chor	No	No	Intera
Let's Dance	Chor	No	No	Intera
BPEL4Chor BPEL^{gold}	Chor	No	Yes	Intera
MAP	Chor	No	Yes	Interc Intera

On top of the Jolie orchestration language, Jorba defines adaptation interfaces specifying function replacements whenever a change in service interface and parameter is needed. However, the relationship between rules and the coverage of variability is implicit and the management of rules as a separate variability model is usually difficult to manage. There is no mechanism to explicitly specify variation points and variants as in VxBPEL tags.

Reo, a comprehensive approach to service composition proposes a hyper-graph transformation approach to manage change. Services as nodes are connected via edges. In other words, variability is provided by reconfiguration of services which is seen as an internal part of the system. Therefore, there is no explicit variability model defined to intervene and change the composition, accordingly no explicit specification of relations between variability of services taking part in the composition.

2.1 Problem Statement

The analysis of the existing choreography languages shows that variability in both orchestration and choreography is not supported in any of the languages. Besides, interface and composition variability support is not explicitly addressed with a single variability model covering choreography, orchestration and atomic services. Concretely we can identify the following problems:

- *Lack of explicit expressiveness of variability in choreography specifications*

There is no language that explicitly represents variability in choreography in order to integrate orchestration specifications. Moreover, variability modelling in choreography, orchestration and atomic services as a whole is not explicitly covered in one single model. This impedes the consistent configuration of choreography and orchestration specifications with regard to variability.

The lack of explicit abstractions for variability easily leads to the scattering of variability concerns over service compositions. Likewise, enabling or disabling a variability results in reorganization of the composition. This complicates the understanding of variable parts, relations amongst them and the overall goal for business process engineers and developers. Tracing these scaterred variations can be achieved to a certain degree, but in large scale systems traceability and understandability decrease gradually. As a result, this scattering reduces the maintenance of the system.

- *Lack of explicit specification of variability associations between interacting services*

Choreography interrelates a set of orchestrations, atomic services and establishes connection with other choreographies. Interacting services' variability constraints and shapes possible choreography abilities and composition. Likewise, variability of choreography dictates proper service variability bindings and specified configurations resulting service interfaces with different functionality and parameters. In order to reveal these dependencies and relations between choreography and services, an explicit association and mapping should be defined. In other words, configuring choreography requires configuring other services in order to consistently collaborate with each other. Therefore, configuration and binding of service variability requires an integrated model comprising choreography, orchestration specifications, and atomic services with variability. There is no language supporting such integrated configuration model dealt with variability of all granularity levels.

- *Lack of support for reusing existing choreographies*

The importance of reusing existing choreographies is addressed in some approaches, but reusing as a part of the other choreography is not emphasized sufficiently. There are ways to handle choreography-to-choreography relationships such as collaborating via exposed choreography interfaces. In case of variability, it is more difficult to utilize choreography specifications with proper bindings. Therefore, the way to bind to other choreographies should be specified.

Although several choreography languages address some of the above stated concerns, no single orchestration or choreography language covers all of them. Moreover, there is no specified mechanism to associate and map orchestration and choreography variability for consistent integration. Even if variabilities for choreography, orchestration and atomic services are explicitly specified, seamless and consistent mapping cannot be achieved due to different concepts and capabilities of different variability models. In that, one variability model can constrain the other one. For instance, COVAMOF model used in VxBPEL orchestration specification does not have external variation definition and can not be completely mapped with a model providing external variation. Therefore, the modeling of variability in choreography consistent with orchestration and atomic services cannot be achieved easily. To support the systematic management of variability and the consistent composition of choreography specifications, a choreography model that incorporates variability concepts is needed.

3 A METAMODEL FOR VARIABILITY MANAGEMENT IN CHOREOGRAPHY

To enable integration of orchestrations, atomic services in the scope of choreography, we propose a metamodel in which atomic services and orchestrations are evaluated under service concept. The main difference in specification between orchestrated and atomic service comes from revealing external behavior to service environment. That is, orchestrated service can define external interaction with other services if required. Moreover, there is no constraint that an atomic service can not specify its interaction. Therefore, atomic services and orchestrations are treated as services in our metamodel.

The metamodel basically enables to define choreographies and services, to specify variability of each one and to integrate these variabilities in order to provide a consistent collaboration. Figure 1 depicts the overview of service and choreography relations based on our metamodel so as to support interface and composition variability. Two main blocks are depicted; choreography and service.

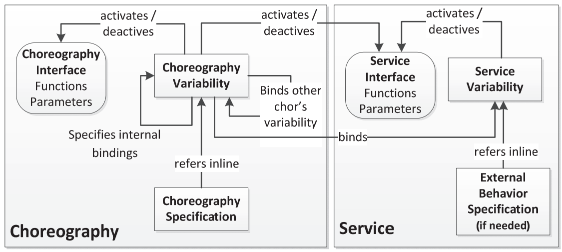


Figure 1: Overview of choreography and service relations based on our metamodel.

Both choreography and service, interfaces without variation are defined fulfilling all possible functional requirements. Choreography interface is only configured with regard to its own variability specification, whereas service interface is configured via both its own variability specification and variability specification of choreography that takes part in. Configuration of service is achieved by activating/deactivating functions and setting/unsetting parameters. With this mechanism, different choreographies utilize different interfaces of the same service which brings service reusability.

Choreography variation leads to proper bindings of variations of other choreography and services via mapping so as to provide interacting interface consistency. Choreography and external behavior specification of services include inline references of their own variability to point out the changeable parts.In this way, choreographies and services include a set of possible required behavior in order to fulfill different composition needs, which enables reuse of choreography and services.

3.1 Variability Specification

The rightmost part of the metamodel in Figure 2 presents the variability specification constructs. This part has been defined based on existing variability metamodels in the literature. A comparative literature study has been conducted in (Lianping et al., 2009). Based on this part of the metamodel, choreography and services can define their internal

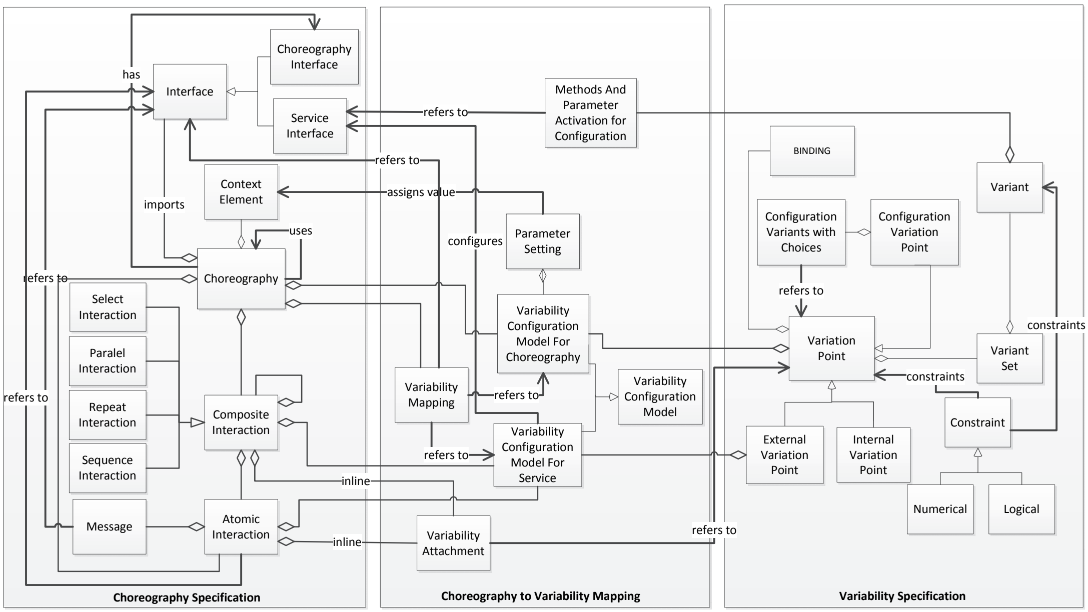


Figure 2: The metamodel for variability management in choreography.

and external variation points, related variants and constraints among them. A special type of variation point, configuration variation point eases management and understanding of variability mechanism by holding details of internal variation point bindings.

For different variation point relationships, contraints provide a mechanism to establish a convenient binding and selection by defining numerical and logical constraints.

3.2 Choreography Specification

The leftmost part of the metamodel represents the elements to define a choreography composition and interfaces of choreography and services. Choreography comprises a set of services and choreographies, identifying composability via service interactions. Service interactions specify the way how the services collaborate which is realized by atomic and composite interactions.

Choreography and service interfaces expose a set of functions without variability specifications. Other than services, a choreography interface states required functions from other services and choreographies.

3.3 Choreography to Variability Mapping

The middle part of the metamodel represents the concepts to define the mapping between choreography and variability constructs. Mainly these constructs are responsible for configuration of interfaces, establishment of variability associations and representing variability references in composition.

Variability configuration model for service and choreography includes a set of variation points, constraints among them and service interactions (for services only). Variability Association facilitates choreography to identify proper bindings of utilized service and choreography variability.

Methods And Parameter Activation for Configuration provides a configuration mechanism to define method activation/deactivation and parameter setting/unsetting of referred service interface.Variability attachment specifies conditions of variation point and variant selections used in choreography composition. Tagging with variability attachment specifications, the parts of the composition gains dynamicity that changes the behavior of choreography. When conditions are satisfied, the part is added to the final composition.

4 XChor LANGUAGE

The authors have developed a new domain specific language, XChor, based on the metamodel that we have described in the previous section. XChor (XChor, 2012) has been implemented using Xtext (Xtext, 2012) in the Eclipse development environment.

XChor Language facilitates to create three different models. Configuration interface models cover variability specifications stated in (Nguyen et al., 2011). Choreography model can specify twelve service interaction patterns described in (Barros et al., 2005).

The basic elements of XChor is shown under three model to cope with variability in choreography. Models are exemplified based on a part of a real life case study, verification of a user in adaptable security system.

Adaptable Security System is an authentication system residing between customers and third party applications or institutions that supports different authentication types of data, including software and hardware (biometric device) parts. The system has the ability to be integrated and applied to a military installation or to a banking system, which requires fulfilling different stakeholder needs. Applicability to different stakeholder systems requires different functionality support and behaviour. User verification can be done offline or online by a third party authority such as web services or certain devices like: PDA, PC, ATM, or mobile phone. The third party authority gets different types of data as required user credentials: (1) username and password, (2) username and password with instant mobile text, (3) e-sign, (4) biometric data; fingerprint, finger vein, and/or iris. Then, according to the online and offline verification result, the system will allow or ban users entering the integrated application.

Device support is important as different devices have different capabilities. ATM, PDA and mobile phone can be used with (1), (2) and (3). PC supports (1), (2), (3) and (4). Therefore, the system should change verification processing functions according to used devices and parameters to be verified.

While modeling this system, user verification is treated as a choreography utilizing other choreographies and services. In the following sections, (i) configuration interfaces for defining and managing variability of choreographies and services, (ii) interfaces of user verification choreography and interrelated services, and (iii) user verification choreography are depicted.

4.1 Configuration Interface

Configuration interface model covers service and choreography variability specifications internally and externally to depict possible abilities, to configure others and to be configured by others. To depict possible abilities; Choreography can specify internal, external and configuration variation points, whereas services can only depict external variation points. The external ones are used to be configured by choreographies and services. Capabilities to configure its own interface or other services' intefaces as activating/deactivating and setting/unsetting parameters are also specified in this model. Numerical or logical constraints among variability specifications are depicted.

Different user authentication types such as biometric authentication, supported authentication modes (online and/or offline), transaction types (real or fake transaction) are the system's behaviours need to be configured differently. Therefore, each is treated as variability in configuration interface of user verification choreography.

To enable authentication variability, both types of authentication and parameters used in encryption function are changed with regard to the usage of biometrics or not. For this purpose a configuration variation point named as "authentication_type" as external and two internal variation points "i_auth_type" and "i_encryption_parameters" are defined. Binding of "authentication_type" configures consistent bindings of "i_auth_type" and "i_encryption_parameters".

"i_auth_type" is specified with "internalVP" keyword (line 5). "username_passw" is a mandatory variant, whereas "onetimepassw" (line 9) and "esign" (line 10) are optional in other words can be selectable. At least one and at most two variants can be selected among the following alternatives: "fingerprint" (line 12), "fingervein" (line 13), "iris" (line 14), and "face" (line 15). The binding time of this variation point is runtime (line 17). "authentication_type" is specified as external (line 26). The variation point has two optional variants specified (lines 29-30); "userinfo" and "biometrics". "userinfo" variant is realized (line 33) by selection of "defaultparams" variant of "i_encryption_parameters" variation point.

For "biometrics", the realization requires two selections at the same time: (i) minimum one variant among "fingerprint fingervein iris face" set should be selected from "i_auth_type" variation point (line 35) and "setparams" variant of "i_encryption_parameters" variation point (line 36). Default variant of the "authentication_type" configuration variation point is "userinfo" (line 37). Configuration type is parameterization and it is bound at development time represented as "devtime" (line 39).

```
1 Configuration interface vconf_verification of choreography userverification
2
3  //determines number of different biometric authentication types
4  @composition
5  internalVP i_auth_type:
6        mandatory
7              variant username_passw
8        optional
9              variant onetimepassw
10             variant esign
11       alternative
12             variant fingerprint
13             variant fingervein
14             variant iris
15             variant face
16             (min:1,max:2)
17       bindingTime runtime
18 //determines authentication mode
19 @composition
20 internalVP i_auth_mode:
21       optional
22             variant mode_online:activateMethods(service:thirdparty,funct:getconnection,savehasheddata,verify)
23             variant mode_offline:activateMethods(service:storage,funct:get_hashed_data)
24       bindingTime devtime
25
26 configuration authentication_type:
27       varType externalVP
28       optional
29             variant userinfo
30             variant biometrics
31       realization "it is realized by i_encryption_parameters and i_auth_type variability points"
32       confvariant userinfo mapping
33             VPName i_encryption_parameters selectedVariants(defaultparams)
34       confvariant biometrics mapping
35             VPName i_auth_type selectedVariants(fingerprint fingervein iris face; min:1, max:1)
36             VPName i_encryption_parameters selectedVariants(setparams)
37       defaultVariant userinfo
38       type parameterization
39       bindingTime devtime
```

Figure 3: Configuration interface of user verification choreography.

```
1 Constraints
2     i_auth_type requires i_auth_mode selectedVariants(mode_online)
3     i_auth_mode mode_online const protocol = "https"
4     i_auth_type esign const i_encryption_parameters defaultparams = "username,password and esign"
5
6  Parameter Settings
7        parameter noofbiometricauthtypeselected = #ofSelectedVariants{fingerprint fingervein iris face} Of i_auth_type
8        parameter defaultparams = [username_passw] + [selected{onetimepassw,esign}]
```

Figure 4: Constraint and parameter setting specification in configuration interface of user verification choreography.

Any variant can activate required functions in service and choreography interfaces. "i_auth_mode", internal variation point (line 20) is responsible for activation of different functions of storage and thirdparty services when its related variants are selected. For instance, "mode_online" varaint activates "getconnection, savehasheddata, verify" functions of thirdparty service when selected (line 22).

Constraints includes a logical constraint (line 2), stated that "i_auth_type" variation point requires "mode_online" variant of "i_auth_mode" variation point to be selected. In lines 3-4 numerical constraints are depicted in which "mode_online" variant of "i_auth_mode" variation point contraints the "protocol" property to be set to "https".

Moreover, any variability in choreography configuration interface that affects context elements in choreography can be defined in Parameter Settings part. Their values are set when the choreography is configured. For instance, "noofbiometricauthtypeselected" in Figure 5 (line 31) identifies the number of times for extracting features from biometric data. Its value is assigned (line 7) when variants of "i_auth_type" is selected.

4.2 Choreography

Choreography model includes composition constructs with variability attachments, context elements and variability associations between interacting services and choreographies. User verification choreography composes nine different services and interacts with three other choreographies. Importing collaborating choreographies and services with or without their own configuration interfaces provides an opportunity to utilize them with different configuration interfaces, that is with different service interfaces.

```
1 choreography userverification
2
3    import configuration vconf_verification
4
5    use choreography chor_warning
6    use choreography chor_warning
7    use choreography chor_connection
8
9    import service encryption with configuration vconf_encryption
10   import service imageretrieval
11   import service credentials
12   import service storage
13   import service attemptcalc
14   import service comparison with configuration vconf_comparison
15   import service responsewindow
16   import service interfaceprep with configuration vm_interfaceprep
17   import service thirdparty with configuration vm_thirdparty
18
19   Context Elements
20    wrongattempts 0
21    fakeinterface false
22    noofbiometricauthtypeselected 0
23
24   Choreography Variability Mapping
25    VP i_encryption_parameters maps service encryption VP encryption_params
26          Variant defaultparams maps Variant withdefaultparams
27          Variant setparams maps Variant withparams
28         ...
29   Function verify:
30       sequence (
31       #vp i_auth_type ifOneSelected( fingerprint fingervein iris)# repeat noofbiometricauthtypeselected times
32       (
33          imageretrieval receive message extractfeatures(biometric_data) refers imageretrieval.extract_features
34       )
35
36       ...
37      #vp i_auth_mode ifSelected(mode_offline)# sequence (
38          encryption send{storage} referedDestinations (comparison) message sendstoreddata() refers storage.get_hashed_data
39          #vp i_transaction_type ifSelected(faketransaction)# storage send{comparison} message compare(hasheddata) refers
                                                                                     comparison.compare
40       )
41       comparison send{attemptcalc} message calculatewrongattemps(result) refers attemptcalc.calculate_wrong_attempts
42       %comp wrongattempts = attemptcalc.calculate_wrong_attempts%
43       ...
44    )
```

Figure 5: User verification choreography specification with XChor.

```
Service interface encryption

function encrypt
      precondition(sessioncreated == true)
      postcondition(data_encrypted == true)
      input(credentials)
      output hasheddata

function setparams
            precondition(params_required      ==
true)
            postcondition(set_params == true)
            input(parameters)

      portName encryption binding hostname:8082
```

```
Choreography interface chor_verification of userverification

function verify
      precondition(authentication_mode_selected == true)
      postcondition  (verification_result_set == true)
      input(user_info)
      output response

portName verifyuser binding hostname:8082

      required interfaces
            from chor_warning function { warn }
            from chor_connection function { closeconnection }
            from chor_alert function { alert }
```

Figure 6: Encryption service and user verification choreography interfaces.

Variables defined with their default values based on the Context Elements part are affected by service interactions. For instance, “wrongattempts” is newly specified here to store the number of wrong attempts to limit verification trials.

User verification choreography associates its internal variation points and related variants to those of utilized services’ in order to configure service interface variability. The association between lines 25-27 ensures that when “i_encryption_parameters” variation point is bound to one of its variants, “encryption_params” variation point of encryption service is bound accordingly to provide a consistent interaction. With this, when defaultparams is selected, encryption service interface is configured with regard to withdefaultparams variant.

User verification choreography carries out “verify” functionality (line 29) comprising a set of interactions. Atomic and composite interactions are tagged with variability attachments whenever the part of the composition is changeable with regard to variability. In Figure 5, the lines 31-34, 37-40, and 39 include attachments referring to specified variation declarations in the configuration interface of the user verification choreography. "#vp i_auth_mode ifSelected(mode_offline)" to depict the point which composition can change (line 37).

4.3 Service and Choreography Interface

Service and choreography interface model comprises only interface specifications without variability. Each choreography and service has its own interface including all possible functionalities to be configured by configuration interfaces.

The interface of encryption service utilized in user verification choreography is shown in the left hand side of Figure 6. Exposed functionalities "encrypt" (line 3), and "setparams" (line 9) with pre-post conditions, input and outputs are depicted. Other services and choreographies can collaborate with it using "encryption" port (line 14).

Interface of user verification choreography; "chor_verification" depicts its functionality "verify" with pre-post conditions, input and output parameters. Different from encryption service interface, it explicitly states required choreographies with a list of functions.

4.4 Tool Support

Xtext is used to implement XChor Language which provides a development environment for domain specific languages to developers with Eclipse IDE integration. XChor files created from three models are: (i) choreography interface, (ii) service interface, (iii) configuration interface for choreography, (iv) configuration interface for service, and (v) choreography specification. These files can be categorized under configuration, services, and choreographies packages respectively in order to increase understandability.

Choreography, orchestration and atomic services are defined with variability specifications in Xtext. Binding variability and revealing a consistent collaboration require association analysis between variability specifications. This analysis requires considering variability constraints, choreography and service configurations (coming from configuration interfaces) with regard to variation selections. For this purpose, XChorS tool is provided

- to analyse variability associations which reveal configuration effects on orchestration and service interfaces,
- to configure choreographies and services with regard to variant selections, and
- to output configured XChor files in a specified destination folder.

XChorS tool employs parsing, variability association analysis, and configuration phases. It also includes some utilities for developers; binding time analysis and variation point redundancy analysis.

The tool parses related XChor files, discovers dependencies and constraints between them which are specified in configuration interfaces and choreography specification. The variability association analysis shows which variation points are related with which services and service functions.

It helps in the configuration phase to determine which services interact with each other and which functions should reside with which parameters in their interfaces. According to variation selections, the tool (i) configures interfaces by enabling and disabling its functions and parameters, (ii) prepares choreography compositions and external behaviour specifications of orchestration by examining whether the parts with variation attachments are included. Finally, the tool outputs configured choreography and related services and configuration interfaces if there are variation points that will be bound at runtime.

5 CONCLUSIONS

Existing orchestration and choreography languages do not address interface and composition variability explicitly.

Moreover, a single variation model covering all granularity levels, namely choreography, orchestration and atomic services is not proposed. Our approach is based on reusing existing architecture via explicit variability definition and management in SOA proposing a solution to fulfill interface and composition variability requirements. Taking into account challenges of variability scattered throughout the architecture, making feasible to develop variable service-oriented systems, and integrating variable orchestration specifications, a new variability meta-model and language; XChor is constructed and explained in detail. Variability constructs are treated as first class entities and can be defined in all granularity levels.

As a result of our contributions, we improve development of variable service-oriented systems reducing their complexity while providing consistent service interaction with regard to variability specifications. We think that in addition to modelling variable choreographies and relating them to orchestrations and services, verification of the model is important. So, verification of variable choreography is taken into consideration as a future work. Moreover, a runtime environment for XChor and relation with standard modelling languages are our current ongoing research.

REFERENCES

Arbab F., 2004. Reo: a channel-based coordination model for component composition, *In MSCS, Journal of Mathematical Structures in Computer Science.* Cambridge University Press.

Barker A., Walton C. D., Robertson D., 2009. Choreographing Web Services, *In Journal of IEEE Transactions on Services Computing.* IEEE.

Barros A.P., Dumas M., ter Hofstede, A. H. M., 2005. Service Interaction Patterns, *In 3rd International Conference on Business Process Management*. Springer.

Decker G., Kopp O., Leymann F. and Weske M., 2007. BPEL4Chor: Extending BPEL for Modelling Choreographies*, In ICWS'07, 5th International Conference on Web Services,* IEEE Computer Society.

Erl T., 2005. *Service-Oriented Architecture: Concepts, Technology, and Design,* Printice Hall, Indiana, 1st edition.

Fensel, D., Lausen, H., Polleres, A., Bruijn, J. de, Stollberg, M., Roman, D., Domingue, J., 2007. *Enabling Semantic Web Services, The Web Service Modeling Ontology*. Springer.

Lanese I., Bucchiarone A., and Montesi F., 2010. A Framework for Rule-Based Dynamic Adaptation, *In TGC 2010, 5th Symposium on Trustworthy Global Computing*. Springer.

Montesi F., Guidi C., Lucchi R. and Zavattaro G., 2007. JOLIE: a Java Orchestration Language Interpreter Engine, *In Journal of Electronic Notes in Theoretical Computer Science*. Elsevier.

OASIS, 2007. *Web Services Business Process Execution Language Specification, WS-BPEL*. OASIS.

OMG, 2011. *BPMN 2.0 Specification.* OMG.

Koning M., Sun C., Sinnema M., Avgeriou P., 2009. VxBPEL: Supporting variability for Web services in BPEL. *In Journal of Information & Software Technology*. Elsevier.

Kopp O., Engler L., Lessen T., Leymann F., Nitzsche J., 2010. "Interaction Choreography Models in BPEL: Choreographies on the Enterprise Service Bus", *In S-BPM ONE 2010 - the Subjectoriented BPM Conference*.

Lianping C., Babar A., Nour M. A., 2009. Variability management in software product lines: a systematic review, *In SPLC'09, 13th International Software Product Line Conference*. Carnegie Mellon University.

Nguyen T., Colman A., Talib M. A., Han J., 2011. Managing service variability: state of the art and open issues, *In VaMoS '11, 5th Workshop on Variability Modeling of Software-Intensive Systems*. ACM.

Sinnema M., Deelstra S., Nijhuis J., Bosch J., 2004. COVAMOF: A Framework for Modeling Variability in Software Product Families, *In SPLC3, 3rd Software Product Lines Conference.* Springer.

W3C, 2005. Web Services Choreography Description Language Specification, W3C.

XChor 2012. XChor Language Representation in Xtext, available at: http://www.xchor.com/XChorLanguage-xtext.pdf.

Xtext, 2012. Xtext 2.3.1, available at http://www.eclipse.org/Xtext/.

Zaha J. M., Barros A. P., Dumas M., Arthur H. M., 2006. Let's Dance: A Language for Service Behavior Modeling, *In OTM 2006, 4th On the Move to Meaningful Internet Systems Conference*. Springer.

A Philosophical Foundation for Business and IT Alignment in Enterprise Architecture with the Example of SEAM

Gil Regev, Biljana Bajic-Bizumic, Arash Golnam, George Popescu, Gorica Tapandjieva, Anshuman B. Saxena and Alain Wegmann

École Polytechnique Fédérale de Lausanne (EPFL), School of Computer and Communication Sciences, CH-1015 Lausanne, Switzerland

{gil.regev, biljana.bajic, arash.golnam, george.popescu, gorica.tapandjieva, anshuman.saxena, alain.wegmann}@epfl.ch

Keywords: Business-IT Alignment, Theory, Philosophy, Methodology, SEAM, Service, Systems Inquiry, PhD Hiring.

Abstract: Business-IT alignment is complicated because of the need to align multiple business and IT points of view. A philosophical foundation can help generate methods that bring together these disparate viewpoints in a common model that all stakeholders can agree to. In this paper, we describe the philosophical foundations of the Systemic Enterprise Architecture Method (SEAM) and show how it can help business-IT alignment with the example of a concrete business process. These foundations are applicable to other methods as well.

1 INTRODUCTION

The subject of business and IT alignment has been the focus of intensive research for over twenty years; see for example (Chan and Reich, 2007). It has also been a major concern for IT executives (Luftman and McLean, 2005). During all this time, it seems that few, if any, methods with a theoretical grounding have been proposed by researchers in this field (Chan and Reich, 2007). This is all the more surprising that it has been noted that cultural issues may be at the heart of misalignment between business and IT and that, despite the general tendency to believe otherwise, misalignment may not be counterproductive to some firms (Chan and Reich, 2007).

For many years, we have been contributing to the business and IT alignment field by building and applying an Enterprise Architecture method called SEAM. SEAM has an explicit theoretical grounding, or more precisely a philosophical grounding, which we describe in this paper.

Enterprise Architecture (Zachman, 1987) was created in the late 1980s in order to help IT departments to design IT systems that support the increasing complexity of businesses. This attempt was based on the premise that businesses increasingly depend on their IT systems, and that these systems (Zachman, 1987) “keep the business from disintegrating.” The term Enterprise Architecture (EA), initially referred to as information systems architecture, reflects this understanding that the information systems of an organization mirror the business itself. This has resulted in research into the combined fields of Enterprise Architecture and Business-IT alignment.

Many EA frameworks have been proposed since then. For example TOGAF (The Open Group, 2009) and ArchiMate (Lankhorst et al., 2009). In general these frameworks have no explicit theoretical grounding. They are implicitly based on strategic management practices that view the enterprise as a machine where executives set vision, goals that are then refined into IT architecture.

The Zachman framework stands out as having an epistemology in the sense that it has an ontology based on the work of a building architect including a different language for each trade.

SEAM focuses mainly on the enterprise architects’ role in helping with business-IT alignment and less on their role in mapping the IT infrastructure.

The term business-IT alignment hides much complexity. In any organization there are indeed many businesses, such as, a groups of people, departments, business units, a project teams. Each one is a business within a greater business with its own identity, worldview, behavior and structure. IT systems reflect the complexity of their environment. Embarking on business-IT alignment in order to embed this complexity in an IT system is a major challenge. It requires methods that enable enterprise architects to understand the multiple viewpoints, desires, and needs of these businesses within the enterprise as well as their external stakeholders. To appreciate and reconcile these points of view, we need to understand what is a business entity and how it sees itself and how it sees the world around it. Current Enterprise Architecture methods do not delve on sufficiently on these issues.

One of the main concepts used in EA discourse is the “system”. Lankhorst et al., for instance, give the examples of large systems such as enterprise information system and software system (Lankhorst et al., 2009). They further note that an architectural approach is needed to manage the complexity of such large systems. General Systems Theory (GST), (von Bertalanffy, 1968) also often called General Systems Thinking (Weinberg, 1975), was designed long ago to provide just the kind of architectural principles. GST can provide theoretical grounding and guide architects of large systems.

SEAM is an EA method that was created from the ground up based on GST. One of the main contributions of SEAM to EA is its reliance on an explicit systemic modeling paradigm (Wegmann, 2003). This paradigm provides a comprehensive explanation of SEAM in terms of its theory, philosophy and methodology. More specifically it provides a way to understand the often disparate viewpoints of the multiple businesses and IT within the organization.

In this paper we provide a fuller explanation of the paradigm. We explain how it can be useful in EA by showing its application in SEAM. We provide a short example of SEAM modeling based on a real university process, the hiring of PhD students at EPFL. SEAM is currently used as modeling method for the EPFL IT organization.

The paper is structured as follows: In Section 2 we present some background on business-IT alignment and EA. In Section 3 we describe the systemic modeling paradigm. In Section 4 we show the application of the paradigm to EA with the example of SEAM. In Section 5 we explain how the use of SEAM for the example of the PhD hiring process at EPFL illustrates the systemic modeling paradigm. In the last section we formulate our conclusions.

2 SYSTEMIC MODELING PARADIGM

Banathy and Jenlink (Banathy and Jenlink, 2004), seeking to provide a comprehensive description of GST, explain it as the interlinked association of three domains of inquiry: systems theory, systems philosophy (which further contains epistemology, ontology and axiology) and systems methodology (see Figure 1). They call this set Systems Inquiry.

Note that Banathy and Jenlink use the term ontology in its philosophical sense of what the real world contains. In the EA world ontology is more often used in its computer and information sciences meaning of “a set of representational primitives with which to model a domain of knowledge or discourse” (Gruber, 2009).

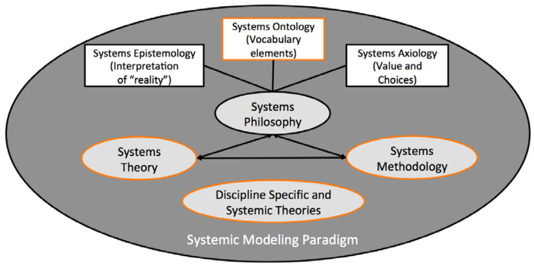


Figure 1: The Systemic Modeling paradigm (expanded from Systems Inquiry).

The systemic modeling paradigm was proposed by Wegmann in (Wegmann, 2003). It combines Systems Inquiry and Kühn’s notion of paradigm change. A paradigm is defined as “a philosophical and theoretical framework of a scientific school or discipline within which theories, laws, and generalizations and the experiments performed in support of them are formulated” (Merriam-Webster, 2013). The systemic modeling paradigm also extends Systems inquiry with the extension of the concept of discipline specific theories.

2.1 Systems Theory

Systems theory, as described by Banathy and Jenlink (Banathy and Jenlink, 2004) espouses the view that modern science and industry have locked themselves in a pursuit of an “ever-increasing specialization.” This specialization results in the inability, and often unwillingness, of specialists to engage, or even understand, other specialists.

The early system thinkers have observed that as each specialized discipline creates its own specialized vocabulary, it nevertheless uses concepts that are similar to other disciplines. It is often the vocabulary that is different but the underlying principles are the same. The same phenomena studied by a biologist can be observed in enterprises, for example. GST was therefore designed as a lingua franca that will enable specialists from different disciplines to collaborate (e.g. a biologist with an economist) and understand each other. GST seeks to define general principles that can be applied to any phenomena across established disciplines, thereby complementing the specialist view.

In addition to the general systems theory, Wegmann (Wegmann, 2003) proposed to use discipline specific theories to complement the general principles offered by the general systems theory.

2.2 Systems Philosophy

As noted by Banathy and Jenlink (Banathy and Jenlink, 2004), the interest of GST with general principles that transcend disciplines implies a close link with philosophy. They define systems philosophy as consisting of three components, Ontology, Epistemology and Axiology (ethics). Ontology describes what things are, e.g. what a person is, what an organization is, what a society is. Epistemology is oriented towards the questioning of ontology, e.g. how we know what is person, an organization, or society? Banathy and Jenlink contend that these two aspects are intimately linked because it is often impossible to completely separate what we know from how we know it. Finally, axiology is concerned the notions of value, ethics and aesthetics. It underlines the choices made by systems thinkers when they select some aspects of reality for attention rather than others. Are these choices good, bad, beautiful, ugly, moral or not, constitute the questions that axiology aims to reply to.

2.3 Systems Methodology

Systems methodology is the study and creation of methods for intervention. Banathy and Jenlink (Banathy and Jenlink, 2004) divide systems methodology into two domains of inquiry: the study of methods (their creation and improvement) and the practical use of these methods. The methods are used for the analysis of systems and systems problems, the design, development and implementation of systems and the management of systems in general. The method depends on the problem context and content as well as the type of systems in which the problem is situated. A specific methodology needs to be chosen from the wide range of available frameworks using a solid justification and analysis of the investigated problem.

3 THE SYSTEMIC MODELING PARADIGM APPLIED TO SEAM

Having briefly introduced the systemic modeling paradigm, we now use it to explain how an EA method, such as SEAM, can benefit from this grounding.

3.1 SEAM Systems Theory

SEAM is a method built on a systemic grounding. Much like GST is interested in federating scientific disciplines, when intervening in organizations, there is a need to understand and transcend the specialist view of the stakeholders (often called "silos" today) that compose the organization. While doing so, the enterprise architect should be careful not to alter too much the way of working of the stakeholders because their effective action depends on them remaining specialists.

In addition to GST, discipline specific theories can be used as well. These theories can be specific to the discipline of each stakeholder involved, e.g. marketing, sales, and software engineering. The theories specific to SEAM are, e.g., refinement theory to verify business-IT alignment, first order logic to formalize beliefs and operational semantics to formalize behavior.

3.2 SEAM Systems Philosophy

Parting from Banathy and Jenlink's explanation we explain the SEAM philosophy staring from the epistemology rather than the ontology.

The *SEAM epistemology* shown in Figure 2 is interpretative (Mintzberg et al., 1998), or interpretive (Checkland and Holwell, 1998). This means that we believe that each stakeholder creates his specialized knowledge of his work by interacting with the work artifacts and through his relationships with other specialists in his domain.

We call universe of discourse this set of entities that the stakeholder sees, which is a subset of the total number of entities available in reality. Two universes of discourse are shown in Figure 2, one for each stakeholder. The enterprise architect is also a specialist who constructs her models from her relationship with stakeholders and other enterprise architects. The universe of discourse of the enterprise architect is implicitly shown. It is made of the two stakeholders. The enterprise architect helps the stakeholders to express their knowledge about

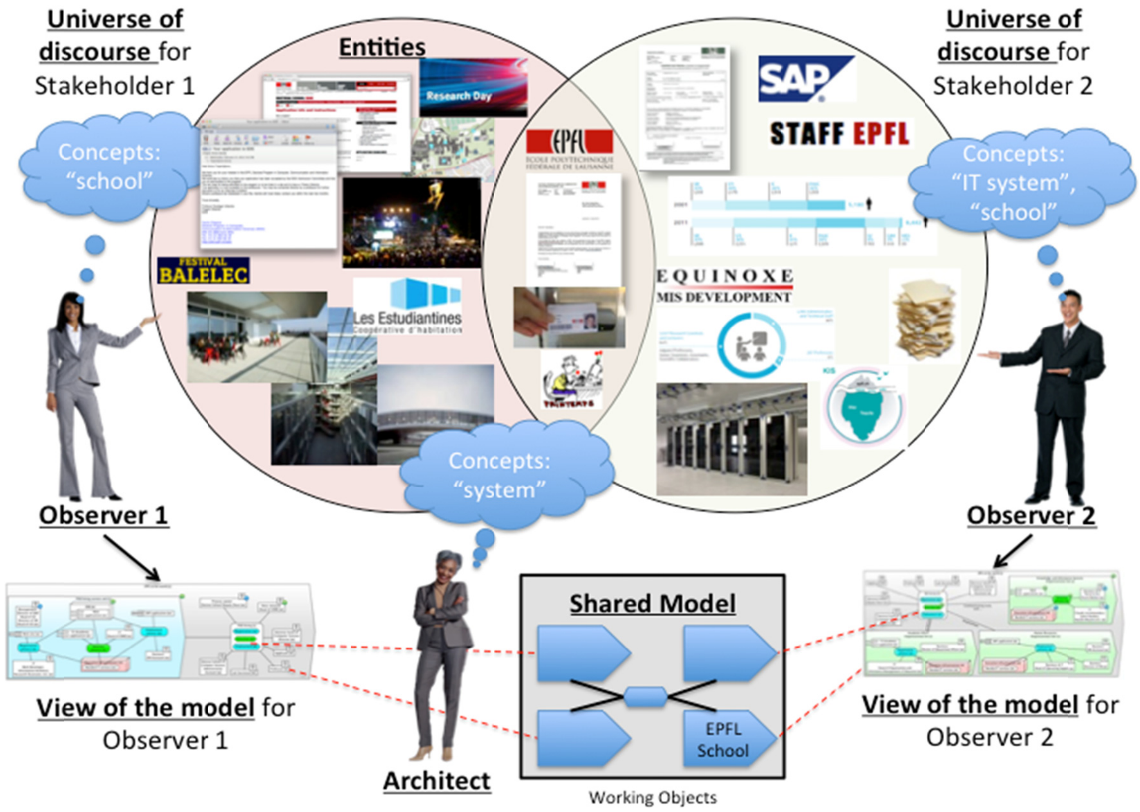


Figure 2: Illustration of the systemic modeling paradigm.

their work in a model that can be merged with other stakeholders' models.

Each stakeholder builds a set of concepts that we call his conceptualization by interacting with his universe of discourse. This conceptualization is the basis of his understanding of the world. Three conceptualizations are shown as clouds in Figure 2, one for each stakeholder and one for the enterprise architect. The enterprise architect constructs her conceptualization based on the set of stakeholders' conceptualizations.

Other terms that convey a similar meaning to the universe of discourse and conceptualization can be found in Vickers's appreciative system (Vickers, 1968), (Regev et al., 2011). Vickers explains that people and organizations form a readiness to see some aspects of reality. This readiness is necessary for effective action, but is also a barrier to collaboration with others because it makes it difficult to see things from a different point of view.

What we call the *SEAM ontology*, in-line with the standard use of the term ontology in computer and information sciences (Gruber, 2009), is the model elements with which an enterprise architect describes the stakeholders' conceptualizations and the shared model that the stakeholders should agree about.

In the SEAM ontology we use the term *working object* to designate a system in the conceptualization. For example, a working object named EPFL School in the model maps to a system that the modeler understands as being a school in the conceptualization. The name EPFL helps mapping to the specific school "EPFL" in the universe of discourse. This explains how the model element in the model relates to entities in the universe of discourse.

The ontology in the form of the working object allows to benefit from the domain specific theories proper to SEAM (e.g., refinement, model checking). A working object refers to a service system (Vargo et al, 2008), (Regev et al., 2011) in the sense that it shows the way value is co-created rather than an organizational entity, such as a company. The working object EPFL School may therefore contain other working objects that map to organizations that most stakeholders will think of as external to EPFL, for example, an IT supplier. Having the IT supplier working object within the EPFL School working object shows that the service provided by the EPFL School includes the service provided by the IT supplier.

The *SEAM axiology* refers to the choices the specialists make about what to include in her model. These choices can have two aspects: aesthetics and ethics (Lemos 1999). Aesthetics include practicality and simplicity. The modeler needs to decide to model what is useful and practical to show the problems and the possible solutions. The goal is not to make an exhaustive universal list of what exists in a company, but rather to analyze a concrete challenge. The modeler needs also to find a way to have simplicity. The modeler should use the abstraction mechanisms of SEAM to illustrate concisely the situation. Even if concise the model should keep the important systemic model elements (such as service system boundaries in the to-be model), so that the stakeholder can understand what is represented. Ethics – the model captures also the ethical choices of the modeled enterprise. For example, is the shareholder the primary "customer" of the company or should it be the "normal" customer. Axiology is useful to explain these two kinds of choices. Axiology is associated with heuristics (as, for example, that it is usually beneficial to understand first the "real" customer rather than the shareholder).

3.3 SEAM Systems Methodology

The SEAM methodology prescribes the way an enterprise architect uses the SEAM theory and philosophy to produce results. The methodology is a collection of techniques, some of which are well known to enterprise architects (such as the as-is and to-be modeling). Others were imported from other disciplines, e.g. contextual inquiry (Beyer and Holtzblatt, 1998). Because it is often costly and time consuming to do contextual inquiry in practice, we use an alternative technique of using concrete names of people and organizations (e.g., EPFL School rather than simply School) as well as anecdotes in workshops. This helps stakeholders remember the context they were in when facing some problems. Without this context, they may often forget to give many details about their work. A related technique encouraged in SEAM is to collect supporting evidence about concrete situations in the form of e.g., pictures, letters, and emails.

We also recommend developing a model bottom-up and top-down at the same time. We obtain the best results when the modeling sessions are short and iterative.

A few techniques were extended from standard techniques, e.g. the blackbox-whitebox technique is used to represent systems structure as is customary in engineering, but also to represent the structure of behavior, which is less frequent.

4 EXAMPLE OF THE PHD HIRING PROCESS WORKSHOP

In this chapter we show the importance of the SEAM philosophical grounding with a concrete and real example. We use the results of a one-day business-IT alignment workshop done in Fall 2012 at Ecole Polytechnique Fédérale de Lausanne (EPFL). We illustrate the relation between the SEAM theory and the workshop practice. EPFL uses a service-oriented strategy and is currently testing SEAM as modeling technique to represent its IT services. A workshop was planned to train IT managers in the use of SEAM so as to enable them to model their own services. The workshop was organized by the Laboratory of Systemic Modeling (LAMS) at the request of the IT governance head of EPFL. It was decided to work on the PhD hiring process as an example of a process that involves many departments and IT systems. The PhD hiring process is a good example of a process that brings together many actors across EPFL with many viewpoints that need to be reconciliated. It was also selected because it is an important process, with no projects currently planned to analyze it. It was therefore "neutral territory".

4.1 Organizational Description

EPFL is a polytechnic university located in Lausanne, Switzerland. EPFL. It is organized into seven schools, which are themselves formed of research and teaching units For the academic year of 2011-2012, EPFL had approx. 8'500 students, including 2000 PhD students. Some 500 new PhD students are hired each year. EPFL has about 4'500 employee.

IT is distributed across the whole organization. Approximately 80 people work in central services, under direct supervision of the Chief Information Officer (CIO). 20 people work in central services, outside of the CIO supervision. These 20 people manage mostly SAP and the academic management system, called ISA. Some 150 people work in the IT groups attached to the seven schools, or are dedicated to the IT of research and teaching units.

Overall, the IT people manage more than 125 central software applications, e.g. SAP for HR and finance, ISA, as well as some scientific infrastructure such as super-computers.

This distributed nature of the business and IT organizations leads to the co-existence of many viewpoints on any single process. There is a need to federate these viewpoints to improve business and IT alignment.

4.2 Description of the Current PhD Hiring Process

The process includes the following 3 phases:

Registration. The registration begins when an applicant fills an application record in ISA. The doctoral program committee analyzes all application records and decides who is admissible to the program. The doctoral program assistant informs, by e-mail, the applicant that he or she is admitted or rejected. The doctoral program assistant also informs by e-mail the professors that the list of admitted applicants is available in ISA.

Selection. The professor organizes interviews with potentially interesting admitted applicants. If the professor and applicant agree to work together, the offer is formalized in an admission letter signed by the professor and by the doctoral program director. The letter is sent to the applicant. No specific IT system supports this part of the process. It is implemented via e-mails, Word and Excel documents.

Employment. The unit's administrative assistant receives a copy of the admission letter. She asks the future students for the usual required documents (CV, passport copy, etc.). Note that the applicant already provides these documents at the beginning of the process - in the registration phase. The documents must be provided again because there is limited exchange of information between ISA and SAP. These documents, together with the admission letter are sent to the HR assistant, who is responsible for preparing the contract and arranging for the visa application, if needed. Once the contract is ready, it is sent for signature to the future PhD student and new records in the SAP human resource and finance management software modules are created.

4.3 The SEAM Workshop

The goal of the workshop organizer was to train the IT managers of the main applications on how to apply a service-oriented view to their application, using SEAM as a modeling method. A side goal was to make the participants aware of some of the technical and people issues concerning the PhD process and to prepare a follow-up workshop to address these issues (such as data integration between the first and third part of the process).

The workshop brought together six IT managers (e.g. SAP and ISA managers), the head of central IT and the person in charge of IT governance. The workshop was managed by one of the authors (Alain Wegmann) with the help of one of the co-authors (Gorica Tapandjieva). While writing this paper, we noticed that Alain Wegmann had three roles in this workshop: (1) workshop facilitator and SEAM trainer, (2) EPFL enterprise architect, (3) professor who hires PhD students. Ms. Tapandjieva had two roles: (1) SEAM trainer assistant, (2) Master's student at EPFL and applicant for a PhD position at EPFL. This means that she had, at the time of the workshop, a pending application in the PhD hiring process.

The workshop was held in the following way:

First, the participants expressed their expectations from the workshop. They were quite a few. For example, learning how to use SEAM to model services, finding ways how to work better with colleagues, or simply attending the workshop to see what comes out of it.

Next, we asked all participants to present the challenges they faced in managing their applications. The major challenges were: (1) understanding what the term "business" meant in business and IT alignment. (2) Defining who are the relevant representatives of the 10'000 or so EPFL users. (3) Understanding what is the IT and business strategy of EPFL

We then introduced the example of the PhD hiring process. We provided a two page textual description, a sequence diagram of the detailed process and a file with a copy of all documents from Ms. Tapandjieva's application. We briefly introduced some of the SEAM principles (how to model systems, services, and processes). The participants worked in three groups (2 groups of 3 and one of 2 participants) and had to make a SEAM model of the PhD hiring process. We ended the session with a debrief session and with a sketch of a SEAM model made by Alain Wegmann. The goal was to encourage the participants to practice SEAM (and thereby to understand the difficulties in using it) and then to show them how a SEAM modeler would create a model that exposes the issues they had identified at the beginning of the workshop.

We ended the morning with a debrief session during which the participants said they liked the concreteness and the dynamic aspects of the method.

Some participants found that the models were "more messy" as the ad-hoc ones they would normally make. Systemic models often appear less simple than add-hoc ones, who are frequently over-simplified.

In the afternoon we created a group-wide model of the PhD process. We discussed the technical and the organizational issues raised by a transition to a service approach. Figure 3 is a picture of the group-wide model that we created together.

The day ended up with a debrief session in which the participants agreed on the technical and organizational issues to address in moving to a service approach. Some raised the concern that we did not find a solution to these issues, but this was not planned for this workshop. It was also clear that a follow-up workshop should formally include more business users.

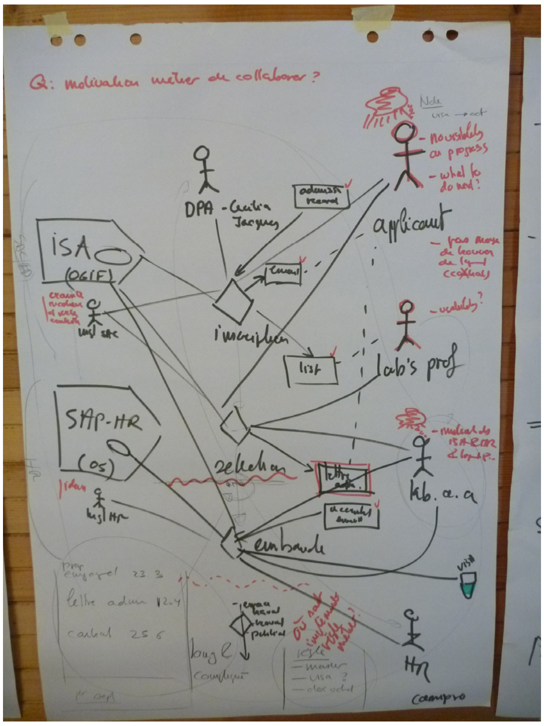


Figure 3: PhD Hiring SEAM model developed during the SEAM workshop.

In the morning the IT managers made their model in three separate groups. They based their model on the sequence diagram of the detailed process we gave them. So they all analyzed the overall process (i.e. the three phases). One of the models happened to be quite similar to the group-wide model shown in Figure 3. The second model represented the point-to-point interactions in the process, a sort of high-level view of the sequence diagram. It did not show the three phases identified in the group-wide model. Most notably, the model did not include the management of the admission letter, probably because this phase is not supported by an IT system. The third model represented the existing organizational boundaries within EPFL. The phases were represented within these boundaries. Remember that in SEAM we represent service systems, therefore these boundaries were not supposed to appear in this model.

4.4 The Importance of the Systemic Modeling Paradigm for the Workshop

Federating different models and different conceptualizations: The three different models made by the three different groups were the result of three different stakeholders conceptualizations.

All the models were valid but seemed incompatible with one another. The systemic modeling paradigm helped us to not quarrel about who is right or wrong but to accept each model as a bona fide representation for the person or people who created it.

To design a common process, it is important that all stakeholders share the same model. This means that it is also necessary to reconciliate their disparate conceptualizations. Changing people conceptualization, the way they see the world, is a known as a very difficult task. Axiology should help here for guiding the enterprise architect in this difficult task and in the choices that are inevitable in selecting what to represent in the common model.

If there is no conceptualization it will not be in the model: Each of the three groups modeled the overall process (with one group mostly focusing on the IT support). If we would not have given them the sequence diagram prior to the modeling exercise, it is very likely that the Selection phase would not have been represented because none of the IT managers provided support for this part of the process. None of the IT managers had it in their conceptualization.

The sequence diagram of the process was created by Ms. Tapandjieva who interviewed several stakeholders of the process and collected evidence about it before to the workshop. In addition, Ms. Tapandjieva was also a PhD applicant and her application was somewhat "stuck" in the Selection phase for a few months. So she was able to testify on the importance of this part of the process for an applicant. Thanks to the testimony of Ms. Tapandjieva and to the collected evidences, it was possible to model the Selection phase and to identify the issues that related to it (e.g. that it has no specialized IT support and that applications could get stuck in this phase).

Each IT manager could model with precision the phase that the application he was responsible for supported. This phase relates directly to his conceptualization because it corresponds to his specialization. One of the challenges during the workshop was to enable all IT managers to represent their phase at the same level of detail as the other phases.

One of the participants offered an additional conceptualization. His training as an auditor enabled him to discover a flaw in the sequence diagram of the process by attentively analyzing the dates of the documents provided as evidence. Without this specialization the sequence diagram would have not been challenged.

In summary, to have the viewpoints of the multiple stakeholders (including the non-IT one) is essential to understand the issues related to the process. This includes the IT issues. For example, the applicant has to submit his documents to ISA and to SAP. This leads to errors and delays. A technical solution can be found to link ISA and SAP. This problem can be identified only if the process is analyzed end-to-end. So, all viewpoints are necessary.

The use of concrete *evidences:* Some of the documents collected by that the way the process is executed leads to major issues for the applicant. For example, the applicant does not receive the necessary documents on time to find a housing. This level of concreteness motivates the other stakeholders to address the issues. They can relate to the applicant's problems. All the participants were able to relate to the feeling the applicant has when the document that would allow her to find an apartment is not received on time. This is much more concrete than the concept of "hard to find an apartment" that would usually be found in abstract models.

Without the evidence provided by the documents collected by Ms. Tapandjieva the auditor would not have found the flaw in the sequence diagram.

5 CONCLUSIONS

In this paper we emphasized the need to have a philosophical grounding for business-IT alignment because it is a crosscutting concern that potentially requires the collaboration of the entire organization. We described one such grounding, called the systemic modeling paradigm, which is based on general systems principles, and is the foundation of SEAM, an enterprise architecture method. The main originality of the systemic modeling paradigm is its breadth. It proposes 4 dimensions for underpinning a general-purpose method that can be effectively used in concrete projects. These dimensions are, theory, philosophy, methodology and discipline specific theories. Together they enable to transcend the divisions within an organization, while also understanding the specificities of each department or individual stakeholder. It is our hope that other researchers would use this paradigm or propose different paradigms to provide a philosophical foundation for their methods, an aspect that business and IT alignment urgently needs.

REFERENCES

Banathy, B. H. and Jenlink, P. M. (2004). Systems inquiry and its application in education., In Jonassen. D.H. (Eds.), Handbook of research on educational communications and technology, (pp. 37-57). Lawrence-Erlbaum,

Beyer, H. and Holtzblatt, K. (1998). Contextual Design: Defining Customer-Centered Systems. San Francisco, CA: Morgan Kaufmann.

Chan, Y.E. and Reich, B.H. (2007). IT alignment: What have we learned? Information Technology, 22, 297-315.

Checkland, P., Holwell, S. (1998). Information, Systems and Information Systems, making sense of the field. Chichester, UK: Wiley.

Gruber, T. (2009). Ontology. In Liu, L. and Özsu, M.T. (Eds.), Encyclopedia of Database Systems. Springer-Verlag.

Lankhorst, M.M., Proper, H.A. and Jonkers, H. (2009). The Architecture of the ArchiMate Language. Proc. BPMDS 2009 and EMMSAD 2009, LNBIP 29, Springer.

Lemos, N.M. (1999). Value theory In Audi, R. (ed.), The Cambridge Dictionary of Philosophy. Second Edition, Cambridge University Press, Cambridge UK.

Luftman, J., McLean, E.R. (2004). Key Issues for IT Executives, MIS Quarterly Executive, 3(2).

Merriam-Webster, (2013). http://www.merriam-webster.com/dictionary/paradigm, accessed May 2013.

Mintzberg, H., Ahlstrand, B. and Lampel J. (1998). Strategy Safary – The complete guide through the wilds of strategic management. Prentice Hall.

Regev, G., Hayard, O. and Wegmann, A. (2011). Service Systems and Value Modeling from an Appreciative System Perspective. In *IESS1.1, Second International Conference on Exploring Services Sciences*, Springer.

The Open Group. (2009). The Open Group Architecture

Framework (TOGAF) Version 9, The Open Group.
Vargo, S.L., Maglio, P.P. and Akaka, M.A. (2008). On value and value co-creation: A service systems and service logic perspective. European Management Journal 26(3), 145–152.
Vickers, Sir G. (1968). Value Systems and Social Process. Tavistock, London
von Bertalanffy, L. (1968). General System Theory. New York: George Braziller.
Wegmann., A. (2003). On the Systemic Enterprise Architecture Methodology (SEAM). International Conference on Enterprise Information Systems.
Weinberg, G. M. (1975). *An Introduction to General Systems Thinking*, Dorset House.
Zachman, J.A. (1987). A framework for information systems architecture. IBM Systems Journal 26(3).

Using Ex-SMCD for Developing Consistent Misuse Case Models with Extended Notation

Mohamed El-Attar and Hezam Akram Abdul-Ghani
Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran, Kingdom of Saudi Arabia
melattar@kfupm.edu.sa, almajd_2@yahoo.com

Keywords: Misuse Case Models, Model Consistency, Extended Notation, Ex-SMCD.

Abstract: Security is almost always a vital aspect in the development of any system. Many mission-critical systems are deemed unusable if they are not secure. Secure software development requires a secure software engineering process that takes into account security concerns from as early as the requirements phase. Misuse case modeling is a technique that was introduced to extend the already widespread technique of use case modeling for the purpose of eliciting, specifying and communicating functional security requirements. The notational set and syntax of misuse case modeling were further extended in order to more accurately specify the functional security aspects of a system. Similar to use case diagram, Misuse case diagrams created using this extended notational set are supplemented with textual descriptions that narrate the actual expected behaviour of a system. Since misuse case descriptions are described in syntax-free natural language, authors are likely to create misuse case models that are inconsistent, ultimately leading to the development a system that is functional non-secure. To counter this problem, this paper introduced a structure that guides authors while narrating misuse cases in order to significantly limit inconsistencies. A case study is presented to demonstrate and the application and feasibility of the proposed structure.

1 INTRODUCTION

Software systems nowadays are always interconnected with other system, which makes them constantly under threat of being misused. Misuse can be attempted by humans directly accessing a system or indirectly via another interconnected system. External defensive mechanisms such as firewalls, cryptography and Intrusion Detection Mechanisms are no longer sufficient (Jürgens, 2004). A software system needs to embody functionality that will allow it to defend for itself in addition to the external defensive mechanisms. Necessary security functionality must be taken into account from as early as the requirements engineering phase.

Similar to ordinary business requirements, security requirements are considered either functional or non-functional. Functional security requirements can be effectively modelled using misuse case modeling. The misuse case modeling concepts and notation were first introduced by (Sindre et al., 2000). Subsequently, many research works were directed towards improving and supporting the misuse case modeling technique, notation and description templates. An *extended* notational set of misuse case modeling was introduced by (Røstad, 2006) to include vulnerabilities and insider threat.

Inconsistencies within a use case model (OMG 2005, Jacobson 1995)can lead to confusion amongst stakeholders leading to the development of a system that does not satisfy its functional business requirements. Similarly, inconsistencies within a misuse case model can lead to the development of a system that is not functionally secure.

This paper introduces a structure named Ex-SMCD (Extended Structured Misuse Case Descriptions) that aids modelers during their authoring of mis/use cases. The Ex-SMCD structure is supplemented with a process named Ex-REMCD (Extended – Reverse Engineering of Misuse Case Diagrams) that will ensure the development of structurally consistent misuse case models. The Ex-SMCD structure focuses on extended notation of misuse case models as proposed by (Røstad, 2006) since it advances the current state-of-affairs of the modeling technique by allowing modelers to create misuse case models that are more accurate, descriptive and specific.

The remainder of this paper is structured as follows: Section 2 provides a brief background and discusses related works. Section 2 sheds light on the importance of developing consistent misuse models case. Section 3 presents the Ex-SMCD structure and its associated Ex-REMCD process. In Section 4, exemplars of misuse case diagrams that include the extended notation from the (Røstad, 2006) paper are used to demonstrate the application and feasibility of the Ex-SMCD and Ex-REMCD. Finally, Section 5 concludes and provides suggestions for future work.

2 BACKGROUND AND RELATED WORK

A misuse case model consists of a diagram that contains use cases, misuse cases, actors and misusers. The diagram acts as a visual summary of the business functional requirements, represented by use cases, and the functional misuse case behaviour that threatens the system while it is executing its business related behaviour, and represented as misuse cases. A subset of use cases are later introduced to mitigates against each misuse case. Each misuse and use case is supplemented with a textual description. The textual descriptions detail the expected behaviour from an interaction perspective. A misuse case model also shares a glossary with other artefacts used in the development project.

Inconsistencies may exist in a use case model between any of its components. Inconsistencies may occur within the textual descriptions, between different diagrams, or between the textual descriptions and the diagram. As such, modelers are highly vulnerable to introducing inconsistencies in their models.

The harmful consequences of inconsistencies in misuse case models have been well documents. Many of these consequences stems from what the literature has reported about inconsistencies in use case models, which is applicable to misuse case models as well. Inconsistencies and its harmful consequences were listed as part of a comprehensive taxonomy presented in (Anda et al., 2001). The taxonomy states that inconsistencies in a use case models will have a very damaging effect on every aspect of the development process. Therefore, inconsistencies will severely hamper the overall quality of the end product. Many examples inconsistency defects were presented in (Lilly, 1999). For example, if the system boundary was not consistently defined ten that will lead to ambiguity with regard to actual functional requirements that needs to be realized by the development team. Consequently, the development team may omit some requirements or waste their resources implementing functionality that is outside the scope of the intended system. A misuse case model maybe rendered useless if it contains a high level of inconsistencies as it no longer represents the behaviour of the underlying system (Ambler, 2013), which in turns impedes downstream development efforts, particularly testing and maintenance. Inconsistencies in a mis/use case model are also symptomatic of vague or omitted information (Chandrasekaran, 1997). Inconsistencies in a mis/use case model have also been linked to the existence of an ambiguous domain model.

Due to the severe detrimental effects of inconsistences in mis/use case models, many research works have been devoted towards improving the level on inconsistency. Some techniques proposed in the literature have highly recommended various types of reviewing to ensure consistency in mis/use case models (Kulak and Guiney, 2000; Armour and Miller, 2000). Automated approaches were also introduced to improve various quality attributes of mis/use case models including consistency such as model refactoring (Bulter and Xu, 2001). McCoy (2003) have presented a tool that provides use case authors with a template to describe their use cases. The use of the template helps ensure a certain degree of consistency.

3 Ex-SMCD AND Ex-REMCD

The chief purpose of the Ex-SMCD structure is to reduce inconsistencies in misuse case models, particularly structural inconsistencies. One of the most attractive features of mis/use cases is its readability. Therefore, the design of Ex-SMCD deliberately accounted for readability. To maintain a high level of readability, the Ex-SMCD structure consists of a set of English keywords that pertain to the misuse case modeling domain. Using Ex-SMCD requires mis/use case authors to embed their textual descriptions with these keywords while abiding to a specific grammar. The grammar formalizes how the keywords should be used. The grammar of the Ex-SMCD structure is shown in the Appendix.

Table 1: A summary of the Ex-SMCD structure constructs.

Section	Keyword	Diagram Representation
Mis/use Case Name	ABSTRACT	*Abstract* mis/use cases are depicted in italic font in the diagrams.
	SPECIALIZES	A generalization relationship link is depicted in the diagram.
	IMPLEMENTS	A generalization relationship link is depicted in the diagram. This is due to the fact that the generalization and implementation relationships are depicted using the same notation.
	VULNERABLE (use cases only)	The use case is depicted with a grey background to indicate that it is vulnerable
	The name of the mis/use case	A mis/use case with the given name is displayed in the diagram.
Description	INCLUDE	Results in the creation of an *include* relationship directed towards the mis/use case stated in the INCLUDE statement.
Extended Mis/use Cases	Base Mis/use Case	An *extend* relationship link is created and directed towards the stated base mis/use case.
	Extension Point	Optional to the user. Results in the augmentation of the targeted extension point name on the *extend* relationship link.
	IF	Optional to the user. The condition is displayed on the *extend* relationship link in square brackets.
Extension Points	The names of public extension points	Each extension point stated is depicted within the oval of the given mis/use case in the diagram.
Threatens (misuse cases only)	Threatens	A threatens relationship link is depicted in the diagram from the given misuse case to the target use case.
Exploits (misuse cases only)	Exploits	An exploits relationship link is depicted in the diagram from the given misuse case to the target use case.
Mitigates (use cases only)	Mitigates	A mitgiates relationship link is depicted in the diagram from the given use case to the target misuse case.

In a misuse case model there are two sets of textual descriptions. One set is used to describe use cases while the other is used to describe misuse cases. For use cases, the Ex-SMCD structure will appear in five sections, these are: (a) Use Case Name, (b) Associated Actors, (c) Description, (d) Extension Points and Extended Use Cases, and (e) Mitigates Section. For misuse cases, the Ex-SMCD structure will appear in five sections, these are: (a) Misuse Case Name, (b) Associated Misusers, (c) Description, (d) Extension Points and Extended Misuse Cases, and (e) Threatens Section. A brief description of each keyword is shown in Table 1. The Ex-SMCD also appears in actor, misuser and insider description in the Name section. Figures 1 and 2 illustrates the concepts explained above and demonstrates the visually the mapping of the keywords in Table 1 using a mock example.

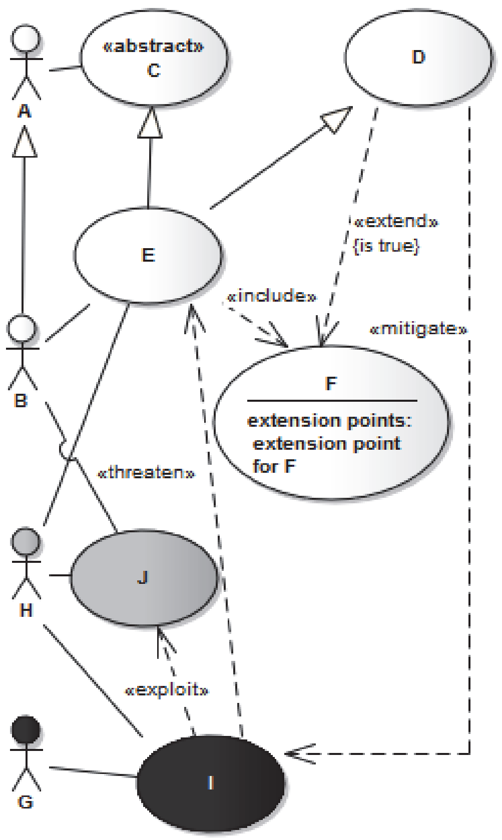


Figure 1: Example misuse case utilizing the extended notation.

The Ex-REMCD Process

When given a set of mis/use case descriptions that use Ex-SMCD, the Ex-REMCD process uses these textual descriptions as input to generate a structurally consistent misuse case diagram. This process consists of four phases:

Phase 1: Identify actors, misusers and insiders to create XML components to represent these entities to be displayed by a UML modeling tool.

Mock Example Textual Descriptions
Actors
Actor Name: A **Description:** A brief description of actor A
Actor Name: B **SPECIALIZES:** A **Description:** A brief description of actor B
Insiders
Insider Name: H **Description:** A brief description of insider H
Misusers
Misuser Name: G **Description:** A brief description of misuser G
Use Cases
Use Case Name: C **ABSTRACT** **Associated Actors:** A **Description:** A brief description of use case C
Use Case Name: D **Description:** A brief description of use case D **Extended Use Cases:** **Base UC Name:** F **AT**: extension point of F **IF**: is true **Mitigates:** I
Use Case Name: E **IMPLEMENTS:** C **SPECIALIZES:** D **Associated Actors:** B **Description:** A brief description of use case E and INCLUDE <F>
Use Case Name: F **Description:** A brief description of use case F. **Extension Points:** Extension point of F
Use Case Name: J VULNERABLE **Associated Actors**: B **Associated Insiders**: H **Description:** A brief description of use case F.
Misuse Cases
Misuse Case Name: I **Associated Misusers:** G **Exploits:** J **Threatens:** E **Description:** A brief description of misuse case I.

Figure 2: Descriptions of the mis/use cases using Ex-SMCD.

Phase 2: Identify use and misuse cases to create XML components to represent these entities to be displayed by a UML modeling tool.

Phase 3: Identify relationships between actors and use cases, and between misusers and misuse cases. Also identify relationships between insiders and mis/use cases. The identified relationships will be used to create corresponding XML components. This step will require cross-referencing with XML components previously created in the previous two iterations.

Phase 4: Identify relationships between misuse cases and use cases, misuse cases and other misuse cases, and use cases and other use cases. The identified relationships will be used to create corresponding XML components. This step will require cross-referencing with XML components previously created in the previous three iterations.

When given a misuse case diagram, the REMCD process uses the XML file that contains the data belonging to the diagram to create structurally consistent Ex-SMCD skeleton descriptions. This process consists of five phases:

Phase 1: Identify actors, misusers and insiders to create a text area for each entity with its name and the appropriate fields.

Phase 2: Identify mis/use cases and create a text area for each mis/use case with its name and the appropriate fields.

Phase 3: Identify the relationships between actors and use cases and amend the corresponding text area to reflect these relationships. Identify the relationships between misusers and misuse cases and amend the corresponding text area to reflect these relationships. Identify the relationships between insiders and mis/use cases and amend the corresponding text area to reflect these relationships.

Phase 4: Identify the relationships between misuse cases and use cases, misuse cases and other misuse cases, and use cases and other use cases. The identified relationships will be used to amend the corresponding text area to reflect these relationships.

Phase 5: Identify the characteristics of each mis/use case, such as if it is *abstract* or *vulnerable*. The identified characteristics will be used to amend the corresponding text area to reflect these characteristics.

Finally, the text areas are combined into one file.

4 EVALUATION: APPLYING Ex-SMCD TO THREE EXEMPLARS

This section demonstrates the application and feasibility of Ex-SMCD. In this section a number of *extended* misuse case diagrams will be constructed systematically from Ex-SMCD descriptions. The misuse case diagrams presented here are similar to those presented in the paper by (Røstad, 2006). The reason for selecting these diagrams specifically is because they are inclusive of all the original and extended misuse case modeling notation.

Evaluation of the correctness of this proposed structure and technique will be based on similarity comparison between the diagrams generated from the Ex-SMCD descriptions with the diagrams presented in the paper by (Røstad, 2006). If the diagrams were structurally identical then the application of Ex-SMCD was successful.

The comparison will be performed by a tool named UseCaseDiff (El-Attar, 2011). UseCaseDiff is differencing tool used to differentiate any structural differences between two use case diagrams. UseCaseDiff was extended in later work to account for the original notation of misuse case modeling. For this research work, UseCaseDiff was once again extended to account for the *extended* notation. Another comparison will be performed via manual inspection of the diagrams as an added measure of assertion.

Figures 3, 5 and 7, present the Ex-SMCD descriptions of three corresponding misuse case diagrams shown in Figured 4, 6 and 7, respectively. The description shown in Figure 3 and its corresponding diagrams shown in Figure 4 relates to emergency access control in healthcare systems. It specifies the use and misuse of the access control mechanism in an Electronic Patient Record (EPR) system. The description shown in Figure 5 and its corresponding diagrams shown in Figure 6 relates to user input in web-enabled systems. It specifies how input from users should be handled. Finally, the description shown in Figure 7 and its corresponding diagrams shown in Figure 8 relates to an insider on the system development team. This model specifies the threats posed by a dissatisfied employee on the development team.

4.1 Evaluation

Applying the REMCD process resulted in the systematic generation of the diagrams shown in Figures 4, 6 and 8 from their corresponding descriptions. For each generated diagram, its data file was used as input into the too UseCaseDiff along with the manually developed model that matches the corresponding diagram in (El-Attar, 2011).

Case 1: **Emergency Access Control in Healthcare Systems**
Actors
1) Actor Name: Authorized User **Description:** <A brief description of the actor>
Insiders
1) Insider Name: Insider **Description:** <A brief description of the insider>
Use Cases
1) Use Case Name: Read EPR **Associated Actor:** Authorized User **Description:** <A brief description of the use case>. INCLUDE<Normal AC>
2) Use Case Name: Emergency Read EPR **Associated Actor:** Authorized User **Description:** <A brief description of the use case>. INCLUDE< Emergency AC >
3) Use Case Name: Normal AC **Description:** <A brief description of the use case>. **Extended Use Cases:** **Base Use Case:** Access Control (AC)
4) Use Case Name: Access Control AC **Description:** <A brief description of the use case>.
5) Use Case Name: Emergency AC VULNERABLE **Description:** <A brief description of the use case>. **Extended Use Cases:** **Base Use Case:** Access Control (AC)
6) Use Case Name: Auditing **Description:** <A brief description of the use case> **Mitigates:** Unauthorized Read EPR
7) Use Case Name: Awareness Training **Description:** <A brief description of the use case> **Mitigates:** Unauthorized Read EPR
Misuse Cases
1) Misuser Name: Unauthorized Read EPR **Associated Misusers:** Insider **Description:** <A brief description of the misuse case>. **Exploits:** Emergency AC

Figure 3: Descriptions of the mis/use cases of the "Emergency Access Control in Healthcare Systems" diagram.

The tool UseCaseDiff did not report on any core structural differences. Moreover, the manual inspection phase did not report any differences between the diagrams.

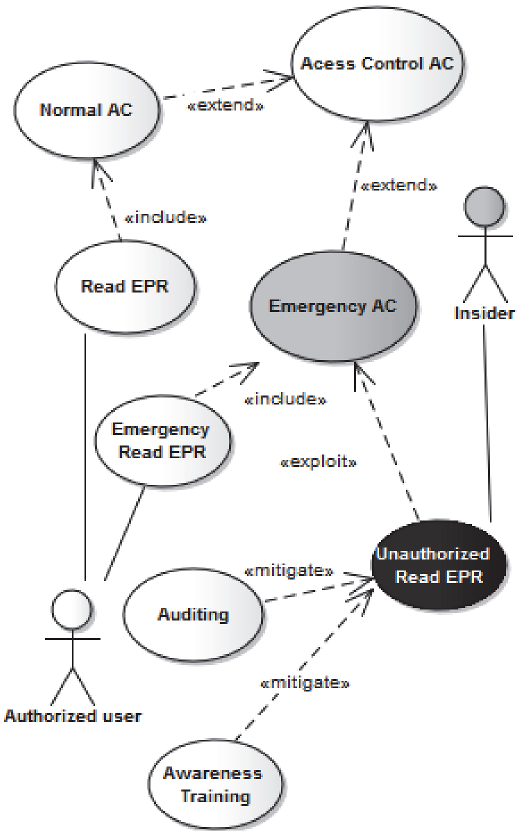


Figure 4: Misuse case diagram for emergency access control in healthcare systems.

5 CONCLUSIONS

Security is increasingly becoming an important quality in software systems. Misuse case modeling is a useful technique for eliciting and modelling security requirements and threats. However, inconsistencies in misuse case models may have very damaging effects downstream in the development life cycle leading to the development of systems that are insecure. It is therefore crucial to develop misuse case models that are consistent. This paper presents a structure named Ex-SMCD that guides modelers during the authoring phase to develop misuse case models that are structurally consistent. Moreover, the structure will also serve as a guideline that prompts authors to consider documenting important information in misuse cases that might otherwise be overlooked.

Case 2: **User Input in Web-Enabled Systems**
Actors
1) Actor Name: Authorized User **Description:** <a brief description of the actor>
Misusers
1) Misuser Name: Attacker **Description:** <a brief description of the misuser>
Use Cases
1) Use Case Name: Enter Username VULNERABLE **Description:** <A brief description of the use case>.
2) Use Case Name: Input Validation **Description:** <A brief description of the use case>. **Mitigates:** Injection Attack, Overflow Attack
3) Use Case Name: Enter Password VULNERABLE **Description:** <A brief description of the use case>.
4) Use Case Name: Use System **Description**: <A brief description of the use case>.
Misuse Cases
1) Misuse Case Name: Injection Attack **Associated Misusers:** Attacker **Description:** <A brief description of the misuse case>. **Exploits:** Enter Username, Enter Password **Threatens:** Use System
2) Misuse Case Name: Overflow Attack **Associated Misusers:** Attacker **Description:** <A brief description of the misuse case>. **Exploits:** Enter Username, Enter Password **Threatens:** Use System

Figure 5: Descriptions of the mis/use cases of the "User Input in Web-Enabled Systems" diagram.

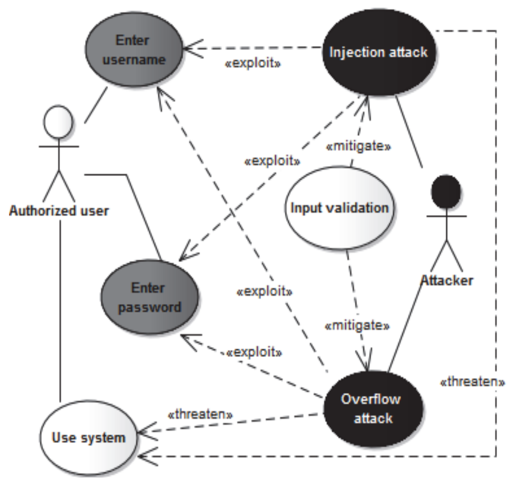


Figure 6: Misuse case diagrams for user input in web-enabled systems.

Case 3: An Insider on the System Development Team
Actors
1) Actor Name: System developer **Description:** <A brief description of the actor>.
Insiders
1) Insider Name: Insider **Description:** <A brief description of the insider>.
Use Cases
1) Use Case Name: Implement system **Associated Actor:** System developer **Associated Insiders:** Insider **Description:** Perform Implement system
2) Use Case Name: Code audit **Description:** <A brief description of the use case>. **Mitigates:** Inject backdoor, Inject bug
3) Use Case Name: Security testing **Description:** <A brief description of the use case>. **Mitigates:** Inject backdoor, Inject bug
Misuse Cases
1) Misuse Case Name: Inject backdoor **Associated Insiders:** Insider **Description:** <A brief description of the misuse case>.
2) Misuse Case Name: Inject bug **Associated Insiders:** Insider **Description:** <A brief description of the misuse case>.

Figure 7: Descriptions of the mis/use cases of the “An Insider on the System Development Team” diagram.

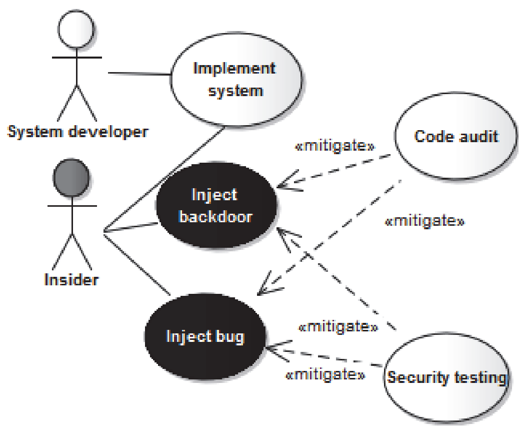


Figure 8: Misuse case diagrams for an insider on the system development team.

This paper also introduces the Ex-REMCD process which uses misuse case descriptions embedded with the Ex-SMCD structure to develop structurally consistent misuse case diagrams. The Ex-REMCD process can also use misuse case diagrams to develop description skeletons that are structurally consistent with the given diagram. As the misuse case model evolves and changes are introduces in the misuse case diagrams or descriptions, it is ensured to remain consistent at all times due to the cyclic feature of the Ex-REMCD process. The feasibility of the Ex-SMCD and the Ex-REMCD were successfully demonstrated using the misuse case models that were presented in the paper by (Røstad, 2006), which is the same research work that presented the extended notation for misuse case modeling. Using Ex-SMCD and Ex-REMCD, the descriptions for the three diagrams presented in (Røstad, 2006) were developed and systematically used to generate diagrams that are identical to those in (Røstad, 2006).

Future work can be directed towards empirically evaluating the usability of Ex-SMCD by its potential users. This study will be very beneficial as the usability of any technique is a core factor for its adoption by practitioners.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the support provided by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) for funding this work through project **No. IN111028**.

REFERENCES

Ambler, S.: Agile Modeling http://www.agilemodeling.com/essays/whenIsAModelAgile.htm. Last Accessed March 2013.

Anda, B., Sjøberg, D., and Jørgensen, M., 2001. Quality and Understandability in Use Case Models, 15th European Conference Object-Oriented Programming (ECOOP), edited by J. Lindskov Knudsen. Springer-Verlag, Budapest, Hungary, pp. 402-428.

Armour, F., Miller, G., 2000. Advanced Use Case Modeling. Addison-Wesley.

Butler , G., Xu, L., “Cascaded refactoring for framework evolution,” Proceedings of 2001 Symposium on Software Reusability, ACM Press, pp. 51-57.

Chandrasekaran, P., 1997. How Use Case Modeling Policies Have Affected The Success of Various Projects (or How to Improve Use Case Modeling), Addendum To The 1997 ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, pp. 6-9.

El-Attar, M., 2011. UseCaseDiff: An Algorithm for

Differencing Use Case Models, 9th International Conference on Software Engineering Research, Management and Applications, pp. 148-152.
Jacobson ,I., Ericsson, M., Jacobson, A., 1995. The Object Advantage. ACM Press.
Jürjens, J., 2004. Secure Systems Development with UML. Berlin, Springer.
Kulak, D., Guiney, E., 2000. Use Cases: Requirements in Context. Addison-Wesley.
Lilly, S., 1999,.Use Case Pitfalls: Top 10 Problems from Real Projects Using Use Cases, Proceedings of TOOLS USA, IEEE Computer Society,- pp. 174-183.
McCoy, J., 2003. Requirements Use Case Tool (RUT), Companion of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, pp. 104-105.
OMG 2003, "UML Superstructure Specification", Object Management Group, http://www.omg.org/docs/ptc/03-08-02.pdf, 2003. [retrieved: October 2012].
Røstad, L., 2006. An Extended Misuse Case Notation: Including Vulnerabilities and The Insider. Threat. In: Proc. 12th Working Conf. REFSQ'06.
Sindre, G., Opdahl , A. L, 2000. Eliciting security requirements by misuse cases. In 37th International Conference on Technology of Object-Oriented Languages and Systems, pp. 120–131, Sydney, Australia.

APPENDIX

Table 2: E-BNF grammar of the Ex-SMCD structure.

```
S ::= UseCaseDescrption+ Actor+ MisuseCaseDescription* Miuser*

Actor ::= 'Actor:' Name Specializes?

MisuserActor ::= 'Misuser Profile:' Name Specializes?

UseCaseDescrption ::=  NameSection
                                 AssociatedActorsSection?
                                 ExtendedSection?
                                 DescriptionSection?
                                 ExtensionPointsSection?
                                 MitigatesSection?

MisuseCaseDescrption ::=     MisuseCaseNameSection
                                     AssociatedMisusersSection?
                                     ExtendedMisuseCasesSection?
                                     DescriptionSection?
                                     ExtensionPointsSection?
                                     ThreatensSection?

NameSection ::= 'Use Case Name:'
                        Abstract?
                        UseCaseName
                        Implements?
                        Specializes?

MisuseCaseNameSection ::= 'Use Case Name:'
                        Abstract?
                        UseCaseName
                        Implements?
                        Specializes?

Abstract ::= 'ABSTRACT'
```

Table 2: E-BNF grammar of the Ex-SMCD structure (cont.).

```
Implements ::= 'IMPLEMENTS ' Name*

Specializes ::= 'SPECIALIZES ' Name*

AssociatedActorsSection ::= 'Associated Actors: ' Name *

AssociatedMisusersSection ::= 'Associated Misusers: ' Name*

ExtendedSection ::= 'Extended Use Cases: '
                              UseCaseExtensions*

ExtendedMisuseCasesSection ::= 'Extended Misuse Cases: '
                                               MisuseCaseExtensions*

UseCaseExtensions ::=   'Base UC Name: ' Name
                                   ('Extension Point: ' EPName)?
                                   IfStatement?

MisuseCaseExtensions ::=     'Misuse Case Name: ' Name
                                      ('Extension Point: ' Name)?
                                      IfStatement?

AssociatedActorsSection ::= CharactersAndOrDigitsOrInclude*

DescriptionSection ::= 'Basic Flow:'
                                   CharactersAndOrDigitsOrInclude*

CharactersAndOrDigitsOrInclude ::= IncludeStatement |
                                                     CharactersAndOrDigits

IncludeStatement ::= 'INCLUDE <' Name '>'

ExtensionPointsSection ::= 'Extension Points: ' Name*

MitigatesSection ::= 'THREATENS ' Name*

ThreatensSection ::= 'MITIGATES Name*

IfStatement ::= 'IF' Condition

Name::= CharactersAndOrDigits+

CharactersAndOrDigits ::= Character | Digit

Character ::= 'a'| 'b' |...| 'z'| 'A'| 'B'|...| 'Z'

Digit ::= '0'| '1' |...| '9'
```

SHORT PAPERS

Supporting Collaborative Product Development through Automated Interpretation of Artifacts

Matheus Hauder[1], Sascha Roth[1], Florian Matthes[1], Armin Lau[2] and Heiko Matheis[2]
[1]*Lehrstuhl für Informatik 19 (sebis), Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany*
[2]*DITF Denkendorf, Centre for Management Research, Koerschtalstr. 26, 73770 Denkendorf, Germany*
{*matheus.hauder,roth,matthes*}*@tum.de*, {*armin.lau, heiko.matheis*}*@ditf-mr-denkendorf.de*

Keywords: Product Development, Collaboration Model, Business Modeling, Business Process Coordination.

Abstract: Small and medium-sized enterprises collaborate with partners to develop highly innovative and knowledge-intensive products. Collaboration models describe in detail how this goal can be achieved. These models can be supported by an implemented and executable version thereof. This paper describes an implementation of a collaboration model from the European research project SmartNets using an automated interpretation of artifacts. We explain the collaboration model, describe the implementation and evaluate our proposed solution by means of an example from one of the projects industrial networks.

1 INTRODUCTION

Increasing complexity of products and processes, condensed product life-cycles, quickly changing market requirements and globalized competition are major challenges that companies, and in particular European small and medium-sized enterprises (SMEs), have to face. The key to mastering these challenges is an effective and efficient development, production and marketing of new products, processes, and services. For many SMEs, due to their limited availability of resources and their concentration on core competences and niche markets, it is very difficult to elaborate such innovations from the idea to the final product completely on their own. Therefore, more and more, SMEs collaborate with trusted partners in loosely coupled development networks to combine their knowledge and resources and to share the associated risks (Mazzarol and Reboud, 2008). Even though there are several examples of success, the lack of appropriate methods and tools to support conjoint innovation within networks is still hampering the effectivity and efficiency of such collaboration (Dooley and O'Sullivan, 2007). The objective of the European research project SmartNet is to develop and to evaluate methods and tools supporting collaborative development and production networks in the product development, in network coordination and in the transformation from development to production[1]. For that, SmartNet follows the concept of the smart organization (Filos, 2006).

[1]https://www.smart-nets.eu/, last accessed Dec. 6, 2012

Social software solutions as information and communication technologies could play a crucial role as enabler for collaboration in future. They support the conjoint documentation, exchange and elaboration of information, and with the help of appropriate ICT systems, process models can be implemented and enacted, promoting and supporting the use of these processes in the daily work. Although social software provides manifold possibilities for a better support of these challenges, current research is mainly focused on the use of social software for advertising and marketing purposes (van Osch and Coursaris, 2013). We argue that social software can be used to endorse a better support for collaborative product development.

In the following, the SmartNet Collaboration Model, a model for collaboration in development and production networks developed in the course of SmartNet will be introduced. It will be demonstrated how the model has been implemented on the projects collaboration platform Tricia. Special attention will be given to the SmartNet Navigator, a tool to assess the process status of a development project. This processes status is automatically computed with an own substitution language that is utilized to present the current progress of the involved artifacts in an interactive visualization. Its application in collaborative product development will be evaluated with an example from a project dealing with the conjoint development of an innovative motorcycle helmet.

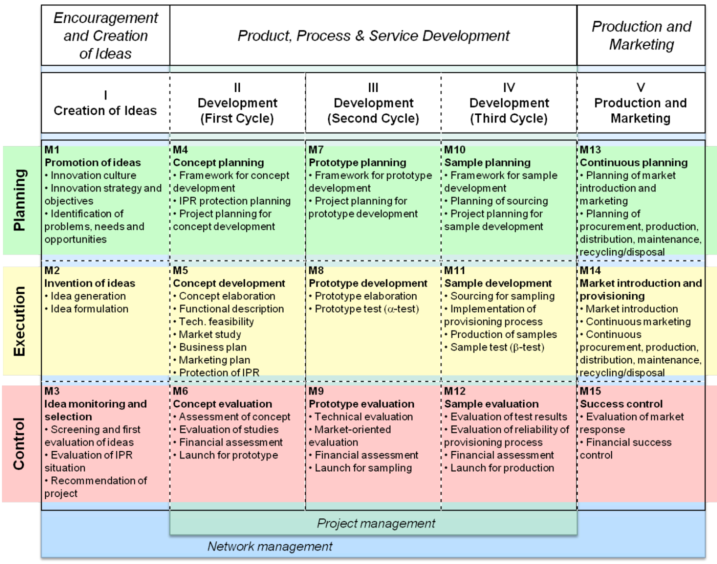


Figure 1: Reference process - key element of the SmartNet Collaboration Model (Lau et al., 2012).

2 SMARTNET COLLABORATION MODEL

In the operative planning, execution and control of collaborative innovation activities, there are several questions, which will continuously come up throughout the development. Six generalized key questions have been identified which will emerge in one or another form (Lau et al., 2009):

- *What* do I have to do, and when shall I do it?
- *How* can it be done?
- *Who* can do it?
- *How* can I do what, and when?
- *What* shall I do when and with whom?
- *How* can I collaborate with whom?

The SmartNet Collaboration Model provides answers to these questions by connecting three key elements of collaborative development: process model for development and production, internal and external actors, innovation management methods and techniques (Lau et al., 2009). Each element is described in detail with numerous instances, e.g. with a list of roughly eighty actual methods and forty-eight process activities. Figure 1 shows the complete reference process, to which we will refer later on when discussing the particular part of the SmartNet Navigator.

The relations between these three elements will be attributed with values indicating support potential, i.e. for which step in the process, which kind of actor is best suited to be involved (Lau et al., 2012). A dedicated processing model is provided which explains how these structures can be applied in the practice of collaborative networks (Lau, 2012).

2.1 Implementation of the Model on a Collaboration Platform

We implemented the above-described collaborative process utilizing the Hybrid Wiki extension (cf. e.g. (Matthes et al., 2011)) of the Enterprise 2.0 platform Tricia. Hybrid Wikis are based on a non-rigid typed system. They allow users to alter respective models at runtime by modifying the contained data on which it is based. Attributes and entities can be added or removed by end-users in such an intuitive manner that the actual process of modeling, steps into the background and users are first and foremost concerned about their data (data first, schema second).

2.1.1 Tricia - Hybrid Wiki for Enterprise Collaboration

With Hybrid Wikis, collaborative work that is commonly based on a model in particular can profit from the lightweight modeling approach. Hybrid Wikis do not require a special syntax for modeling or contributing data, i.e. everyone may add and manage information/knowledge, which is commonly referred to as crowd sourcing. Besides the typical unstructured content, a Hybrid Wiki page also includes structured content consisting of key value pairs (e.g. management activity type and process phase in Figure 2). By default, an attribute added by an end-user is of the type string.

The type can also be assigned to a value explicitly, which can be regarded as the schema definition. A small icon appears to indicate that this is a type defined by a so-called type definition. The actual type assignment of a wiki page is as simple as adding ordinary tags to a web 2.0 page. By assigning such a Type Tag, the type definition refers to this concept (here: activity type). In such a definition, each attribute may be assigned a type, i.e. string, number, date, enumeration, or a link to another page. In the latter case, also the type of the target page can be specified. With these type definitions, Hybrid Wikis cover the range from not at all typed to strictly typed (mandatory attributes with cardinalities) meta-models as in common relational database systems. Once defined, the type definition is applied to all instances of this type of pages.

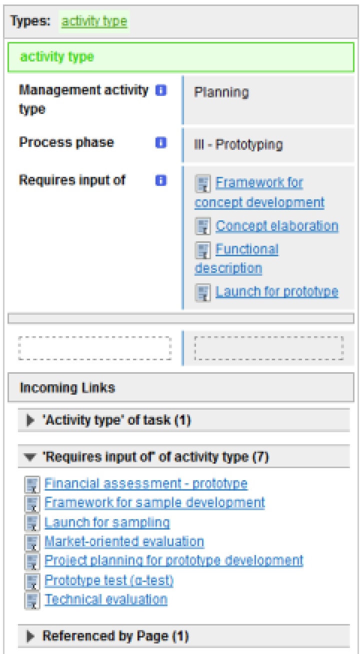


Figure 2: Attributes of the activity type.

2.1.2 Elements of the Collaboration Platform on Tricia

For the implementation of the SmartNet Collaboration Model the extended functionalities of Hybrid Wikis are essential. These above described functionalities enable a contextualization of simple pages to combine textual descriptions of process activities, methods and actors and relations between these entities. The implementation of the collaboration model was started with the reference process in Figure 1 because of its guideline characteristic. Every single activity that is mentioned in this process is implemented as a page of a special type, which is called *activity type*. All activity types consist at least of an explanation on what this activity is about and its position in the reference process. The position in the reference process is described by the process phase to which an activity belongs and by its management activity type from other activity types to identify possible previous steps in the process. Accordingly, reverse relations that can be found under *Incoming Links* in Figure 2 point out potential following steps. The storage of that information enables the implementation of a kind of workflow to suggest possible next steps on the basis which activity types were already finished. This kind of structuring allows adapting the reference process very flexibly to specific needs of different fields of application. Due to the fact that these activity types are one of the key elements of the collaboration model these adaptations affect all other derived tools like the SmartNet Navigator, which is deeply related to the reference process. The second element of the collaboration model, which is already implemented on Tricia, is a methodology to support the development and the collaboration in the networks. In a first step, around eighty methods - suitable for the application in enterprise networks and across companys borders - were identified and described on the project platform. These descriptions consist of basic information about the methods with strengths and weaknesses and application guidelines to support the project partner in implementing the methods into their development projects. Also actors are implemented in a similar approach as type tag with various attributes, identifying mostly their positioning in the supply chain. By this, it will be possible to extract information about actors from models of the network topology of the project networks. Finally, the relations between the key elements of the collaboration model were built. For that, more information is required. For example, to describe methods sufficiently, it is necessary to identify:

- Which results can be reached by applying the methods,
- The activity types for which the methods are suitable, and
- Their contribution to knowledge management, including required and created knowledge.

All this information was documented using the hybrid attributes of the Tricia system either by building relations to other information (e.g. for the different degrees of suitability), using prepared enumerations (e.g. to categorize the method type and the build-

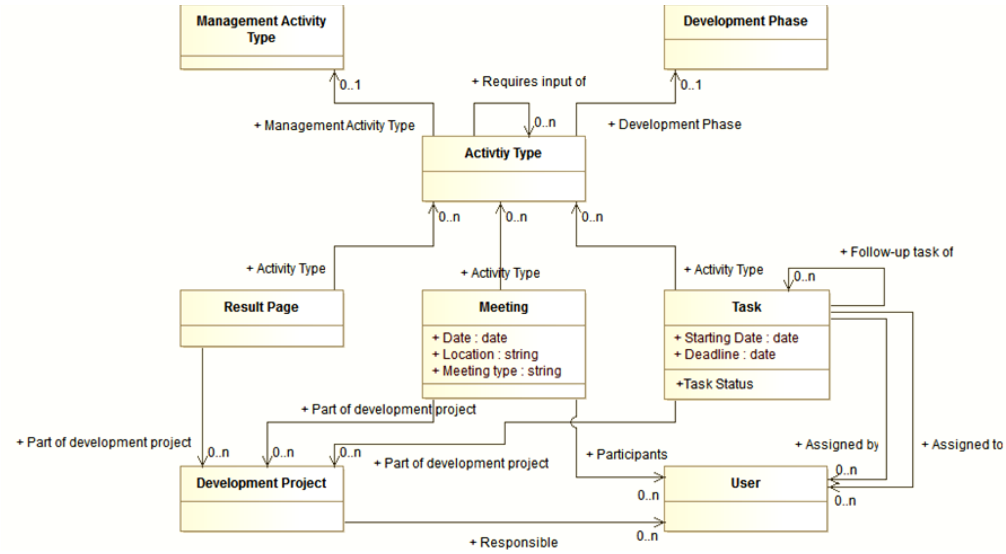


Figure 3: Meta-model of the described SmartNet collaboration model.

ing blocks of knowledge management (Probst et al., 1998)) or normal text (e.g. to specify input and output knowledge). By using all this data it will be possible to answer the six key questions presented in Section 2 in future.

2.1.3 SmartNet Navigator - Tool for Automated Status Analysis

The automated status analysis is performed using a substitution language developed within the Hybrid Wiki project that can be used to query the data (attributes) of the collaboration model. Their computation is based upon a meta-model that is shown in Figure 3. The collaboration model consists of several meetings that are part of a development project. Meetings can contain activity types that might require preceding activities as input. They can also be related with a management activity type, e.g. project management, network management, and a development phase, e.g., production and marketing.

An example instance for an activity type is shown in Figure 2. Result pages are used to document the findings from different activity types in a development project. Activity types also contain tasks that are assigned to respectively by users. Every task contains a *starting date* and a *deadline* until the task has to be finished. The progress of the status is tracked with an attribute *task status*. In addition, a development project can have several users who are responsible for the project.

Figure 4 shows an example for the computation of states for the process phase. Variable *allStati* is computed containing all states from a particular process phase by selecting all *activitiy types* from this phase. The state of an activity is retrieved with a function called *statusOfActivityType*. In the second variable valid combinations for the visualizations of the resulting state are stored. Only in case both values are finalized (or open) the aggregated value is finalized (or in-progress). In case the variable *allStati* is empty, the process phase is regarded as finalized. Alternatively, the variable is not empty and a higher-order function is used to compute the aggregated value of all containing activities.

3 PRACTICAL APPLICATION

The SmartNet Navigator is applied in all three industrial networks of the SmartNet Project. From the beginning of the project all partners started documenting their development activities such as meetings and tasks by using the Tricia platform. Using the possibilities of the hybrid attributes, provided by the platform, enables the industrial partners to extend the basic information with additional data as for instance starting date and deadline for tasks and participants and location for meetings. In addition to this very content specific information they related all their development activities on the one hand to certain development projects and on the other hand to activity types.

The following chapters give an exemplary view of the situation in one of the industrial networks with a short description of the network and the targeted product and the realization of the SmartNet Navigator.

statusOfProcessPhase

```
let allStati = find("activity type", "Process phase", ph)
  .select(?(at) (this.statusOfActivityType(at))) in

let combine = ?(s1,s2) (
   s1.equals("finalized").and(s2.equals("finalized")) ? "finalized"
      : s1.equals("open").and(s2.equals("open")) ? "open" : "in-progress") in

allStati.isEmpty() ? "finalized" :
allStati.aggregate(combine, allStati.first())
```

Figure 4: Code listing computing states of the process phase executed when loading the SmartNavigator.

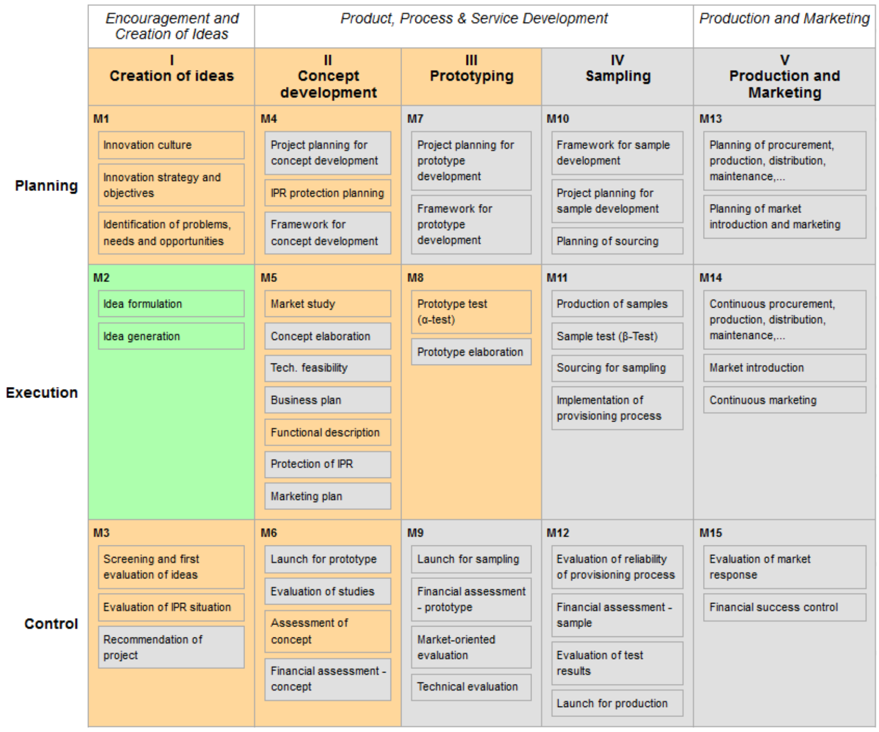


Figure 5: The SmartNet Navigator of the Innovative Helmet Network with an automated interpretation of artifacts.

3.1 Introduction to the Innovative Helmet Network

The Innovative Helmet Network (IHN) mainly consists of three SMEs that are supported by a research organization and a consultancy. Together they are developing a new motorcycle helmet, which will be made out of completely new materials with special properties. This new helmet will reach a new level in protection and will give more freedom to the designer especially in shape and size. Besides the development of this new helmet, the network also develops new production technologies and processes to increase the effect of the new material.

To handle the arising challenges, the core of the network consists of a chemical company which is responsible for the development of the material and its processing, a product engineering and manufacturing company which has a lot of experience in the development and production of protection gears for different fields of application and which will be the manufacturer of the final helmet.

The third partner in the network is an engineering consultancy with main focus on technical design, which supports both other partners and coordinates the collaboration during the development phases. Both associated partners support the network in general issues, e.g., management and protection of intellectual property rights (IPRs), the application of innovation management methods and the SmartNet Collaboration Model.

3.2 SmartNet Navigator Supporting the Development Network

Product development is a complex and challenging topic in particular in networks of several enterprises. In this network context the enterprises need some space where they are able to collaborate, share knowledge and information, document and coordinate their work with each other. For this specific network that participates in the SmartNet Project the partners use the project collaboration platform 'Tricia' to face all these issues.

Typically for the industry sectors, documentation will only be done if it is either recommended or required due to quality or risk assurance or the enterprises really can benefit from it. Thus, the SmartNet Navigator has to consolidate the documentation of all development activities to show the current status of the collaborative development project. This automatically interpreted status supports the coordination of the collaborative product development project.

For the IHN this means that all meetings, tasks and results that were documented on the project platform are related to activity types and to the development project they contribute to. Based on the rules exemplarily described in Section 2.1.3 the SmartNet Navigator application calculates the current status of the development project and visualizes (cf. e.g. (Hauder et al., 2012)) the result as shown in Figure 5.

This example shows that the IHN has collected and formulated their ideas (M2 is finalized; colored green) on how this new motorcycling helmet should protect its bearer. They are currently working on the processing of the new material to set up several prototypes of the complete helmet. Parts of it like the inner shell have already been tested against the homologation criteria (see Prototype test (α-test) in Figure 5).

Obviously, the SmartNet Navigator in Figure 5 reveals several open (grey) or in-progress (orange) modules and activity types owing to insufficient documentation or switching between several activities and forgetting to finalize or start them. In both cases the SmartNet Navigator reminds the team on the one hand to finalize the activities and on the other hand to collect all information they may need in later activities when it is easy to get them. A second big advantage for the network or single decision makers is to get a quick overview on the current status of the development.

4 CONCLUSIONS

The SmartNet Navigator helps enterprises acting in collaborative networks to identify states of development projects. Network partners highly esteemed this aggregated view, as it gives a good overview at a single glance and serves as a nice instrument to evaluate the information basis. The information about the current state will be used for further implementation of tools supporting the application of the collaboration model itself, e.g. by using the information to propose methods and techniques of innovation management to the users or to recommend network partners.

REFERENCES

Dooley, L. and O'Sullivan, D. (2007). Managing within distributed innovation networks. *International Journal of Innovation Management (ijim)*, 11(03):397–416.

Filos, E. (2006). Smart organizations in the digital age. *Integration of Information and Communication Technologies in Smart Organizations. Idea Group Publishing, Hershey*, pages 1–38.

Hauder, M., Matthes, F., Roth, S., and Schulz, C. (2012). Generating dynamic cross-organizational process visualizations through abstract view model pattern matching. *Architecture Modeling for Future Internet enabled Enterprise*.

Lau, A. (2012). *Methodisch-technologische Unterstützung kollaborativer Innovationsprozesse in Smart Networks*. Verlag Dr. Hut.

Lau, A., Fischer, T., Hirsch, M., and Matheis, H. (2012). Smartnet collaboration model-a framework for collaborative development and production. In *Engineering, Technology and Innovation (ICE), 2012 18th International ICE Conference on*, pages 1–10. IEEE.

Lau, A., Fischer, T., Rehm, S.-V., and Hirsch, M. (2009). Collaborative innovation in smart networks–a methods perspective. In *2009) Proceedings of the 15th International Conference on Concurrent Enterprising: ICE*.

Matthes, F., Neubert, C., and Steinhoff, A. (2011). Hybrid wikis: Empowering users to collaboratively structure information. In *6th International Conference on Software and Data Technologies (ICSOFT)*.

Mazzarol, T. and Reboud, S. (2008). The role of complementary actors in the development of innovation in small firms. *International Journal of Innovation Management (ijim)*, 12(02):223–253.

Probst, G., Raub, S., and Romhardt, K. (1998). *Wissen managen: wie Unternehmen ihre wertvollste Ressource optimal nutzen.*, volume 2. Aufl. Gabler, Wiesbaden.

van Osch, W. and Coursaris, C. K. (2013). Organizational social media: A comprehensive framework and research agenda. *Hawaii International Conference on System Sciences HICSS 46*.

ERP Implementation Success through Effective Management of Roles and Responsibilities among Stakeholders

A Holistic Framework Adopted from Two Case Studies

Waiel Al-Rashid, Mustafa Alshawi
University of Salford, Manchester, U.K.
walrashid@yahoo.com, M.A.Alshawi@salford.ac.uk

Majed Al-Mashari
King Saud University, Riyadh Saudi Arabia
majed@ksu.edu.sa

Keywords: Enterprise Resource Planning, ERP, Stakeholders, Roles and Responsibilities, Public-private Partnership, Return on Investment, ROI.

Abstract: This paper describes a case study of Enterprise Resource Planning (ERP) implementation in a public-private partnership organisation. The primary focus of the study is to explore, from a holistic view, how Return On Investment (ROI) can be achieved through effective management of roles and responsibilities in an ERP implementation context. The paper starts with an introduction, which includes highlights from the literature indicating the significance of roles and responsibilities management among various ERP stakeholders. The introduction is followed by a brief description of the research methodology used, and then followed by a description of the chosen case study for this paper. The case is then analysed with more focus on how roles and responsibilities among ERP stakeholders interrelate with the implementation outcomes. A separate section is dedicated to extracting appropriate lessons that improve ROI from ERP investment. The findings from the case are assessed through the literature and a published case study that is addressed to ERP stakeholders. The result of discussing the case findings will be presented in a new version of the management framework for the stakeholder's roles and responsibilities in the context of ERP.

1 INTRODUCTION

ERP is a system that consists of an integrated commercial software solution which fulfils the function of uniting the different functions of an organisation (Ifinedo & Nahar, 2009). One of the key benefits which are perceived to be associated with the implementation of an ERP system is the fact that it enables managers to have a holistic perception of the workings of the organisation and, as a consequence, the popularity of such system implementations has grown rapidly. Indeed, the benefits which are associated with ERP systems are perceived to be so strong that ERP systems have been widely heralded as one of the most significant developments to have taken place in the field of organisational information systems within the last decade (Grabski, Leech and Schmidt, 2011).

Al.Rashid, Al.Shawi and Al.Mashari (2009) review relevant literature and found ERP stakeholder perspective is a wide research area where authors suggested studying relationships amongst stakeholders at all implementation levels as well as the success of the implementation.

Research which has been conducted by Arlbjorn and Haug (2010) suggests that the key reason for the failure of ERP implementations is due to the fact that the methods used to manage the human issues which arising from the ERP implementation are ineffective.

According to research which has been conducted by Madhani (2012), the implementation of a new ERP system can result in significant changes in the management processes, culture and structure of an organisation and the success of the implementation of a system therefore requires a holistic view of such changes to be adopted. Murphy, Chang and Unsworth (2012) has conducted a series of case studies to argue that an insufficient amount of attention has been dedicated to managing the effects that the new ERP system is likely to have on the culture of the organisation. This work has given rise to recommendations that a more cohesive and more comprehensive approach is adopted, which acknowledges and manages the changes that the ERP system has on the entire range of stakeholders which exist within the organisation.

However, despite the array of advantages which are associated with the implementation of ERP systems, this is belied by the large number of high profile ERP failures which have been discussed within the media. For example, in a survey which was conducted by Murphy et al (2012) of 250 companies, it was found that the proportion of companies who stated that they were 'very satisfied' with their ERP system was just 10 per cent in 2010, while the number of companies who claimed that they were 'very dissatisfied' with the outcome of their ERP system rose significantly from 2 per cent in 2006 to 31 per cent in 2010 (Tiwana & Klei, 2010).

There appear to be a number of reasons why the reputed benefits which are associated with ERP systems fail to materialise for many companies.

A study by Burns (2008), that sought to discover the ten most frequent selection and implementation mistakes of 2007, found that clearly defined roles and responsibilities were amongst the key factors vital to the success of ERP implementation.

A recent survey study of ERP implementation in the retail sector in India, conducted by Garg and Garg (2013) found evidence that the management of stakeholders' roles and responsibilities was crucial to success.

In fact a review of the literature regarding critical success factors (CSF) in ERP implementation, conducted by Finney and Corbett (2007), resulted in a key finding being made, namely that there was a huge gap in the literature regarding what key stakeholders perceived as the CSF that resulted in implementation success, which suggests that their views are not considered valuable nor is the effective management of their roles and responsibilities.

Al.Rashid et al (2012) attempted to study ERP implementation from a stakeholder's management through a case study of an agricultural organisation. The study is concluded by producing a framework (Figure 1).

Figure 1: Framework for the first revision of the roles and responsibilities among stakeholders. Source: Al.Rashid et al (2012) pp 70.

This framework is based on a single case study and limited to the preoperational stage of ERP implementation only. It did not address the two stages of the core and post implementation. Therefore, this paper aims to test the framework using another ERP implementation case study and to expand the framework to cover part of the issues needed for the other two implementation stages; i.e. implementation and post implementation. A new version of a holistic framework is then developed that can assist in improving ROI from ERP investment through effective management of stakeholders.

2 RESEARCH APPROACH

This paper has chosen another exploratory study of ERP implementation from a stakeholder's perspective, through qualitative data collection, to verify the applicability of the existing framework (Figure 1) and suggest how the framework can be enhanced.

ServCo is a given name for a public-private partnership organisation to manage water supply services. ServCo's experience in implementing ERP is studied in the context of effective identification and management of the roles and responsibilities of various stakeholders involved or affected by the implementation. The data collection followed a qualitative approach through a number of interviews with key personnel involved in the implementation at different stages. Other sources of data collection are used to complement and assess the data collected from interviews including various documents such as implementers' proposals, status reports, power-point presentations, minutes of meetings, etc. The data is then analysed and discussed through supporting related literature and the case study by Al.Rashid et al (2012). The discussion of findings from this case is concluded with a holistic framework of effective management of roles and responsibilities which can be considered as an enhanced and upgraded version of the framework (Figure1) that is suggested by Al.Rashid et al (2012).

3 CASE DESCRIPTION

A government decree is announced in a developing country requesting Ministry of Water to privatise water services. To prepare a new company for business, the Ministry started the development of support finance, HR and logistic policies and procedures for the new organisation including the initiation of supporting information systems.

The CEO decided to adopt the same ERP solution that has been just implemented in the Sewage organisation (another entity governed by the Ministry).

The purchasing department in the Ministry of Water produced a public Request For Proposal (RFP) for an ERP service, where a number of IT consulting companies applied. The technical committee awarded one implementor (I-a), who had just completed ERP implementation in the water desalination organisation. The contract, also developed by the Ministry contracting department, states that 'I-a' should complete the implementation in eight months in the centralised region only before adding a second region that has recently fallen under ServCo's responsibility.

A month later, ServCo announces that it has signed an operations and maintenance (O&M) agreement with the technical partner to manage the operations of the second region. Gradually, I-a became more and more frustrated because the requirements for changes never stopped, and they were unable to complete the remaining implementation activities.

3.1 Operations & Maintenance (O&M) Partnership

The ERP team was not aware of the contract signed between ServCo and the O&M partner. Six months from the O&M signing date, the ERP teams received several enquiries about the ERP implementation status. The ERP team discovered that ServCo appointed the O&M partner to manage the operations and maintenance by which they need an ERP system to facilitate their activities. ERP project manager explained clearly that ERP plans did not include any consideration of the new requirements of the O&M partner. A formal report by the O&M partner submitted to ServCo stating clearly that the ERP implementation is significantly delaying the O&M handover plans. The report includes specific rectifications and a road map that includes several alternatives for ServCo to recover the situation by aligning ERP with the O&M plans. One week later, the ERP project manager circulates a memo that sets out clearly a fast-tracked ERP release to fulfil the partnership agreement with the O&M partner.

During the ERP fast-track implementation process, a new CIO is hired to manage all information systems requirements. The first priority assigned to him is to assure all O&M information systems requirements are fulfilled including consistency and integration with ERP implementation. The CIO takes the lead in capturing all necessary tasks needed to meet this mission and starts a new discussion with the implementor 'I-a'. The implementor responds that these requirements are new and require a new implementation assignment. The CIO takes a firm stand and decides to black list 'I-a' and refuses to release their remaining payments. This conflict with the existing implementor urges the CIO to find an alternative implementation partner. The implemented modules are self-explained in table 1.

Table 1: Implemented modules in Phase I.

Areas	AS IS Modules
Finance	*G-Ledger & Account Payable*
Supply Chain	*Inventory & Purchasing*
Human Resources	*HR & Payroll*

A new implementor 'I-b' is hired for a six month contract to provide post-implementation maintenance and support services for selected ERP modules.

3.2 The Emergence of the Third ERP Implementor

In parallel to the assignment of 'I-b', the CIO is developing a long-term plan that can assure fulfilment of O&M ERP requirements. An International implementor is hired to conduct a total assessment review of the existing implementation gaps. The process starts by stating a number of objectives that include evaluating the actual modules/functionalities installed in ServCo to assess of the current usage of the system, functions activated but not used, functions not used and recurring issues. The assessment includes also evaluating the completeness of current implementation for the implementation of the new module requested for phase II.

The assessment is concluded by a detailed report that includes gap analysis for all implementation areas. Those gaps were categorised based on the severity level (Table 2).

Table 2: Analysis of the Implementation Gap Categories Distribution.

	Severity			
Area	**H**	**M**	**L**	**Total**
All		1	1	2
Finance	2	34	10	46
HR	1	9	47	57
SCM	4	4	2	10
Finance & SCM	2			2
Total	9	48	60	117

Two months after ServCo reviews and discusses the assessment review the 'I-c' is awarded to implement the third cycle. ServCo & 'I-c', who are announced as the strategic partner in information technology, celebrate the new project, which will significantly contribute towards raising the efficiency and the quality of the services offered by ServCo. The new project is introduced as a global initiative that seeks to transform and enhance the way ServCo operates its business and delivers its service to customers. More specifically, the transformation is claimed to arrive through unified, lean and robust business processes and state-of-the-art technology. Customers are expected to benefit from the project, through a Customer Care & Billing system. Also employees have been promised that they will benefit from the project, thanks to the new maintenance processes and the implementation of the enterprise asset management system. The implementation starts with close coordination between the two project managers i.e. from ServCo and from 'I-c' where each project manager facilitates the resources and services required by the implementation. In parallel with the normal implementation process, the change management team sets up a ServCo college to take care of all the required training. The college trains nearly twenty five trainers who take on the training of the end users. The implementation is completed on-time and 'I-c' advertised the perceived implementation benefits are achieved.

4 ANALYSIS

The analysis of the case suggests implementation is to be divided into three phases (Table 3).

Table 3: Implementation phases.

No	Implementation Description
1st Phase (I-a)	Preparation activities that include the development of the policies & Procedures, selection process and contract development. Core implementation of the As-Is of ERP modules based on the policies & procedures developed in the first phase.
2nd Phase (I-b)	A new implementor hired to provide support and to renovate the existing Implementation
3rd Phase (I-C)	The advance implementation that covers all user requirements and the operations and maintenance partner considerations.

4.1 High Level Stakeholders Identification

The high level stakeholders involved at the initiating ERP implementation process are the Ministry of Water, the Ministry of Finance and the Operations and Maintenance partner (Figure 2).

Figure 2: High level stakeholders of the first Implementation Cycle.

The Ministry of Water at the time of adopting ERP is the sponsor of water services in the whole country. Therefore, ServCo business practices are dominated by inherited government culture and practices. The Ministry of Finance governs all financial transactions across ministries. Its mandates include the management of the expenses and revenues of all ministries. This includes the unified government purchasing system that is used to award the first implementor 'I-a'.

For newly set up companies purchasing sophisticated systems like ERP, government purchasing systems can be seen as inappropriate since the stakeholders involved in the purchasing process lack adequate knowledge to make ERP purchasing and contract decisions.

This can justify why the contract and the scope of the first implementation can be seen as inappropriate to the nature of ERP implementation. It is obvious that stakeholders involved from both ministries including purchasing and contracting departments have dealt with ERP as an off-the-shelf software type of product. No considerations are made towards the need to review and improve the business process for ERP to succeed. For example, the process of managing and organising fragmented warehouses is underestimated and hinders several ERP functionalities from being used due to the huge amount of data that needs to be prepared and the processes that need changes.

Besides the knowledge gap within different stakeholders involved from the two ministries, there is a clear roles and responsibilities gap at that stage. A clear example is why the ERP team disengaged from the O&M partnership agreement.

In fact, the ERP implementation team may be shocked with the magnitude and un-criticality of the operations and maintenance partner's requirements that should have been thoroughly considered at early implementation phases. The proposal by the O&M partner indicates clearly before the contract is signed that ERP is expected to be ready before operations are started. However, the purchasing and contracting stakeholders who prepared the contract never communicated this to ERP team in a timely manner. All stakeholders drift from bearing the responsibility of such a mistake Figure 2 demonstrates that ERP is centred on three stakeholders who lack consistency and integrity in their requirements.

That poor management of roles and responsibilities has cascaded down through lower levels to line managers and end users who become part of the dilemma because of the lack of clarity in the definition of their roles and responsibilities. During the requirements definition, the response from end users was slow and incomplete. This behaviour by end users increased the frustration of 'I-a' as this is expected to delay the implementation. End users stated that they received conflicting directions from two parties.

The first party is their line managers who tried to comply with ERP implementation instructions and guidelines, while the second party is the operating and maintenance partner who used their power from the mandate obtained from the contract.

This scenario that first started from poor management of roles and responsibilities resulted into ERP failure during first implementation cycle (Figure 3).

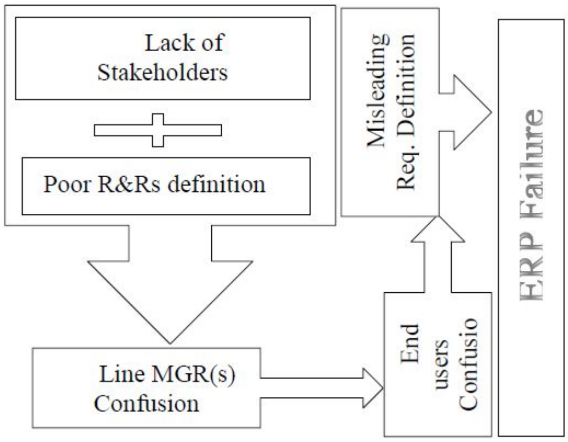


Figure 3: Relationship between R&Rs & ERP failures.

4.2 Policies & Procedures Focus

The attempt at gaining a head start by developing policies and procedures for ServCo through the use of management consultants has heavily affected the implementation.

Those policies and procedures become the primary source for the definition of ERP requirements. When functional managers are hired by ServCo at a later stage they found that large requirements gaps are missing from the current configuration. End users found HR modules inflexible to cope with real business practices. This finding can be seen in the tendency by several end users to manage core business operations outside the system as much as they could, which defeated the point of adopting such advanced systems like ERP.

The driving forces of expediting and developing policies and procedures before hiring line managers are understood. However, the complete ignorance of key stakeholders in the requirements definition by relying only on policies and procedures can be seen as an explicit mistake. Policies and procedures are as good as business owners understand and use them in practice. However, in ServCo's case not only were business owners disengaged from the development process of those policies and procedures but also it had never been used in the company. ERP modules, therefore, are designed on a very weak base as regards defining the requirements. As a result, the outcome of the first implementation cycle is poor in terms of assisting HR practices, most of the issues are resolved outside the system and ultimately another implementation is initiated. The ROI from ERP investment is then significantly decreased since the return has to cover double of the cost.

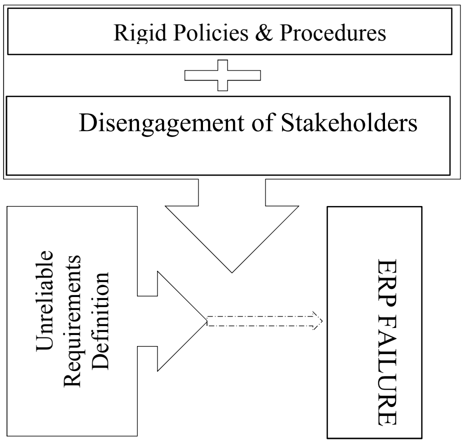


Figure 4: Relationship between disengagement of Stakeholders & ERP failures.

4.3 Can the Case Considered Successful or Failure?

The holistic review of the implementation process including the various implementation cycles leads to the conclusion that deciding whether the implementation is successful is problematic.

The second and the third implementation cycles can be considered successful. Both projects are completed on time, according to budget and the objectives are achieved. On the other hand, the experience of the first implementation cycle was one of failure. The project was not completed, the cost and the time is over run. The combination of three cycles indicates that ultimately ERP achieved most of its target benefits however ROI is significantly less than what it should be as ServeCo invested in two extra unplanned projects to achieve the same objectives.

Despite these implementation deficits, ERP implementation produced side-gains. First, the hard lessons from the first implementation cycles motivated the company to focus significant attention on change management. Second, the organisation learned the significance and the importance of managing roles and responsibilities to ERP success; which can be demonstrated by the examples of table 4 & 5.

5 LESSONS LEARNED

A number of lessons which fall under the category of the management of roles and responsibilities among stakeholders are discussed in the following sections

5.1 R&Rs Lesson from the Second Cycle

While analysing the project plan start-up activities the conclusion can clearly be drawn that the implementer has suggested a very clear, realistic and fair roles and responsibilities definition. The ServCo project manager demonstrated excellent leadership in assuring a full commitment to the definition of those roles and responsibilities. The second implementation cycle project manager offered the following description:

''The second implementor has successfully absorbed the legacy implementation issues, rectified all pending problems, implemented needed additions and produced excellent support services. The roles and responsibilities definition provided by the implementor has proven to be a prime success factor that paved the route for the third implementation phase success''

Table 4: Roles & responsibilities examples from 2nd phase adopted from ServCo project documents.

Role/Responsibility	Activities for the Role
ServCo Project Sponsor	1. Provide management sponsorship and direction to the project 2. Provide limited time for executive interview and review project progress 3. Chair the steering committee meeting
ServCo Project Manager	1. Conduct reviews and weekly status meeting 2. Engage with I-b support manager in decision making process around – support processes 3. Facilitate management decision and approvals. 4. Single point of contact for I-b team from communication perspective

5.2 R&Rs Lesson from the Third Cycle

The third implementation cycle can be seen as the most successful project. The roles and responsibility management have been developed and improved exponentially during this implementation cycle. Table 5 demonstrates how roles and responsibilities of the core implementation members have been defined.

Table 5: Roles & responsibilities examples from third phase adopted from ServCo project documents.

Team	Role	Time	Description
PMO	P/D	20%	Project Director will be involved also in planning and mobilising Finalisation (leverage Compass resource)
PMO	PM	60%	SERVCO PM in Plan phase is assumed high level of involvement based on the nature of the phase
PMO	PMO/A	40%	Support of PMO (leverage Compass resource)
Functional	BL	30%	Usually he is a selected and trusted representative of Business Users Functions, with corporate visibility and authority – He will act as gateway with BU for Planning of Workshop to be held in Analyse phase. He will act as the gateway for the user's community, facilitating communication and decision making.
Techno Team	IT	30%	Specific Skills in current IT capabilities supporting such processes. Able to discuss/ report/ communicate on current practices, and act as a catalyst/ change agent on to-be practices. In this phase just one at HQ level
Change Team	C/M	20%	Representative of HR department - To cover also the Change Management part – He will be assumed to be the gateway with Business Users for Training needs and schedule finalization and for the Communication Plan (leverage Compass resource)
Change Team	BA/C	40%	Assist the Change Manager in his daily work

PD: Project Director CM: Change Manager BA/Change: Business Analyst Change PM: Project Manager

6 DISCUSSION OF FINDINGS

The holistic review of the lessons learned indicates the significance of proper identification of stakeholders in the implementation at initiation stage of the project. ROI would have substantially increased if the proper stakeholder's identification had been made at the initiation ERP stage. The three implementation projects could have been reduced into one project with two phases. These ROI improvements would have been derived from shortening implementation cycles, reducing consultants' involvements and optimising the internal resources that had to be dedicated for ERP implementation for a long time.

This finding is in line with the study by Alrashid el at (2012) for an ERP implementation case in the agricultural field. In that case the implementation only achieved most of its targets in the third implementation cycle as a direct result of poor management and identification of the roles and responsibilities among ERP stakeholders. This is exactly the scenario in ServCo's case as the implementation achieved most of its benefits in the third implementation cycles. Both cases indicate that failure to follow the existing framework recommendations (Figure 1) in terms of proper identification of stakeholders and the roles and responsibilities agreement before deciding to start the implementation resulted in several implementation difficulties that adversely affect the ROI.

However, the case of ServCo provides additional contribution to the framework; which relates to the need to audit and review the application of roles and responsibilities among ERP stakeholders during the implementation process. More specifically, such an audit process needs to be conducted before the decision to go-live. Figure 4 shows how the two assessment reviews by qualified consulting firms assist in identifying ERP implementation deficits; consequently, ServCo amends the roles and responsibilities matrix among stakeholders in line with ROI targets. Such audit process needs to be included in the ERP project master plan. This is will assist ERP sponsors to use the audit outcomes to assess implementation outcomes and to take necessary rectification actions. Proper identification of implementation issues in a timely manner enables ERP sponsors to preserve ROI by freezing implementation costs and expediting ERP benefits.

7 DERIVING THE FRAMEWORK

The discussion of analysis of the three implementation cycles can be better represented by a framework of how effective management of roles and responsibilities among various ERP stakeholders can improve implementation outcomes. The derived model can leverage and be integrated with the framework (Figure 1) by Al.Rashid et al (2012) that covered the first part of the ERP initiation stage only. It can also combine the necessity of including the audit process before the go-live as explained in Figure 4.

The result of the audit process can either be positive findings where implementation can progress further to the go-live and lead to ERP success. A second probability, that the audit process indicates improvement opportunities where a rework process is needed and further rectifications are required before the next audit process can be conducted Figure 6. The integration between the prior research findings framework (Figure 1) and the extracted lessons of conducting an audit process (Figure 4) can be combined to produce an advance version of the framework (Figure 6). The framework can be divided into a number of groups.

Firstly, it suggests that ERP implementation should be divided into three phases, preparation, implementation and go-live & post implementation. Through the three phases approach ERP sponsors can intervene wisely to define and review the ROI and the roles and responsibilities. Two main points for their intervention are suggested between the three phases. The first point is at the time of signing the contract with the ERP vendors and announces the starting of ERP. The second review point is before the go-live where the audit process includes a full review of the ROI status and the management of the roles and responsibilities.

Secondly, the framework recommends a proper understanding of driving forces that brought ERP as a business case into the organisation. This is expected to pave the way for setting and defining the project scope and objectives before stakeholders can be assigned and made accountable for achieving the target ROI.

Thirdly, there are the preparation activities before the implementation starts, including proper planning centred on effective stakeholder's identification and concluded by producing the first roles and responsibilities agreement among various ERP stakeholders.

Fourthly, there is the core implementation activities process in which the framework is suggesting a comprehensive audit of the roles and responsibilities among all stakeholders before committing to the go-live, unless the audit confirms a successful roles and responsibilities review go-live should not be approved.

Finally, the implementation can safely progress to the go-live and post implementation plans where roles and responsibilities among ERP stakeholders by that stage are to be seen as an embedded part of the organisational culture. Once this is achieved; this can ensure optimal ERP ROI is always achieved during the continuous ERP improvements.

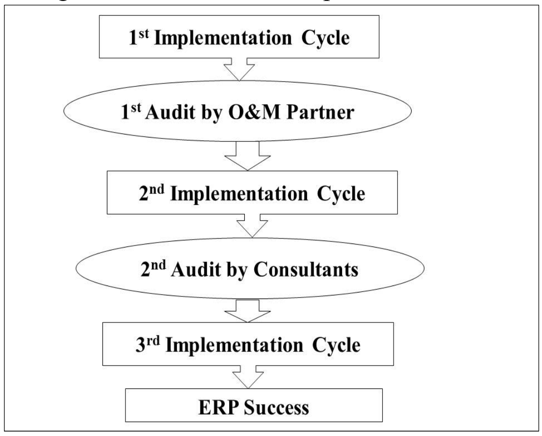


Figure 5: The audit process Vs. implementation cycles.

8 CONCLUSIONS

The paper has explored the research and conclusions from ERP literature that the management of roles and responsibilities among various ERP stakeholders is significant. Prior research suggests a framework centred on the management of stakeholders to improve ROI (Figure 1). That framework covers only the initiation stages of the implementation. A case study of a public-private partnership organisation is studied to verify the applicability of that framework and to extend it to cover the implementation and post implementation stages. The case has been described; analysed in a qualitative manner and related lessons have been extracted. The analysis shows, through examples from the case, how ROI is adversely affected as a direct result from the absence of the management of effective roles and responsibilities and the failure to implement the existing model at the initiation implementation stage. The findings of the analysis have been discussed using the existing framework from the previous case by Al.Rashid et al (2012). The paper found that ERP sponsors are urged to prioritise and carefully plan for a comprehensive audit process of the roles and responsibilities before deciding has to move to the go-live stage and discard legacy systems. The audit results assist ERP sponsors to take the necessary actions that rectify implementation mistakes and assure optimal results are achieved from the go-live and post implementation. The results of the findings and discussions have then been integrated with the existing framework in the form of a framework (Figure 6).

REFERENCES

Al-Rashid, W., Alshawi, M. and Al-Mashari, M. (2009), " Exploring Enterprise Resource Planning (ERP) Implementation From Stakeholders Perspective – Analysis of Case Study" , in *Proceeding of The 4th European Conference On Intelligent Management System In Operations* (IMSIO4) Greater Manchester, UK 7-8 July 2009 PP 140 -147

Al-Rashid, W., Alshawi, M. and Al-Mashari, M. (2012), " Exploring Enterprise Resource Planning (ERP) Implementation From Stakeholders Perspective – A Case Study" , in *Proceeding of The 5th International Conference on Communications, Computers and Applications* (MIC-CCA2012) Istanbul, Turkey: 12-14 October 2012 ISBN: 978-1-938302-07-7 PP 65 - 70

Arlbjørn, J. and Haug, A. (2010) Business Process Optimization. Academica.

Barkier, T. and Frolick, M. (2003) ERP implementation failure: A case study. Information Systems Management, 20, pp. 43-49.

Burns, M (2008), "Enterprise software survey 2008", CA Magazine Sep pp 14-15

Grabski, S. V., Leech, S. A. & Schmidt, P. J. 2011. A review of ERP research: A future agenda for accounting information systems. Journal of Information Systems, 25, 37-78.

Ifinedo, P. and Nahar, N. (2009). Interactions between contingency, organizational IT factors, and ERP success. Industrial Management & Data Systems, 109(1), 118-137

Madhani, Pankaj M., Enterprise Resource Planning (ERP): Enhancing and S ustaining Value (May 18, 2012). ICFAIUniversity Press, 2008. Available at SSRN: http://ssrn.com/abstract=1966815

Murphy, G. D., Chang, A. & Unsworth, K. 2012. Differential effects of ERP systems on user outcomes—a longitudinal investigation. New Technology, Work and Employment, 27, 147-162.

Poonam Garg, Atul Garg, (2013) "An empirical study on Critical Failure factors for Enterprise Resource Planning implementation in Indian Retail Sector", Business Process Management Journal, Vol. 19 Iss: 3

Sherry Finney, Martin Corbett, (2007) "ERP implementation: a compilation and analysis of critical success factors", Business Process Management Journal, Vol. 13 Iss: 3, pp.329 - 347

Tiwana, A. & Keil, M. 2009. Control in Internal and outsourced software projects. Journal of Management Information Systems, 26, 9-44.

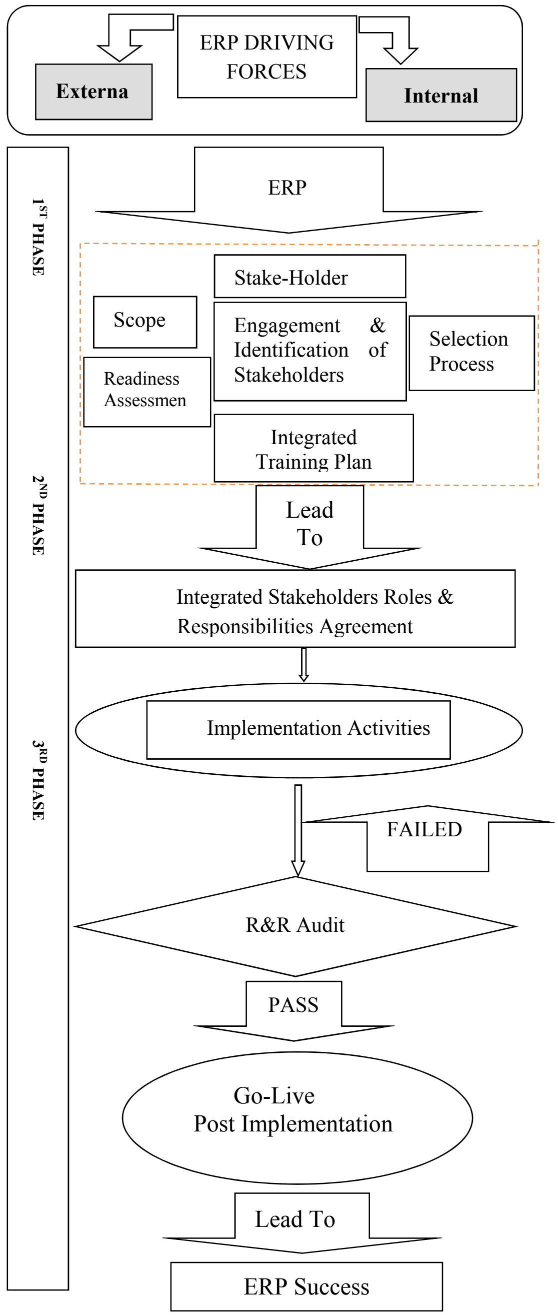


Figure 6: The framework for effective management of roles and responsibilities among stakeholders adopted from the case.

Parameter Estimation and Equation Formulation in Business Dynamics

Marc Drobek[1,2], Wasif Gilani[1] and Danielle Soban[2]
[1]*SAP UK Ltd., Belfast, U.K.*
[2]*Department of Mechanical and Aerospace Engineering, Queens University Belfast, U.K.*
{*marc.drobek, wasif.gilani*}*@sap.com, d.soban@qub.ac.uk*

Keywords: System Dynamics, Business Dynamics, Parameter Estimation, Equation Formulation, Non-linear Dynamic Equations, Machine Learning, Big Data, Modelling, Classification and Regression Trees.

Abstract: System Dynamics enables modelling and simulation of highly non-linear feedback systems to predict future system behaviour. Parameter estimation and equation formulation are techniques in System Dynamics, used to retrieve the values of parameters or the equations for flows and/or variables. These techniques are crucial for the annotations and thereafter the simulation. This paper critically examines existing and well established approaches in parameter estimation and equation formulation along with their limitations, identifying performance gaps as well as providing directions for potential future research.

1 INTRODUCTION

As the world increases in complexity, so too do the myriad systems that comprise it: from products such as mobile phones and automobiles, to large and small scale businesses, to our transportation system and even to climate change. These complex systems can be characterized as multi-dimensional, highly non-linear, and containing dynamic feedback. The field of System Dynamics has long been used to model, understand, and predict the behaviour of these complex systems. Business Dynamics, a specialized offshoot of System Dynamics, has been particularly successful in examining and analysing the complex business models of todays commerce (Sterman, 2000). For example, envision a business analyst whose goal is to try and predict the behaviour of customers, particularly their trend in returning to business. Using the Systems Dynamics approach, she starts by familiarising herself with the business, including all important processes and strategic goals. She then collects all influencing elements on the customer and connects them together to create a meaningful Systems Dynamics model. After some reiterations and further discussions with the process owners, product managers, and customers, she finally has a sufficiently accurate model to address the returning customer scenario. Up to this point, she has leveraged her skill and expertise in defining and understanding the problem, and has stayed well within her boundary of knowledge and capability. The next critical step, however, involves defining the parameter values and equations in her model, which drive the simulations. Even though she has access to stored business data, as well as some stock estimation techniques, she still needs to *manually* determine the parameters and equations in her model (Peterson 1976). This process has traditionally been found to be time consuming, cumbersome, resource intensive, and often necessitates a level of mathematical and technical expertise that may or may not be consistent with the analysts basic knowledge set of the initial problem. In addition, by virtue of this process being a manual one, the opportunity for error increases dramatically.

This process, called Parameter Estimation and Equations Formulation (PEEF), is arguably the most critical step in the entire modelling process, since it is key to reliable and sufficient system behaviour simulations. But it is also one of the most challenging tasks in the traditional System Dynamics process. This paper begins with a survey of the state of the art approaches to parameter estimation and equation formulation in a System Dynamics model. A detailed overview of these concepts is provided, and advantages and limitations are then summarized and discussed. The paper concludes by making a strong case for the automation of the PEEF process, in order to ultimately improve the overall efficiency, accuracy, and effectiveness of the System Dynamics approach.

2 BACKGROUND

This section contains a brief explanation of the System Dynamics concept. The authors introduce the eight step modelling process by Burns, and discuss PEEF.

2.1 Overview of System Dynamics Concepts and Modelling Processes

The concept of System Dynamics has been widely applied to a large variety of fields, be it the simulation and modelling of enterprises in ”Industrial Dynamics” and city growth in ”Urban Dynamics” as shown by (Forrester, 1961) and (Forrester, 1971), the world population in ”Limits to Growth” (Meadows et al., 1972), the System Dynamics National Model as simulation of social and economic change in countries (Forrester et al., 1976) or the decline of the Mayan empire in history (Hosler et al., 1977), among others. These systems under study are highly non-linear dynamic systems which are continuously changing over time. They consist of both static parameters, which never change during each simulation run, and variables which may or may not change during simulations. These parameters and variables are mostly interdependent, meaning that there are circular dependencies in the system under study. System Dynamics modellers incorporate the circular dependencies by modelling feedback loops to visualise cause and effect with causal loop diagrams (CLD) and/or state and flow diagrams (SFD). The figures 1 and 2 illustrate a CLD and an SFD representing an economic problem of returning customers, which also contains a feedback loop. A feedback loop in a system can either be characterised as balancing or reinforcing. Balancing loops drive the system behaviour sooner or later towards a steady state, thus equilibrium, whereas reinforcing loops emphasise the growth itself, either positive or negative in each iteration of the loop. The CLD and SFD are widely accepted in the System Dynamics community and support the modellers understanding of the system under study, as explained by Lane (Lane, 2000). Whereas the CLDs main purpose focusses on the identification of basic elements (quantities) and their connections (couplings) in the system under study, the SFD is used to map the system to a set of stocks (levels), rates, variables (auxiliaries), constants (parameters), flows and connections (information couplings). The SFD, furthermore, visualises the resources or materials flowing through the system under study. Such resources/materials are determined by the system and might be, for instance, money, pollution, population, water or customers as explained in

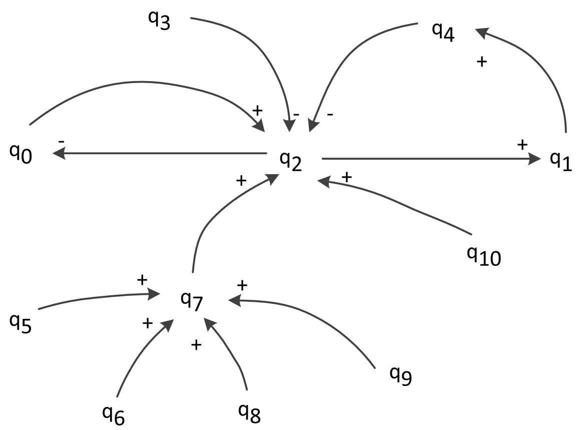


Figure 1: A causal loop diagram (CLD).

the introduction example. The previously mentioned System Dynamics model types CLD and SFD have been formally defined by Burns (Burns, 1977), who relied on the concept of set theory to provide formal definitions of all elements. The steps involved in modelling a specific system using the System Dynamics approach have been discussed for decades (Burns, 1977; Ford, 1999; Binder et al., 2004). Burns gave the following procedure:

1. Determine the concrete problem which shall be modelled and the system boundaries.
2. Identify quantities in the system which reflect the system (e.g. by considering smaller components of the system) .
3. Blue-print the causal diagram by addressing dependencies of previous defined quantities using a set of connections.
4. Migrate the causal diagram into a schematic (flow) diagram to highlight the resources or material flowing through the modelled system.
5. Formulate the model equations and estimate parameters with the help of the schematic diagram and expert knowledge.
6. Transform equations into a machine program to simulate the model.
7. Run the simulation and verify/validate the simulation output with observed or expected real world behaviour of the system.
8. Gain insights of simulation output, identify possible consequential policies and give client recommendations.

In the early days of System Dynamics, Forrester was for instance starting the modelling process with an SFD and used the CLD close to the end of a whole modelling process to summarise and visualise the dominant loops in the current model. But later

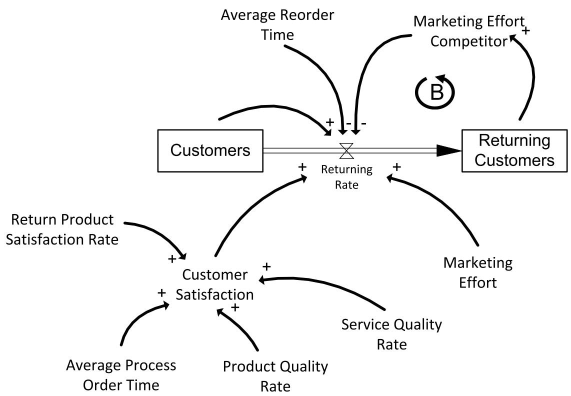


Figure 2: An economical state/flow diagram (SFD).

on it was stated by other researchers, e.g. Haraldsson (Haraldsson, 2000), that it might come handier to start with the CLD to get a better understanding of the involved quantities and their connections. Even though the order of the previously defined modelling process might change from time to time, the general process of modelling is still valid after more than 40 years.

2.2 PEEF (Parameter Estimation and Equation Formulation)

The System Dynamics modeller has to rely upon a huge knowledge base to identify the systems main connections and the way in which the quantities are influencing each other. Forrester stated that the quality of one model highly depends on the usage of all known information about the system under study (Forrester, 1991). This statement also holds for annotating a created SFD model with parameters and equations. He divided the types of available knowledge into three different classes, namely the mental data base (experience and knowledge of humans), the written data base (natural language text, written instructions) and the numerical data base (numbers in a table). It was stated in the introduction section that one of the critical phases in System Dynamics is the computation of parameters and equations in the model. There are existing approaches in place that address this phase. However, these approaches either lack automation, are very focussed towards specific problems or are unable to fully exploit all available data. Therefore, a clear potential exists for groundbreaking research on ”Equation formulation and parameter estimation”, which corresponds to step 5 in the traditional modelling process (see previous section). The authors believe that PEEF is one of the major requirements to systematically run the simulations and predict model behaviour. Furthermore, the process should be automatically supported and it should be able to leverage all available data sources. The available memory and computation power, which until recently were considerable constraints to an automated process, are no longer an issue with the current availability of cloud infrastructures and data centres. Leveraging this new technology allows the modeller to use all resources that might necessarily support her in creating accurate simulation output.

3 STATE OF THE ART

This section discusses System Dynamics with respect to PEEF, which is, after roughly 40 years, still mostly manually done by the modeller. Relying on modeller experience, assumptions and the knowledge of domain experts, if available, the initial parameters and equations of resulting models are usually not providing satisfactory results after simulating the model. In most cases, eventually a series of try (adjust parameters/equations) and fail (rerun the simulation) replays will deliver acceptable results in the end (see (Forrester, 1991; Graham, 1981; Richardson, 1992)), but this trial and error process is expensive and inefficient. Nonetheless a lot of excellent research has been done in these fields (Senge, 1974; Peterson, 1976; Burns, 1977; Graham, 1980; Chen and Jeng, 2002; Medinaborja and Pasupathy, 2007). To simplify further explanation of these concepts, we will borrow the definition of a very simple system from Peterson.

$$X(t) = A * X(t-1) + W(t) \tag{1}$$

$$Z(t) = X(t) + V(t) \tag{2}$$

Let X be the state of the system, A an unknown parameter which has to be estimated, Z the actual measured state of the system, W the equation error (driving noise) and V the measurement error. Additionally $\hat{Z}$ is defined as the simulated state of the system. In the now following subsections, we are going to summarise the approaches of the former named researchers.

3.1 Estimation through Simulation

The concept of imitating a real-world process over time, so called simulation, is widely used in a variety of technical fields, such as aeroplane design, building constructions, weather forecast etc. It is also one of the common methods for parameter estimation in System Dynamics. Senge and Peterson showed the estimation of a parameter A by consecutively rerunning the system simulation with new assumptions of A until the simulation produces satisfying results. Peterson called this method the Naive Simulation (NS).

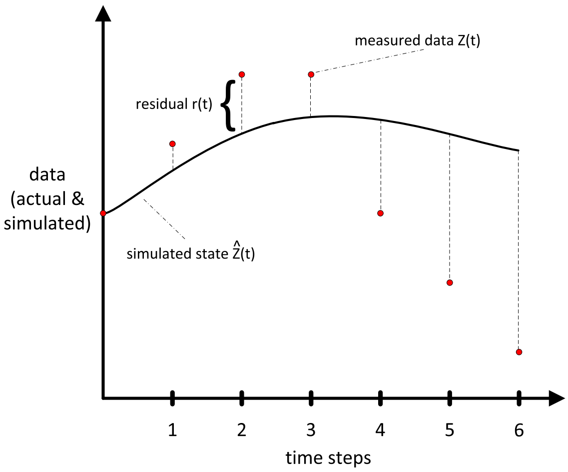


Figure 3: The naive simulation.

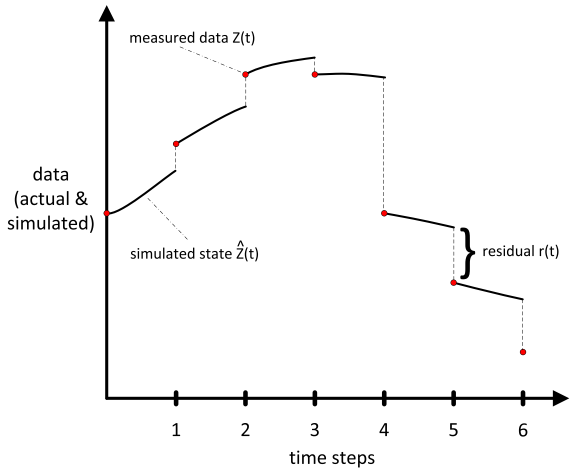


Figure 4: The ordinary square simulation.

It is mostly accomplished by determining an initial value for the parameter A (e.g. by guessing), running the simulation to get new results $\hat{Z}$, calculating the difference between $\hat{Z}$ and Z (so called residuals r) and finally repeating this methodology until a value for A has been found which causes the simulation to produce acceptable results. To measure the success of the parameter estimation, they were initially using the Least squares method invented by Gauss (Aldrich, 1998). This statistical concept is mostly used in over-determined systems with more equations than variables to estimate. The idea is to minimise the sum of all squared residuals to get the best fit for the estimated parameter. Figure 3 shows an example simulation and the residuals to be minimised with the least squares approach. The NS approach works very well in perfect systems not being influenced by external circumstances (reflected in the equations as driving noise W). In fact Peterson has shown that for systems containing driving noise W the NS approach might deliver completely wrong parameter estimations, because most of the available data is simply ignored and the system might completely drift away from the simulation result (Peterson, 1975). In such cases where driving noise is present, but measurement errors are still absent, an advanced NS approach might deliver better results. Whenever a new data point is available, Peterson referred to the *Ordinary Least Squares* (OLS) method to reset the current system state when simulating to counteract the drift (Peterson, 1976). The method delivers satisfying results for modelled systems with driving noise W, but it is easy to understand that measurement errors V of each available data point will also end in unsatisfying parameter estimations due to the wrong state of the system when resetting the system. One can argue that nowadays the quality of stored data in warehouses and data bases is considerably more accurate than back in the days, but at the time Peterson formulated these ideas, stored data was rare and mostly not checked automatically for quality or the observed data was even retrieved manually. An excellent idea of preventing the problem of high influence from measurement errors to the estimation of parameter A is Petersons idea of the Full-Information Maximum Likelihood (FIMLOF) algorithm. It is based on the Kalman filtering technique (Kalman, 1960). FIMLOF was designed to determine the most likely state of the system at each time t where data is available, by considering all given measured, simulated and expected error data. Whereas the measured and simulated data will be the same input as by NS and OLS, the expected error data is additionally computed by using the standard deviation for the predicted state $\hat{Z}$ and the variance for the measured data Z. Given the two most likely cases that either V is high and W is low (high measurement errors, but low driving noise) or V is small and W is high (the prediction is incorrect, but the observed data has high quality) the algorithm will behave as follows: In the first case, FIMLOF will choose a value close to the predicted output for the current time step, whereas in the second case, the algorithm tends to choose a value close to the measured data point. Either way, FIMLOF has a very high chance to choose the most likely value of the current system for the next simulation step. This specific characteristic increases the accuracy of the parameter estimation, because the more accurate the predicted system state can be retrieved, the fewer errors are passed through the parameter estimation. Nevertheless, each of these approaches forces the modeller to rerun the whole simulation several times until finally computing a satisfying estimation of parameter A. The simulation approach is therefore no end-to-end process and works on hard assumptions

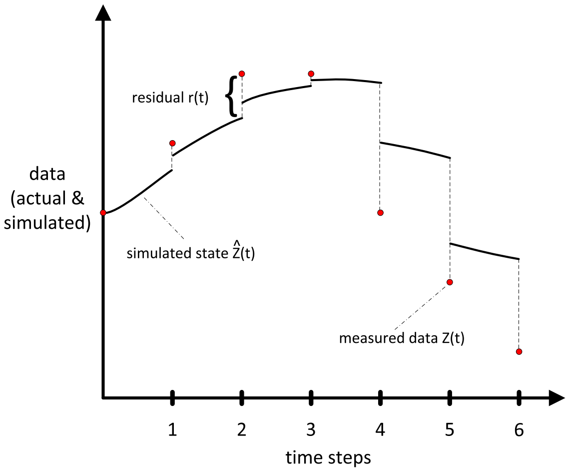


Figure 5: The FIMLOF simulation.

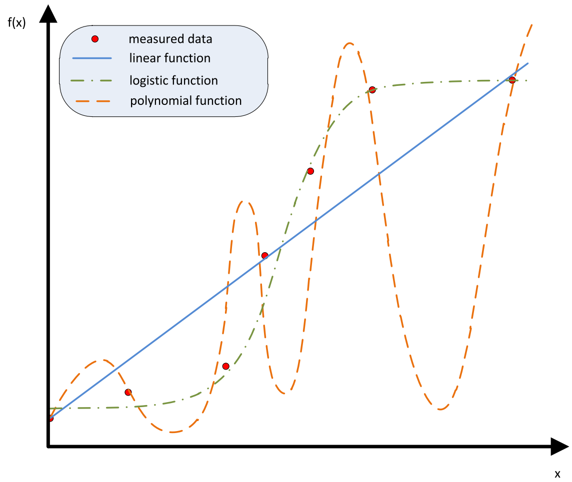


Figure 6: Equation calculation with statistical analyses like regression.

for the inital values of *A*, which is summarised in the limitations *L1* and *L5*.

3.2 Estimation by Data Type

Graham divided the model quantities into representations of data below the level of aggregation and data at the level of aggregation (Graham, 1980). Data below the level of aggregation (also called disaggregated data) refers to observations and measurements made in the real world which can be directly addressed and therefore conforms to a specific observable characteristic. Examples are the number of sold items in a market at a specific time or the amount of vacation days of one specific employee in a year. On the other hand, data at the level of aggregation describes quantities which are accumulated out of a number of different basic values and are not atomic, e.g. the time for a TCP/IP packet sent from one client machine to another client machine. The main problem with data at the level of aggregation is that it hides the main root causes which are driving and influencing the aggregated data. Graham shows different approaches of parameter estimation for data below the level of aggregation and data at the level of aggregation (Graham, 1980).

The actual approach for estimating parameters from disaggregated data depends on the available observed data. If few data points are observed, the modeller is able to determine a parameter by choosing a value between the given observed limits. Dependent on the size of the limit interval, the modellers guess of parameter *A* might be more or less accurate. For more available data points, Graham proposed to use a table function with specific interpolations to determine the parameter *A*. The trick in this case is to identify the right interpolation to get slopes with smooth curves between the normal observed values and the extreme observed values. And last, if the modeller has access to numerical estimates or process observations which represent the modelled quantity, this data might be used to calculate the actual value of the modelled quantity. Graham uses the example of the rabbit birth rate in an ecological model to explain this methodology. Even though the modeller might not have an observed value for the rabbit birth rate itself, she is at least in the position to acquire observed behaviour about rabbit reproduction to calculate the rabbit birth rate. In every case, parameter estimation for disaggregated data never uses actual model equations to retrieve parameters but instead relies on statistical approaches like regression (linear or polynomial). Figure 6 shows that by using statistical approaches an optimal fit for a given data set might be computed, but without background knowledge about the analysed data set the resulting function might be completely misleading for future data points. This problem is covered with limitation *L4* and fully applies to the concept of estimation via disaggregated data.

Graham additionally explains the concepts of Equation estimation and Model estimation to estimate parameters from aggregated data. Both methods rely on model equations and transposition (usage of algebra) to estimate the parameters. Using equation estimation, the modeller manipulates exactly one model equation which consists of aggregated quantities like stocks, flows and variables to compute a value for one parameter *A*. The methodology of model estimation involves transposition of all model equations to calculate parameters. In both cases, the modeller would start to transpose the chosen model equation(s) up to the point where *A* can be simply computed by inserting the aggregated data values and resolving the equation(s). But working with aggregated data and model equations usually involves assumptions made by the modeller (see limitation *L1*), which in return gives room for possible errors (Graham, 1980). However, the quality of the parameter estimation for these methods is obviously highly dependent on the accuracy of the underlying model equation(s) transposed and used for calculation. Additional data of further involved variables or rates, accessible by the modeller, is not at all incorporated in the parameter calculation. This approach is therefore only working on a possible fraction of the available and accessible background data. The two limitations *L2* and *L3* are described in the limitation section.

3.3 Equations by Dimensions

Burns explained the approach of transforming a previously created causal diagram D (as in figure 1) into a state/flow model by using a square ternary matrix (STM) and modified square ternary matrix (MSTM) as intermediate steps (Burns, 1977). The STM contains all quantities q_i as rows and q_j as columns of a given causal diagram and defines either -1, 0 or 1 for a connection from q_i to q_j or no connection (usually empty cells), as shown in table 1 for the example of the returning customers. The sign in front of a 1 indicates the influence of q_i to q_j, which is either negative or positive. After identifying a set of definitions (D1 - D9) and a set of axioms (A1-A7) which reflect the structure of a SFD according to Forrester, Burns was able to create systematic algorithmic rules. These rules, when applied to a STM, deliver an SFD. The SFD might also be represented visually with a modified STM (so called MSTM), as shown in table 3. The MSTM differs from the STM in the representation of the connections. Instead of having -1 and 1 as negative or positive connection, the MSTM contains either $-F$, F or $-I$, I to indicate whether the represented connection is an out- or inflow or a negative/positive information coupling. Having the MSTM and the dimensions (dim) of each quantity enables the modeller to retrieve equations for stocks, rates and variables as follows. Stock equations are apparently trivial to identify, because all stock equations are of the form: calculate the difference between the inflow r_{in} and the outflow r_{out} for the current time step Δt and add this value to the last value of stock x_i, which translates to the general equation:

$$x_i(t+\Delta t) = x_i(t) + \Delta t(r_{in} - r_{out}) \tag{3}$$

The specific equations for each stock of a model are therefore easily retrievable from the MSTM by identifying the inflow and outflow of each stock. The System Dynamics expert is furthermore able to determine rate and variable equations by investigating the MSTM and the dimensions (units) of these quantities. Having a closer look to the MSTM columns reveals the affecting quantities $A_q(q_i)$ for each variable or rate quantity q_i. As a matter of fact, q_i has to be at least calculated from its affected quantities, otherwise the given causal diagram must have been incorrect. Burns defined this relation with equation 5 (see (Burns, 1977) pp. 705 for further information). His mathematical function f is a mapping from all affecting quantities $q_j \in A_q(q_i)$ to the quantity q_i (see equation 4).

$$f : Q^n \rightarrow Q \tag{4}$$

$$\begin{aligned} q_i &= f[\{A_q(q_i)\}] \\ q_i &= f[\{q_{j1}, q_{j2}, \ldots, q_{jn}\}] \\ q_i &= q_{j1} \otimes \cdots \otimes q_{jn} \end{aligned} \tag{5}$$

The goal of f is to establish dimension consistency between all affecting quantities q_j and the target quantity q_i. This is achieved by applying the mathematical operators ($+$, $-$, $*$, $/$), abbreviated by the $\otimes$ operator, to all affection quantities q_j as shown in equation 5. Because of the given mathematical operators, the equation defined in f is always of a linear form. This method apparently fails at the point when some affecting quantities of q_i are dimensionless or the dimensions are not fitting together. In this case, Burns proposed to assume a table function for the affecting quantities A_q.

Apart from the limited linear form of the extracted equations (see limitation *L6*), the expressions in each equation are also not decorated with weighting factors and therefore might lead to inaccurate simulation results. For example in the business world we can easily build cases having one or more variables with weighted dependencies: The price p and the quality q of a product are both influencing the amount of sold product units u_{def}, and can therefore be connected to each other with the $\otimes$ operator as shown in equation 6. Since their dimensions are not fitting, a table function T, which maps the result to product units, has to be applied.

$$u_{def} = T(p \otimes q) \tag{6}$$

But dependent on the product, we can fairly assume that either the price or the quality of the product have more influence on the amount of sold units and should be weighted with weights ω_1 and ω_2. The equation 7 shows the weighted connection of the price and the quality,

$$u_{wei} = T((\omega_1 * p) \otimes (\omega_2 * q)) \tag{7}$$

where $\omega_1, \omega_2 \in [0, 1.0]$ and $\omega_1 + \omega_2 = 1.0$.

Table 1: Square ternary matrix for the example of the returning customers causal diagram.

	dim	0	1	2	3	4	5	6	7	8	9	10
0	CO			1								
1	CO					1						
2	$\frac{CO}{TU}$	-1	1									
3	$\frac{1}{TU}$			-1								
4	$\frac{MU}{TU}$			-1								
5	-								1			
6	$\frac{O}{TU}$								1			
7	-			1								
8	$\frac{RP}{TU}$								1			
9	$\frac{CC}{TU}$								1			
10	$\frac{MU}{CO}$			1								

Table 2: Description of used dimensions.

abbreviation	name	description
CO	customers	the amount of customers in the system
TU	time unit	a unit of time relative to the overall system time
MU	monetary unit	standard currency unit in the system
O	orders	all processed orders
RP	returned products	all returned products in the systems
CC	customer complaints	all customer complaints for orders

3.4 Equations with Surrogate Modelling

The sheer complexity of the System Dynamics domain including modelling, parameter estimation, equation formulation, confidence checking, etc., can be addressed by borrowing ideas and techniques from other well established domains.

Surrogate Modelling is one such potential interdisciplinary field, which can be employed in the System Dynamics domain to address the complex equation formulation part. By blending the concepts from the domains of Machine Learning and Statistics, Surrogate Modelling offers a technique to create a surrogate function $\hat{g}(x)$ for an unknown real function $g(x)$ by applying an analyses algorithm to a given training dataset. Dependent on the chosen analysis algorithm different equations can be formulated, e.g. low-order polynomials with the least-squares regression algorithm, neural networks with a back-propagation training algorithm or classifications with support vector machines. Since $\hat{g}(x)$ is only a substitute of the real function $g(x)$, it does not necessarily produce the same outputs for the same given inputs. A calculated surrogate function $\hat{g}(x)$ might therefore be either more accurate (computational intensive) or more computational efficient (less accurate) depending on the given constraints (time, computation power, etc.). Forrester et al. and Vapnik have provided an excellent overview of Surrogate Modelling and available analyses algorithms (Forrester et al., 2008; Vapnik, 1998). However, research effort in this direction was initiated by Chen & Jeng (Chen and Jeng, 2002) based on the work of Dolado (Dolado, 1992). Chen and Jeng discussed the usage of artificial neural networks (ANN) for System Dynamics as another representation of an SFD in the first place. ANNs were first pioneered by McCulloch & Pitts in the early 1940s and further improved by Rosenblatts perceptron theory, Hopfields energy approach and Werbos back-propagation learning algorithm (McCulloch and Pitts, 1943; Rosenblatt, 1962; Hopfield, 1982; Werbos, 1974). Chen and Jeng used one partial recurrent neural network (PRN) to represent a complete system dynamics model and introduced a transformation from SFD to PRN. To enable such a transformation, there has to be a mapping of System Dynamics elements (quantities and connections) to neural network elements as follows. A stock variable is transformed into an input, state and output neuron. The input and output unit handle the

Table 3: Modified square ternary matrix for the example of returning customers.

	dim	0	1	2	3	4	5	6	7	8	9	10
0	CO			I								
1	CO					I						
2	$\frac{CO}{TU}$	-F	F									
3	$\frac{1}{TU}$			-I								
4	$\frac{MU}{TU}$			-I								
5	-								I			
6	$\frac{O}{TU}$								I			
7	-			I								
8	$\frac{RP}{TU}$								I			
9	$\frac{CC}{TU}$								I			
10	$\frac{MU}{CO}$			I								

input and output function of a stock, whereas the state unit serves as storage. Flows and their rates are represented by a hidden unit and the connection between a hidden unit and an output unit, which is part of a stock representation. Auxiliary variables are not mapped as such, because Chen and Jeng argue that these variables might be expressed as subdivided parts of a rate equation ("a rate in front of another rate" (Chen and Jeng, 2002)). Furthermore, parameters (constants) are either imitated with stocks without having a connection to hidden neurons to prevent changes in the simulation or parameters are treated as multipliers in rate equations and therefore are not specially represented with a neuronal network element. Finally information couplings are illustrated with links between hidden and state neurons. Given these transformation rules Chen and Jeng present a transformation algorithm (FD2PRN) to convert a given SFD into a PRN. They are furthermore applying standard algebra to the activation functions of the PRN to proof the mathematical compliance of the transformed PRN and the typical stock, rate, initialisation and constant equations.

Up to this point the ANN is only used to illustrate any SFD and is therefore just another representation of a SFD like Burns MSTM. But as mentioned earlier, ANNs have the ability to unveil hidden patterns in a given dataset and therefore are capable of providing predictions for the future development of the dataset. The neural network mimics the equation which produces the values of the given dataset. Having this equation enables a modeller to predict future values. In other words, if there is input data available for a given ANN, the ANN can be trained and afterwards used to predict results. This statement also holds for Chens & Jeng's created PRN and they leverage this concept by training the raw untrained PRNs of their System Dynamics test models with previously simulated data. The trained PRNs might then be used to predict the system behaviour, similar to simulation runs of SFDs. The results for training of the PRN in their paper are quite promising and given the learning ability of ANNs, they are highly adjustable to external changes in the system under study. These insights motivate for deeper research in this field and we, the authors, believe that the concepts of Surrogate Modelling and Machine Learning in general are very well suited to tackle the problem of automated PEEF in System Dynamics. We are especially highlighting this, because these concepts are embodying the least of our addressed limitations. Nevertheless, there are open questions arising from Chen & Jengs work. For instance, the prediction accuracy for known worse neural network equations like alternating behaviour might not be appropriately represented by a neural network.

3.5 Formulation via Decision Trees

For decades the System Dynamics community relied on Forresters recommendations of the three different models explained in the beginning of this paper on how to retrieve knowledge for building System Dynamics models. Forrester values the mental model far above the written and numerical model, because there was simply not enough data to replace the human mind of the modeller and domain experts. This guideline is still valid, but in the modern business world where every digital step of each customer is monitored and stored in huge databases, the written and numerical models are becoming more and more useful and relevant. Research communities in the area of business intelligence and business process management are exploiting this huge amount of available data and proposing enhanced solutions in the area of business decision support. For instance, Medina-borja & Pasupathy are leveraging this data for semi-automated model creation and equation formulation (Medina-borja and Pasupathy, 2007). They are showing two statistical approaches to identify predictors of model variables from a given data set and afterwards one algorithm to leverage these dependencies and reveal their mathematical representations. Classification and Regression Trees (CART) and Chi-Square Automatic Interaction Detection (CHAID) are both decision tree methods which are used to divide the given data set into groups and subgroups to assign them to nodes. After the tree has been grown and possibly pruned, most of the remaining nodes in the tree represent important independent variables. Common usages in the literature for CART and CHAID are the identification of predictors for customer behaviour and market segmentation, direct marketing to group customers in classes or the field of processing mining to classify process instances. On the other hand Structural Equation Modelling (SEM) is a statistical approach of validating or exploring a predefined model with a given data set, see for instance (Hayduk, 1985) or (Pearl, 2000). One idea to use SEM is to first create a model which supposedly fits the given data set and afterwards applying the SEM algorithm to the defined model and given data set to figure whether the model fits the data and if so, how much. The model consists of measured variables (indicators) and unobserved/abstract variables (latent variables). The outcome of SEM is the cause and effect sizes (structural coefficients) which might be used for equation formulation. The idea proposed by Medina-borja & Pa-

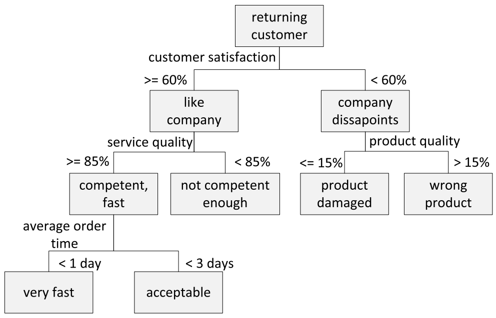


Figure 7: A regression tree for the problem of returning customers.

supathy is to use CART or CHAID to uncover the dependencies of a given data set and create a model using the generated decision tree. Afterwards SEM can be used to determine the fit of the model to the data and to provide the structural equations of the model. The resulting model and its equations can be used as a SDM and eventually fed into simulation/analyses tools. Unfortunately SEM is only capable of creating linear structural equations and is thus subject to limitation *L6*. However this concept shows a semi-automated procedure from a given data set to final simulation results.

4 GENERAL LIMITATIONS

All of the above stated approaches are extremely helpful for a System Dynamics expert to either retrieve parameter values or gain help in formulating equations in a System Dynamics model. As each of these concepts require specific prerequisites, there are certain minor or major limitations associated with these algorithms and additional questions arise which need further research to be answered. We have identified and collected a number of these limitations (*L1* - *L6*) which are either stated by the authors of the algorithms themselves or are obvious when applying the algorithms.

L1. *Assumptions.* We have observed that some of the algorithms are working with hard assumptions, for instance to guess initial values. Assumptions generally lead to errors because there is always room to speculate. This limitation also implies a decrease in the quality of the retrieved parameters/equations.

The algorithm works on assumptions.

L2. *Predefined Equations.* The availability of System Dynamics model equations is a strong prerequisite for simulations. For instance, in the case of the estimating by data type approach, model equations have to be manually provided to start transposing them and finally resolving parameters. This possesses a significant limitation for the applicability of the algorithm, because the equation formulation requires a huge amount of effort and domain expertise. Given the fact that the modeller is particularly interested in the simulation result output, she is forced to additionally perform the complex equation retrieval process by hand.

Model equation information needed by the algorithm restricts its usage and forces the modeller to deal with additional intermediate steps.

L3. *Limited Data Utilisation.* Many of the algorithms have a very restricted view on the available data sets; they only consume a fraction of the available data. Good examples are observed in the equations by dimensions algorithm where only the dimensions of all quantities are incorporated and in the estimation through simulation algorithm where only the historical measured data sets are captured. Historical measured data, for instance in the equations by dimensions approach could be readily used to further refine the retrieved equations with weights. The limited data view drives towards inaccurate equation formulations and thereby misleading simulation results.

Limited data utilisation leads to inaccurate equation formulations.

L4. *Interpolation.* Many algorithms (especially statistical algorithms) are very much capable of providing optimal equations that fit a given data set (see polynomial regression algorithm figure 6). However, these algorithms do not incorporate the actual semantics hidden in the data while interpolating a given data set. The resulting equations are therefore lacking the accuracy to compute future data points outside the given data set range.

The algorithm does not capture hidden patterns and semantics.

L5. *Automation.* None of the algorithms support an automated end-to-end process for PEEF. When using these algorithms, there are always intermediate manual steps involved. For example, determining the interpolation approach, aggregating data, providing basic equations for further refinement, creating a model from a given decision tree, training the algorithm. Manual execution of an algorithm or intervention while the algorithm is executed is not only tedious and requires a lot of domain knowledge, but also slows down the actual process and raises additional possibilities for failures.

Table 4: Overview of concepts for PEEF and their limitations.

Method	Algorithm	Type	L1	L2	L3	L4	L5	L6
Simulation	NS	PE	✓	✓	○	○	✓	n.a.
	OLS	PE	✓	✓	○	○	✓	n.a.
	FIMLOF	PE	○	✓	○	-	✓	n.a.
Data type (disaggregated)	Estimate between limits	PE	✓	-	✓	✓	✓	n.a.
	Estimate table functions	PE	✓	✓	○	✓	✓	n.a.
	Calculate numerical data	PE	○	✓	-	✓	○	n.a.
Data type (aggregated)	Equation estimation	PE	✓	✓	✓	✓	✓	n.a.
	Model estimation	PE	○	✓	○	✓	✓	n.a.
Dimension	STM/MSTM	EF	-	-	✓	✓	✓	✓
Surrogate Modelling	FD2PRN	EF	-	-	-	-	○	-
Decision trees	CART, CHAID, SEM	EF	○	-	-	○	○	✓

The algorithm is not designed to operate in an end-to-end fashion without manual intervention.

L6. *Non-linearity.* Linear equations can describe the system properly, but as stated in ***L4*** a wrongly selected interpolation leads to inaccurate simulation results. Additionally, in the business world were we have to deal with highly non-linear behaviour, algorithms are needed, that are capable of computing non-linear equations. A good example is that of the returning customers (presented in figure 2). Since some of its influencing factors, such as the average process order time, can't be written as independent linear combinations, the returning customers problem is a non-linear system. The reason is that there are so many influencing factors like marketing effort, the number of one-time customers or even the average process order time indirectly linked via customer satisfaction to the returning rate. These variables and the flow can more precisely be captured with complex non-linear cubic, logarithmic, exponential, etc. equations. We observed that, not all analysed algorithms, which are intended for equation formulation, are capable of producing the non-linear equations that can optimally capture the system behaviour.

The algorithm is not capable of extracting non-linear equations.

All identified limitations are summarised in table 4. For all analysed algorithms the following three symbols are used to indicate how much the limitation applies to the current algorithm.

1. The hyphen symbol (-) implies that this limitation does not apply at all.
2. A circle symbol (○) suggests that this limitation is partly valid.
3. The check mark symbol (✓) shows that this limitation completely holds.

The table contains a method column which describes the methodology used by the algorithm to compute its results, an algorithm name, a type column which either contains the abbreviation PE (parameter estimation) or EF (equation formulation) to show the main usage of the algorithm, and one column for each defined limitation, respectively.

5 CONCLUSIONS

In this paper the authors have analysed methodologies and techniques for PEEF in the domain of System Dynamics. These methodologies have facilitated the work of System Dynamics modellers to run and simulate the models and finally get output for future system behaviour. Researchers like Burns, Graham and Senge have developed concepts to estimate parameters and to formulate equations for System Dynamics models from the early 70s/80s. Nevertheless, each of the studied approaches is embodying specific limitations which are posed by the very nature of the concept itself or these approaches were not originally meant to be used for PEEF in the first place. Especially for the equation formulation none of the algorithms offers an end-to-end process from model and data to annotated, ready to simulate model. The authors believe that this concept of an end-to-end automatic PEEF process is worthwhile to be researched, because it would significantly decrease the manual workload of the modeller to retrieve parameters/equations. Chen & Jeng and Medina-borja & Pasupathy have shown, that machine learning and classification approaches are very much suitable to first create the formal models and afterwards annotate them with parameters and equations. Since nowadays more and more business data is generated and stored, we see high potential especially in the surrogate modelling concepts to leverage this data for Business Dynamics. Our future goal is to create a semi-automated framework which is capable of transforming business data into ready-to simulate SFDs. For this, we will have to incorporate an automated version of PEEF within our planned framework. This will free up the analyst from doing unnecessary tasks of manual PEEF, allowing her to focus more on her actual modelling tasks. We further plan to reuse and embed the existing machine learning and classification approaches in our framework. We will invest further research to help automate most of the crucial steps of System Dynamics.

REFERENCES

Aldrich, J. (1998). Doing Least Squares: Perspectives from Gauss and Yule. *International Statistical Review*, 66(1):61–81.

Binder, T., Vox, A., Belyazid, S., Haraldsson, H. V., and Svensson, M. (2004). Developing System Dynamics models from Causal Loop Diagrams. Technical report, University of Luebeck, Germany; Lund University, Sweden.

Burns, J. R. (1977). Converting signed digraphs to Forrester schematics and converting Forrester schematics to Differential equations. *IEEE Transactions on systems, man, and cybernetics*, 10:695–707.

Chen, Y.-T. and Jeng, B. (2002). Yet another Representation for System Dynamics Models , and Its Advantages. In *System Dynamics Society Conference*.

Dolado, J. J. (1992). Qualitative Simulation and System Dynamics. *System Dynamics Review*, 8(1):55–81.

Ford, A. (1999). *Modeling the environment: An Introduction to System Dynamics Models of Environmental Systems*. Island Press, Washington, D.C.

Forrester, A., Sobester, A., and Keane, A. (2008). *Engineering Design Via Surrogate Modelling: A Practical Guide*. Wiley-Blackwell.

Forrester, J. W. (1961). *Industrial Dynamics*. MIT Press; currently available from Pegasus Communications; Waltham, MA, Cambridge, MA.

Forrester, J. W. (1971). *Urban Dynamics*. Pegasus Communications, Waltham, MA.

Forrester, J. W. (1991). System Dynamics and the Lessons of 35 Years. pages 1–35.

Forrester, J. W., Mass, N. J., and Ryan, C. J. (1976). *The system dynamics national model: Understanding socio-economic behavior and policy alternatives*.

Graham, A. K. (1980). Parameter formulation and estimation in System Dynamics Models. *Elements of the System Dynamics Method*, pages 143–161.

Graham, A. K. (1981). Lessons on modeling from the System Dynamics National Model. In *1981 System Dynamics Research Conference*.

Haraldsson, H. V. (2000). Introduction to Systems and Causal Loop Diagrams. Technical Report January, Department of Chemical Engineering, Lund University, Lund.

Hayduk, L. A. (1985). The Conceptual and Measurement Implications of Structural Equation Models. *Canadian Journal of Behavioural Science*, 17.

Hopfield, J. J. (1982). Neural Networks and Physical Systems with Emergent Collective Computational Abilities. In *In Proc. National Academy of Science*, pages 2554–2558.

Hosler, D., Sabloff, J. A., and Runge, D. (1977). Simulation model development: a case study of the Classic Maya collapse. *Academic Press: London*.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45.

Lane, D. C. (2000). Should System Dynamics be Described as a ‘ Hard ’ or ‘ Deterministic ’ Systems Approach? *Systems Research and Behavioral Science*, 22(June 1999):3–22.

McCulloch, W. S. and Pitts, W. (1943). A Logical Calculus of Ideas Immanent in Nervous Activity. *Bull. Mathematical Biophysics*, 5:115–133.

Meadows, D. H., Meadows, D. L., Randers, J., and Behrens, W. (1972). *The Limits to Growth*. Universe Books, New York, NY.

Medina-borja, A. and Pasupathy, K. S. (2007). Uncovering Complex Relationships in System Dynamics Modeling: Exploring the Use of CART , CHAID and SEM. In *System Dynamics Society Conference*, pages 1–24.

Pearl, J. (2000). *Causality: Models, Reasoning and Inference*. Cambridge University Press.

Peterson, D. W. (1975). *Hypothesis, Estimation and Validation of Dynamic Social Models*. PhD thesis, MIT, Cambridge, Mass.

Peterson, D. W. (1976). Statistical Tools for System Dynamics. *Elements of the System Dynamics Method*, pages 841–871.

Richardson, G. P. (1992). Problems for the Future of System Dynamics. Technical report, Rockefeller College of Public Affairs and Policy, University Albany.

Rosenblatt, R. (1962). *Principles of Neurodynamics*. Spartan Books, New York.

Senge, P. M. (1974). An experimental evaluation of generalized least squares estimation. In *System Dynamics Group Memo*.

Sterman, J. D. (2000). *Business Dynamics: Systems thinking and modeling for a complex world*. McGraw-Hill, New York, NY.

Vapnik, V. N. (1998). *The Nature of Statistical Learning Theory*. Springer, 2nd edition.

Werbos, P. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University.

Analysis on the Value of Process Support Implementations for Quality Management

Michael Seitz[1] and Stefan Jablonski[2]
[1]*PRODATO Integration Technology GmbH, Erlangen, Germany*
[2]*University of Bayreuth, Chair of Applied Computer Science IV, Bayreuth, Germany*
michael.seitz@prodato.de, stefan.jablonski@uni-bayreuth.de

Keywords: Quality Management, Process Support, Value Creation, Maturity Models, Adaptation.

Abstract: Many organizations face competitive pressure to enhance their business process capabilities and to comply with quality management directives. Methods like the Business Process Maturity Model (BPMM) therefore provide assistance by stating concrete requirements and measuring their fulfillment. However, there is still uncertainty about how much technical support (e.g. guidance, documentation) is actually reasonable in order to adapt the business process to a specific maturity level. In this paper an analysis approach is introduced that enables to assess the value of process support implementations for quality management and assists practitioners with the decision-making whether the value is appropriate for the use case. The application of the concept is illustrated using the example of four representative implementations ranging from manual human-controlled to automatic system-controlled support.

1 INTRODUCTION

In order to remain competitive, many organizations continually enhance and optimize their business process maturity. Maturity models like the Business Process Maturity Model (BPMM) (OMG, 2008) therefore serve as orientation as they "have been designed to assess the maturity (i.e. competency, capability) of a selected domain" (Bruin et al., 2005) and derive points for improvements. Maturity levels (ML) describe concrete requirements, e.g., the orientation towards a reference process, but do not provide indication of supporting methods or tools to enact and execute a reference process model. The issue of adequate technical support for process enactment that is managed on basis of the quality requirements of maturity models is rather complex: Reasonable support has to be adapted to current as well as constantly changing requirements of the deployed quality management (QM) standard, i.e. conditions for qualitatively appropriate execution of business processes (for example deliver in time with consistent performance). Moreover, "the proper fit between the tasks in the business processes and information technology/systems must exist" (Trkman, 2010). Since many companies have adopted a value-based approach to manage the deployment of information technology (IT) and decisions on IT are made on basis of the contribution to strategic business objectives (Mauch and Wildemann, 2007), the question arises how the value of process support implementations for QM can be determined and serve as decision guidance. It is obvious: the more tool support is chosen, the more expensive process execution becomes. However, it has to be asked whether this additional tool support has an adequate value with respect to QM. In these matters, decision makers need to rely on more than just "gut feeling".

There already exist some in-depth investigations on the ability of IT for support of specific processes and use cases. Zur Mühlen and Rosemann (2000) outline the economic aspects of workflow-based process monitoring and controlling. Faerber (2010) conceptually designs a process navigation system as implementation for process-oriented QM. Becker et al. (1999) present a structured framework which enables the evaluation of the potential of business processes to be supported by workflow management systems. The listed related work in each case is concentrated on a specific implementation approach. There is still a lack of a general approach considering the whole spectrum of process support including both human- and system-controlled approaches. The key issue about the reasonable degree of process support, i.e. which tasks should be accomplished fully/semi automated or rather manually, is not yet thoroughly assessed. It is still unclear how the actual value of process support for QM can be measured, independent of the degree of IT assignment and specific implementation approaches.

In this paper the value of process support for QM is investigated. The term "value" should be perceived as the contribution that is made to being compliant to requirements of QM standards. In practical terms, it is evaluated to what extent process management tasks – such as information, coordination or documentation – no longer have to be carried out manually but are handled in whole or part by process support tools. The value is measured by the appropriate fulfillment of maturity level requirements on process support.

Therefore, an adequate analysis approach is introduced and illustrated by the example of four different representative implementations for process support. The objective of the analysis is to get an appreciation of to what extent the implementations penetrate the process – in particular with regard to the degree of IT assignment and the remaining degree of freedom for the users – and thereby support the users in performing their tasks in accordance with requirements of QM standards. The analysis results are intended to provide practitioners a better basis for decision-making about a broad spectrum of suitable implementations.

In our previous work (Seitz and Jablonski, 2012; Seitz and Jablonski, 2013) we approach the problem of adequate process support from a business point of view without directly referring to QM. In this paper, the IT side is analyzed, especially as regards the selection of specific implementations that comply best with the demanded capabilities. Therefore, in addition to Jablonski (2010) an even more specific classification of process support is performed with respect to each process perspective (e.g. data, organization). The concept is tightly focused on QM so that conclusions can be drawn – on the one hand for the value, i.e. the promoted quality at runtime (guidance during execution) and after runtime (evidence through documentation) and on the other hand for the caused costs (e.g. modeling efforts). In this manner, we want to give a more precise answer to the question of adequate process support.

2 APPROACH

The concept is divided up into three steps. Firstly, the classification framework for the degree of process support according to Seitz and Jablonski (2012, p. 95) is introduced and applied in the context of QM support functions. Secondly, the evaluation instrument for setting the benchmark is presented. Maturity levels thereby serve as measure how valuable process support is. Thirdly, a procedure is suggested that aims for reaching a decision on the question which is the most valuable approach for process support with respect to a specific maturity, i.e. requirements of QM standards.

2.1 Classification

Process support for QM comprises four basic functions (Faerber, 2010, p. 75): information provision, data integration, coordination, and documentation. Depending on the required ML these basic functions can be implemented quite differently. For example, if the goal is to reach a high maturity, it is recommended to coordinate work packages and project staff accurately. Relevant control information should be integrated electronically to be able to collect and analyze key performance indicators systematically. However, a low maturity just demands to achieve the results (anyhow) and therefore allows for the coordination function to be performed rather rudimentarily or not at all. It may be also sufficient to retrieve process instructions or measurement data by hand.

Seitz and Jablonski (2012) introduce an adequate framework for the classification of the degree of process support that is based on the perspective-oriented process model (POPM) (Jablonski, 1994). The five main perspectives of POPM according to Jablonski and Goetz (2008) are: functions (process steps and their purpose), data (used data, e.g. documents, and data flow between the process steps), operations (invoked services and tools), organization (people or machines and their responsibilities) and behavior (control flow). The functional perspective thereby represents the composition ("skeletal structure") of the process on which the other perspectives are built on. This is way the functional perspectives can be excluded from the classification. The framework covers the whole spectrum for both internal and external enactment of process models (under vs. beyond the control of information systems) as well as the range between strict and flexible execution (little or no freedom vs. high degree of freedom and decision making by the users). In the following, the characteristics (perspectives) and the values of the framework are explained using the four basic functions for QM support (see Figure 1).

Information Provision: Users are provided with detailed information across all perspectives about the process steps to be performed, like some kind of handbook or guideline. The more detail of information is supplied the tighter the process execution is restricted. However, this support function is completely separated from the actual process execution; there we call it "external enactment" because it is limited to a passive role by indeed presenting the users all relevant facts of the process but not being able to influence the actual process execution or even the user's behavior. Regarding the operational perspective, directives on mandatory tools can be set. Concerning the organizational perspective, responsibilities are defined, either at the level of a group of persons acting with a common purpose – maybe a role or department ("non-agent") – or at the level of individuals ("agent"). The behavioral perspective is covered by a clear textual or visual description of the chronological sequence of the process steps and their dependencies.

Data Integration: A distinction is made between unstructured (e.g. an image) and structured (e.g. a form or a record) data. They are consolidated from different sources in order to make them centrally available in electronic form, e.g. for presentation purpose (data perspective). In doing so, it is also possible to establish application interfaces and to make 3rd party tools more accessible (operational perspective). Those can either be suggested to the users for manual execution – possibly through a launch pad (assisted enactment) – or automatically be invoked and parameterized (automatic execution). The latter option provides less flexibility and is often preceded by a costly development and deployment.

Coordination: Project staff and work packages have to be reconciled and harmonized taking into account restrictions and due dates. Hence, the organizational (task assignment) and the behavioral perspective (temporal and logical sequence of process steps) is concerned. Process support for the coordination function is considered to be system-integrated, because adequate implementations must keep track of the actual course and are in need of feedback about the current process context (internal enactment). It can be performed either flexible or strict: Assigning a task to a role or department provides more flexibility than to a specific person or server process. Accordingly, there arise far more possibilities from a suggested set of suitable process steps than from exactly prescribing the execution order.

Documentation: The compliance with quality requirements has to be proved through documentation. This can be achieved either manually by the users or automatically through IT. The actual process execution is documented paper-based or electronically on different level of detail for each process perspective. For instance, in many cases it is sufficient to simply record that a certain process step has been accomplished (e.g. by presenting the process results), whereas often also additional information such as the executing agent or the applied tools have to be documented.

Process support through information provision (external enactment) is complemented or rather substituted on the one hand by data integration regarding data and operations and on the other hand by coordination regarding organization and behavior (internal enactment). While external enactment just enables to communicate the way the process should be executed, internal enactment also ensures that it will actually be done (within the granted degree of freedom). In the following it is assumed that both data integration and coordination include information provision with regard to the respective process perspectives. Furthermore, with internal enactment by data integration and coordination also the documentation of the related process perspectives is covered more or less automatically.

Aspect	Values: External (manual) – Flexible	External (manual) – Strict	Internal (system integrated) – Flexible	Internal (system integrated) – Strict
Data	Information Provision: Paper based		Data Integration: Electronic	
	Unstructured	Structured	Unstructured	Structured
Operations	Free choice	Directives on mandatory tools	Assisted enactment	Automatic execution
Organization	Not specified / Task to Non-agent	Task to agent	Coordination: Task to Non-agent	Task to agent
Behaviour	Not specified	Process description	Set of process steps	Order of process steps
Documentation				

Figure 1: Classification framework.

This simplified example for a single process step of the travel expense report shows how support functions and their interdependencies work:

Information Provision: Via the enterprise intranet it is communicated that the traveler (organization) must make the reimbursement application (data) within three days after returning from the travel (behavior) and send to the travel department by email (operation).

Data Integration: The document template can be downloaded by clicking on the provided link (data). A word processing application is recommended for editing (operation).

Coordination: As soon as the travel is approved (behavior) the traveler is given the task to perform the travel expense report (organization).

Documentation: By means of the email containing the reimbursement document the process step is – more or less automatically – documented. It proves who applied which reimbursement application and when (all perspectives).

Some higher QM standards demand to take corrective actions in case of exceptions and moreover to incorporate improvements in future instances of the process model. In this framework this aspect is not be dealt with explicitly. Even though the investigated support functions the concept is based on do not directly cover continuous improvement they include the collection of relevant data (data integration) and the delegation of further measures (coordination) for improvement and innovation to human agents or 3rd party tools that provide process support through assistance and planning according to Jablonski (1994).

The contribution of the classification framework to the research question is to determine to what extent the QM support functions are implemented. In order to establish a common scale for the evaluation of which value a certain implementation thereby actually creates, in the next section, the quality requirements of the MLs are mapped to the support functions.

2.2 Evaluation

Maturity models "are used as an evaluative and comparative basis for improvement" (Bruin et al. 2005) and therefore are suitable to establish a common scale for the evaluation of the value proposition. The Business Process Maturity Model (BPMM) is chosen for this evaluation because it is focused on all kinds of processes of an organization (Hogrebe and Nüttgens, 2009). In the following, based on Seitz and Jablonski (2012) and Schönig et al. (2012), the quality requirements of business process MLs on the previously introduced support functions are investigated. In turn, it can be determined which ML implementation approaches are able to reach for each support function, and to what extent additional support by other tools or by hand is required.

Initial: ML1 just demands to achieve the process results. It does not place any specific requirements on process support. In this respect all implementation approaches meet ML1 (by definition).

Managed: ML2 demands proper results in time. Therefore, the process has to be planned like a project. In order to set up and perform a project schedule, information about organization and behavior have to be provided and the execution has to be coordinated (either flexibly or strictly). Besides the achieved results it must be documented that the project plan was adhered to (organization and behavior).

Standardized: ML3 demands that the process execution follows a reference process. Therefore, all information about the reference process have to be specified across all perspectives. Document templates and input screens (if needed) should be made available centrally and applications or tools to be used should be suggested or prescribed (flexible or strict data integration). Similar to ML2, adequate coordination is necessary. All relevant process perspectives have to be documented properly to provide evidence for being compliant to the reference process.

Predictable: ML4 additionally demands measurable results. Therefore, data for the defined key performance indicators (KPI) have to be collected systematically. In this respect control data, result data and – if needed – data from external sources should be integrated and made available in electronic and structured form for documentation purpose and further statistical analysis (strict data integration and strict documentation of data and operations). Furthermore, corrective action is demanded in case of KPI exceptions. This requirement is covered through the support functions coordination and documentation and can be modeled through definition of respective controlling tasks and appropriate recording of recognized deviations and taken countermeasures.

Innovating: While corrections and improvements in ML4 only affect the currently running instance, ML5 demands (automatic) continuous improvement and innovation of the reference process und future instances. Within the scope of the support functions this paper deals with there are no further requirements through ML5.

The ML requirements on process support are summarized in Table 1. They can be used as a common scale for the evaluation of the value proposition of implementations. In the next section, it is outlined how to decide which implementation is the most appropriate to a particular situation.

2.3 Decision

In this section, a procedure for reaching a decision and finally selecting an adequate implementation for process support is suggested.

The principle objectives for the decision about adequate process support stated by Seitz and Jablonski (2012) therefore indicate a general direction. On the one hand, process support must guide the attainment of the process strategy, e.g. to reach a specific ML. On the other hand, implementations are in need of a process model that is defined properly and completely. Following the principle of utility maximization and cost minimization, the implementation approach must be chosen that fits best the desired process maturity (nothing more and nothing less) and simultaneously requires the least modeling effort and gains the broadest possible acceptance by the users. So the decision process may look as follows:

As a first step – assuming the demanded ML is set by management or other stakeholders – it must be decided to which ML the process support is adapted. Thereby, it is differentiated between the promoted quality at runtime (through information provision, data integration and coordination) and the proven quality afterwards (through adequate documentation). As a rule, it is quite useful to make sure that the aspired ML is properly documented. For example, if ML3 is demanded by the customer, all relevant process perspectives for ML3 should be traceable and comprehensible in the requisite degree of detail. However, at runtime, under certain conditions it may be sufficient to promote a lower quality in favor of lower costs and higher execution flexibility. Depending on the granted freedom, quality is just supported, rather covered or even enforced. It should be worked out to what extent "undefined paths" should be secured or should remain flexible to create necessary space for creativity. For example, although ML3 is demanded process execution is only supported according to ML1 or ML2, because the participants are in need of a certain creative freedom and their scope of actions must not be restricted through standardization. The decision also depends on the risk for process errors and their consequences. Consequently, the tighter process execution must be secured the more the demanded ML should be adapted. Documentation should always be compliant to the demanded ML.

The second step involves the assessment of possible implementation approaches with respect to their supported MLs for each QM support function. Therefore, each approach is first classified based on the framework outlined in Section 2.1 (see also Figure 1) in order to determine "how much" process support is delivered. Then, based on this classification, the approaches are evaluated according to the ML requirements on process support discussed in Section 2.2 (see also Table 1) to find out which ML can be reached.

The highest benefit is realized when information provision, data integration, coordination and documentation are implemented at the best with the required ML. The third step will therefore be to limit possible implementation approaches to the ones

Table 1: Process support requirements of the maturity levels.

	Information Provision	Data Integration	Coordination	Documentation
ML1	None	None	None	None
ML2	Organization and Behavior	None	Organization and Behavior	Organization and Behavior
ML3	All perspectives of the reference process	Data and Operations	Organization and Behavior	All perspectives of the reference process
ML4	All perspectives of the reference process	Data and Operations (strict)	Organization and Behavior	All perspectives of the reference process (thereof Data and Operations strict)
ML5	All perspectives of the reference process	Data and Operations (strict)	Organization and Behavior	All perspectives of the reference process (thereof Data and Operations strict)

providing the best “fit” with both the required runtime and documentation quality.

In a fourth step, the costs are evaluated for each selectable implementation. It is appropriate to reflect costs by the efforts for process model engineering. Those can be measured in view of two dimensions: One major factor the costs depend on is how complete the process model must be. The degree of completeness is related to the scope, especially which process perspectives must be modeled, and the detail for each perspective (e.g. task assignment to non-agents vs. agents). Another major factor is the needed degree of formalization, in particular what proportion of the process model must be interpretable by technical means with regard to system-controlled execution. Furthermore, modeling costs may vary according to the used modeling paradigm. While “for an imperative model, every possible path must be foreseen […] and encoded explicitly”, “in declarative modeling, on the contrary, only undesired paths and constellations are excluded so that all remaining paths are potentially allowed and do not have to be [defined] individually” (Schönig et al., 2012). Finally, extraordinary costs have to be considered when the implementation does not entirely promote the desired maturity and some support functions have to be carried out manually in order to be compliant to the desired ML nevertheless. In turn, there probably are also unnecessary costs when the implementation “overfulfills” the desired maturity. To sum up, on the one hand, there are investigated modeling costs that rise with the increasing demand of completeness (strictness of execution) and the growing degree of formalization (use of IT), and, on the other hand, follow-up costs for insufficient or exceeding support due to deviations from the target maturity.

In the last step, the cost-benefit-ratio is determined and the implementation promising the most reasonable ratio is chosen. Thereby, it can be considered if it is useful to take a loss of quality fit in favor of lower costs. For example, in contrast to WfMS a paper-based support tool indeed does not reach high maturity runtime support but can be implemented at significantly lower costs.

3 ILLUSTRATION

In the following, the application of the concept presented in the previous chapter is illustrated. Firstly, existing implementations which are generally accepted and recognized as enactment approaches for process management as described by Schönig et al. (2012) are introduced and classified according to their degree of process support. Secondly, it is evaluated which quality requirements they promote. Finally, the decision-making is illustrated using the example of the CL as potential implementation for standardized process execution according to ML3.

3.1 Classification

In the following, different implementation approaches are described and classified according to the previously introduced framework in Section 2.1. The selected approaches can be considered to be representative, because according to Jablonski (2010) they cover the whole spectrum of process usage from human-controlled to system-controlled (degree of IT assignment) and from flexible to strict execution support (degree of freedom).

Wallpaper (WP): The WP approach provides various possibilities to present processes visually and to depict compressed information (Information Provision) with both low (flexible) and high (strict) detail. It uses the process model “as it is, e.g. printed out as wallpaper, outlined on a flip chart or published online as process graphic in wiki” (Seitz and Jablonski, 2013). It is one strength of the WP to outline the process flow precisely and to strictly state the behavior, even though “the process itself happens completely offline” (external enactment). This is why data integration, coordination and documentation are not supported.

Checklist (CL): “A checklist comprises the main process steps including documents that must be produced and agents that are responsible to perform the corresponding process” (Jablonski, 2010). With the process steps being serialized the process behavior can only be specified roughly by the arrangement so that the actual execution order remains flexible. Depending on its implementation suitable or mandatory tools can be additionally stated. Similar to the wallpaper approach the checklist is enacted externally and therefore cannot support data integration or coordination. The documentation support is designed to collect the executing agents (organization), the sequence of execution (behavior, e.g. via timestamps) and optionally a statement of used applications (operations).

Process Navigation System (PNS): This approach is intended to support flexible, human-centric processes. It suggests suitable actions and tools and refers to restrictions, but never enforces them (Schönig et al., 2012). Hence, the PNS supports flexible coordination by recommending a set of process steps (flexible behavior), normally in interaction with human agents (strict organization). Standardized data interfaces and automatic execution of 3rd party tools are not intended, whereas adequate document and application links are contextually provided (flexible data integration). It is therefore perceived as a decision support system. The feedback of the users about the actual process flow is made available electronically as structured data and utilized to document the order of the executed process steps (strict behavior), the performing agent (strict organization) and the usage of applications and tools (flexible operations).

Workflow Management System (WfMS): Traditional WfMS strictly execute the specified workflow logic and thereby communicate with human users and IT applications (Schönig et al., 2012). Due to rigid runtime control functions this approach can be classified as strict coordination support. Documents, databases and 3rd party tools can be connected via pre-defined interfaces (strict data integration). "Most WfMS log data on cases and tasks executed" (van der Aalst, 2004). Besides (unstructured) result data like documents there are also collected (structured) control data about the interaction with human users and external systems that can be used as documentation and further analysis. For this reason, documentation support of WfMS covers all perspectives in high detail (strict).

The classification of the implementation approaches for each support function is summarized in Figure 2, Figure 3, Figure 4 and Figure 5. The figures show the relevant detail from the classification framework for each support function (see Figure 1). On the X axis the support function is drilled down on the associated process perspectives. The Y axis differentiates between flexible and strict execution. Furthermore, the enactment type (external and/or internal) is labeled. The figures illustrate "how much" process support is delivered by the investigated approaches. In the next section it is shown how this classification combined with the ML requirements on process support is used to determine which MLs the approaches are able to reach.

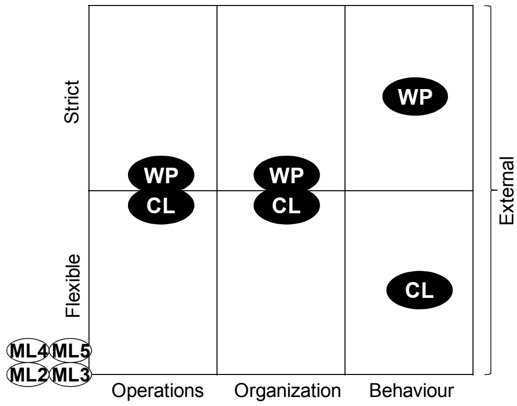


Figure 2: Information Provision.

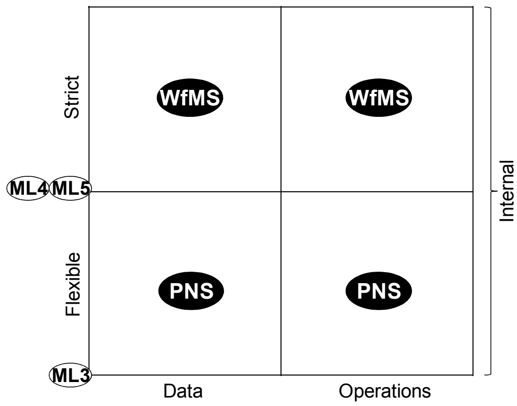


Figure 3: Data Integration.

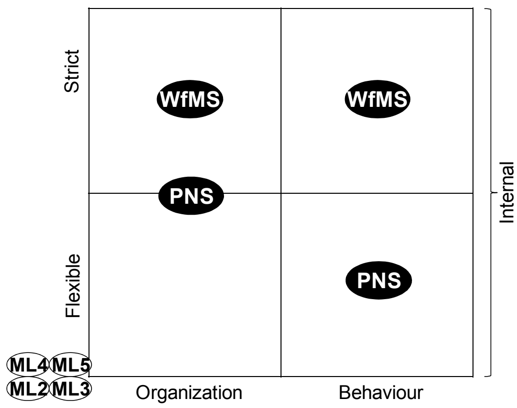


Figure 4: Coordination.

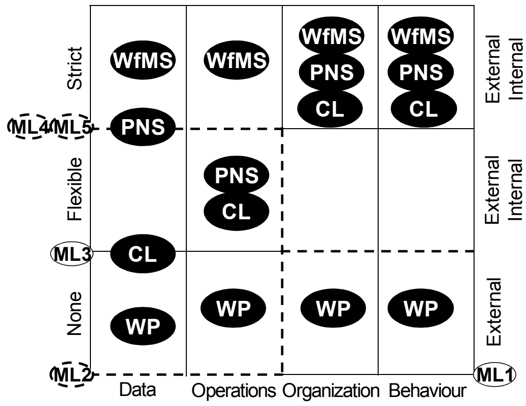


Figure 5: Documentation.

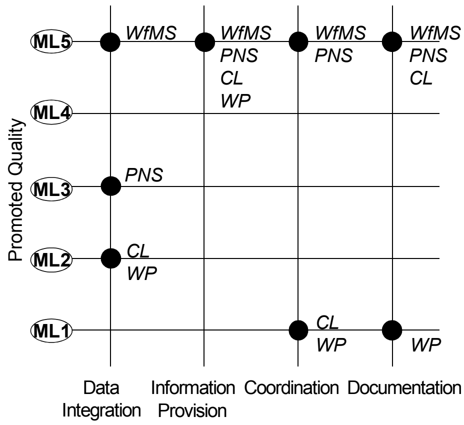


Figure 6: Promoted Quality (Value).

3.2 Evaluation

In Figure 2, Figure 3, Figure 4 and Figure 5 the MLs are located as discussed in Section 2.2 and summarized in Table 1. An implementation approach is considered to reach a specific ML for a particular support function if it is classified above the ML line.

Principally, all implementation approaches fulfill ML1. As for ML2, the deployment of the CL and the WP should be accompanied by appropriate coordination (e.g. manually by a project leader).

Both the WfMS and the PNS approach support ML3 per se, while as for the CL and the WP – along with coordination – there is also a lack of central and standardized access to process data and 3rd party tools.

ML4 is only covered by WfMS, whereas PNS could be extended by adequate interfaces in order to not only establish ML4 through coordination and documentation but also promote ML4 quality through enabling access to all required data source.

In Figure 6 the evaluation is summarized. It shows the assignments of support function and ML for each implementation approach. ML1 is reached by each approach. ML2 is fulfilled by WfMS, PNS and CL (CL in case of additional coordination support only). ML3 is implemented by WfMS and PNS. ML4 and ML5 are only reached by WfMS.

Whether the value is actually appropriate for a specific process case depends on the desired ML of the process and the cost-benefit ratio of the possible implementation approaches. An exemplary decision is outlined in the next section.

3.3 Decision

In the preceding section the value proposition of the implementation approaches was discussed, in particular how the support functions for QM are fulfilled and which MLs can be reached. Now, in this section, it is illustrated how the evaluation results can be used in order to decide whether an approach is appropriate to a particular situation. The evaluation matrix in Figure 6 serves as decision support for the identification of adequate process support for a specific ML. Therefore, the deviations (both gaps and exceedings) of the accomplished maturity in comparison with the required maturity are analyzed, as to whether they result in additional costs. An example for ML3 and the CL approach is depicted in Figure 7. There are following deviations:

Gap 1 arises from the lack of a central provision of document templates according to the reference model. One solution would be to place the CL items directly into the header of the resulting documents. The templates could be published electronically as download in the enterprise intranet or printed out and handed over as paper-based forms. Gap 2 is due to the missing coordination function of the CL approach. It can be closed by appointing a project manager that allocates tasks according to the CL and monitors that the process flow ranges in the course of the reference model. As a consequence, there arise additional costs for the provision of the document template and project management in order to “secure” process execution against mistakes.

Exceeding 1 and Exceeding 2 can be compensated by adequate configuration of the CL. Information should only be stated and documented if it is actually needed in order to meet ML3. For example, if the reference process does not prescribe a specific order of process steps it is not worth to inquire date and time of finished tasks when filling in the CL. Hence, unnecessary costs can be avoided.

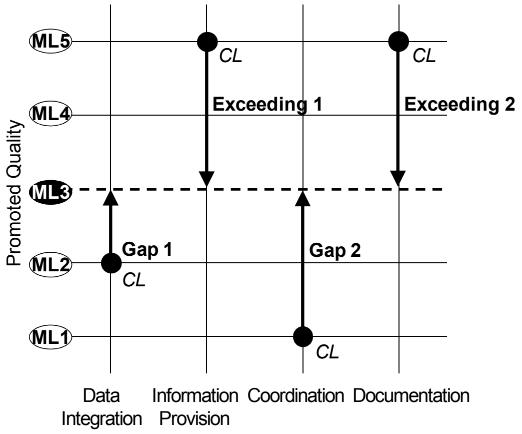


Figure 7: Gap/Exceeding Analysis (Example).

In summary, it can be seen that the CL approach – provided that coordination support at runtime can be neglected for the respective use case or substituted by manual project management – is indeed appropriate for ML3, as it enables sufficient flexibility in modeling information provision and documentation support.

4 CONCLUSION AND OUTLOOK

In this paper an approach for the analysis on the value of process support implementations for QM was introduced. On the basis of a classification framework it was shown how to determine the degree of process support differentiating between execution and documentation assistance (i.e. information provision, data integration, coordination and documentation). With the help of the BPMM requirements on process support a comparative basis for the selection of adequate implementations was created and applied to the Wallpaper, Checklist, Process Navigation System and Workflow Management System, which are four representative implementations. Finally, a procedure for decision-making was suggested and some cost aspects were discussed. The illustration of the concept and the decision revealed that the Checklist is definitely appropriate to support process execution and documentation according to ML3 (standardized).

The analysis approach can be enhanced in several respects. Besides modeling efforts also "soft" factors like user acceptance and general conditions like political constraints can be additionally considered for the cost assessment. Furthermore, it could be deliberated about whether the scope of supportive functions should be extended by, e.g., incorporation of improvements into the running process (in terms of exception handling and individual optimization) and into the reference model (in terms of adaptation and global optimization). Currently, the analysis approach is designed for a snapshot. Future work could be concentrated on how to take a more dynamic, prospective view on the evaluation (e.g. long-term planned maturity).

REFERENCES

Becker, J., Uthmann, C. v., Zur Mühlen, M., Rosemann, M. (1999). *Identifying the Workflow Potential of Business Processes*. In Proceedings of the 32nd Hawaii International Conference on System Sciences.

Bruin, T. de, Rosemann, M., Freeze, R., Kulkarni, U. (2005). *Understanding the Main Phases of Developing a Maturity Assessment Model*. In Proceedings of the 16th Australasian Conference on Information Systems (ACIS), Sydney.

Faerber, M. (2010). *Prozessorientiertes Qualitätsmanagement. Ein Konzept zur Implementierung*. Wiesbaden: Gabler.

Hogrebe, F., Nüttgens, M. (2009). *Business Process Maturity Model (BPMM): Konzeption, Anwendung und Nutzenpotenziale*. HMD – Praxis der Wirtschaftsinformatik, 266, 17-25.

Jablonski, S. (1994). *MOBILE: A Modular Workflow Model and Architecture*. In Proceedings of the 4th International Working Conference on Dynamic Modeling and Information Systems, Noordwijkerhout.

Jablonski, S. (2010). *Do We Really Know How to Support Processes? Considerations and Reconstruction.* In Engels, G., Lewerentz, C., Schäfer, W., Schürr, A., Westfechtel, B. (Eds.), Graph Transformations and Model-Driven Engineering. Essays Dedicated to Manfred Nagl on the Occasion of his 65th Birthday (pp. 393-410). Berlin/Heidelberg: Springer.

Jablonski, S., Goetz, M. (2008). *Perspective Oriented Business Process Visualization*. Business Process Management Workshops – BPM 2007 International Workshops, BPI, BPD, CBP, ProHealth, RefMod, semantics4ws, Brisbane, Australia, September 24, 2007, Revised Selected Papers. Lecture Notes in Computer Science (Vol. 4928, pp. 144-155).

Mauch, C., Wildemann, H. (2007). *Wettbewerbsfaktor IT.*

Wege zur erfolgreichen IT-Gestaltung; Ergebnisse einer empirischen Untersuchung. München: TCW, Transfer-Centrum.

Object Management Group (OMG) (2008). *Business Process Maturity Model (BPMM), Version 1.0 (2008)*. Retrieved from: http://www.omg.org/spec/BPMM/1.0/ [17 March 2013].

Schönig, S., Seitz, M., Piesche, C., Zeising, M., Jablonski, S. (2012). *Process Observation as Support for Evolutionary Process Engineering*. International Journal on Advances in Systems and Measurements, 5(4), 188-202.

Seitz, M., Jablonski, S. (2012). *Evolutionäres Prozess-Engineering: Zum angemessenen Grad an Prozessunterstützung*. HMD – Praxis der Wirtschaftsinformatik, 287, 93-102.

Seitz, M., Jablonski, S. (2013). *Evolutionary Process Engineering: User Guide and Case Study for Adequate Process Support*. In Proceedings of the 3rd International Conference on Business Intelligence and Technology, Valencia. (forthcoming)

Trkman, P. (2010). *The critical success factors of business process management*. International Journal of Information Management, 30(2), 125-134.

van der Aalst, W.M.P. (2004). *Business Process Management: A Personal View*. Business Process Management Journal, 10(2), 135-139.

Zur Mühlen, M., Rosemann, M. (2000). *Workflow-based Process Monitoring and Controlling – Technical and Organizational Issues*. In Proceedings of the 33rd Hawaii International Conference on System Sciences.

A Case Study on Entropy Generation during Business Process Execution
A Monte Carlo Simulation of the Custom Bikes Case

Peter De Bruyn, Philip Huysmans and Herwig Mannaert
Department of Management Information Systems, University of Antwerp, Prinsstraat 13, Antwerp, Belgium
{*peter.debruyn, philip.huysmans, herwig.mannaert*}*@ua.ac.be*

Keywords: Business Process Design, Entropy, Monte Carlo Simulation, Normalized Systems, Cost Accounting.

Abstract: Contemporary organizations require high-quality information to design and manage their business processes. Important challenges in this context comprise (1) the decision regarding which information should be stored and in what way, and (2) the need to allow adequate reporting for different organizational perspectives. To tackle these issues, we proposed in previous work applying the concept of entropy as defined in statistical thermodynamics to the domain of business process management. In this paper, we further elaborate on this idea by performing a Monte Carlo simulation of the Custom Bikes case to show how guidelines are necessary to control this entropy. In doing so, we extend previous theoretical contributions by releasing some simplifying assumptions made earlier, while simultaneously proving its practical relevance in a case. Finally, this paper discusses the important challenge for the need of adequate reporting from different organizational perspectives.

1 INTRODUCTION

Contemporary organizations require high-quality information to design and manage their business processes. Recently, approaches for gathering and storing large amounts of data have improved significantly. For example, Radio Frequency Identification (RFID) allows to systematically collect data at predefined locations. This data can be stored in "data warehouses", which can manage large amounts of data. However, arbitrary data gathering and storage is not sufficient. For example, the current interest in "Big Data" indicates that many organizations have issues to extract relevant information from the data available. Important challenges in this context comprise (1) the decision regarding which information should be stored and in what way, and (2) the need to allow adequate reporting for different organizational perspectives (e.g., management versus operational versus accounting purposes).

While many different approaches and tools are suggested to handle these issues in practice (e.g., business intelligence tools), a theoretical understanding of its complexity is often lacking. Such a lack obstructs the systematic research and development of methods to address this issue in a scientific way, as proposed by, for example, the Design Science methodology (Hevner et al., 2004). Therefore, we previously introduced entropy as a possible candidate for studying the structure of data generated in business processes (De Bruyn et al., 2012; Oorts et al., 2012). Within the BMSD community, other engineering concepts such as the concept of homeostasis have been suggested as well (Regev et al., 2012). This fits within the Enterprise Engineering paradigm, which states that organizations should be purposefully designed, and that engineering knowledge can be used to better understand organizational phenomena or design organizational artifacts. Baring this in mind, we earlier introduced a set of general prescriptive design guidelines which can be derived from the application of entropy to the design of business processes, resulting in a fine-grained modular structure (De Bruyn et al., 2013).

In this paper, we further elaborate on this idea by demonstrating these guidelines in a realistic case study. This implies abandoning some of the simplifications made in previous work (e.g., considering a single process flow). Moreover, this approach enables more realistic insights regarding which information should be stored (and how), as well as the impact of integrating multiple design perspectives. We employ a case study of which fine-grained business processes have been published previously (Van Nuffel, 2011). First, we generate a realistic data set for the processes. Next, we illustrate how violations against our design principles (to ensure an adequate modularization of the registered data) result in an increase of entropy during execution-time (thereby demonstrating theoretic relevance), resulting in a loss of certain organizational insights (demonstrating practical relevance).

In the remainder of this paper, we first explain our proposed application of the entropy concept to business processes (Section 2). Next, we introduce the case study (Section 3) and the data generation and set-up (Section 4). Section 5 covers a discussion of our findings and the implications for research and practice. Finally, we offer our conclusions in Section 6.

2 ENTROPY GENERATION DURING BUSINESS PROCESS EXECUTION

Entropy as expressed in the second law of thermodynamics is considered to be a fundamental principle in traditional engineering sciences. While many versions exist, all approaches have basically the intent of expressing the (increasing) amount of complexity, uncertainty (i.e., lack of information) and tendency of particles to interact (i.e., couple) within a system. In this paper, we use the statistical thermodynamics perspective towards entropy. Here, entropy was defined as proportional to the number of possible *microstates* (i.e., the whole of microscopic properties of the particles of a system) consistent with a single *macrostate* (i.e., the whole of externally observable and measurable properties of a system) (Boltzmann, 1995). The amount of microstate configurations which result in the same macrostate is called the *multiplicity* of that macrostate. A common way of dealing with entropy, is to increase the *structure* or the knowledge of the internals of the system. Structure can be used to control entropy by allowing less interaction between the constituting components before the information is observed. This way, as fewer microstate configurations remain consistent with a single macrostate, less uncertainty remains. The mechanisms related to entropy reasoning have found their reflection in many domains, including business and management topics (Jacquemin and Berry, 1979). Even for the business process management domain, some contributions can be found (Jung et al., 2011).

As our application of entropy to business processes has been discussed in detail in previous publications (De Bruyn et al., 2012; Oorts et al., 2012; De Bruyn et al., 2013), we only recall our reasoning here briefly. First, the run-time instantiation space of a business process needs to be defined (De Bruyn et al., 2013). The modular structure of a business process has been defined to consist of process modules and task modules (Van Nuffel, 2011). Therefore, the design of a business process (such as the `Order Handling` process in Figure 1) and its constituting tasks need to be defined (e.g., we used BP_1 for a business process and t_1 for a task). Moreover, the run-time instantiation space requires the identification of each instantiation of the business (e.g., instantiation $BP_{1,1}$ process and its tasks). Therefore, a possible business process instantiation space might be:

$$\begin{cases} BP_{1,1} = \{t_{1,1}, t_{2,1}, t_{3,1}, t_{5,1}\} \\ BP_{1,2} = \{t_{1,2}, t_{2,2}, t_{4,1}, t_{5,2}\} \\ BP_{1,3} = \{t_{1,3}, t_{2,3}, t_{3,2}, t_{5,3}\} \end{cases}$$

Second, the interpretation of microstates and macrostates has been elaborated upon (De Bruyn et al., 2012). A microstate of a process is the union of values of properties for each individual process particle (i.e., a task instantiation). A macrostate refers to the observable information of the process as a whole (e.g., throughput time or quality measures). Third, a set of aggregation dimensions has been proposed, which indicate how the multiplicity of a business process macrostate increases when information of the task instantiations is aggregated. Fourth, four principles have been discussed to structure the task information: (1) Separation of States (SoS): states should be introduced as measuring points throughout a process; (2) Separation of Concerns (SoC): a unique state should be introduced for separating information regarding each individual relevant information unit or concern; (3) Data instance Traceability (DiT): each business process and task instance should be linked to its information input and the information object (specifics) it is operating on; (4) Action instance Traceability (AiT): each task instantiation should be linked to the specific business process instantiation it is embedded in. Principles 1 and 2 relate to the Normalized Systems (NS) principles (Mannaert et al., 2011) at design-time. In contrast, principles 3 and 4 can only be applied in the run-time instantiation space (Mannaert et al., 2012). While a conceptual example was already discussed (De Bruyn et al., 2013), a call was made for the application of the entropy concept to a more realistic case, which will be part of the present paper.

3 CASE DESCRIPTION

The "Custom Bikes case" used in this paper has originally been introduced by Van Nuffel (2011) and describes the business processes of a small company producing customized bicycles. In order to adapt the

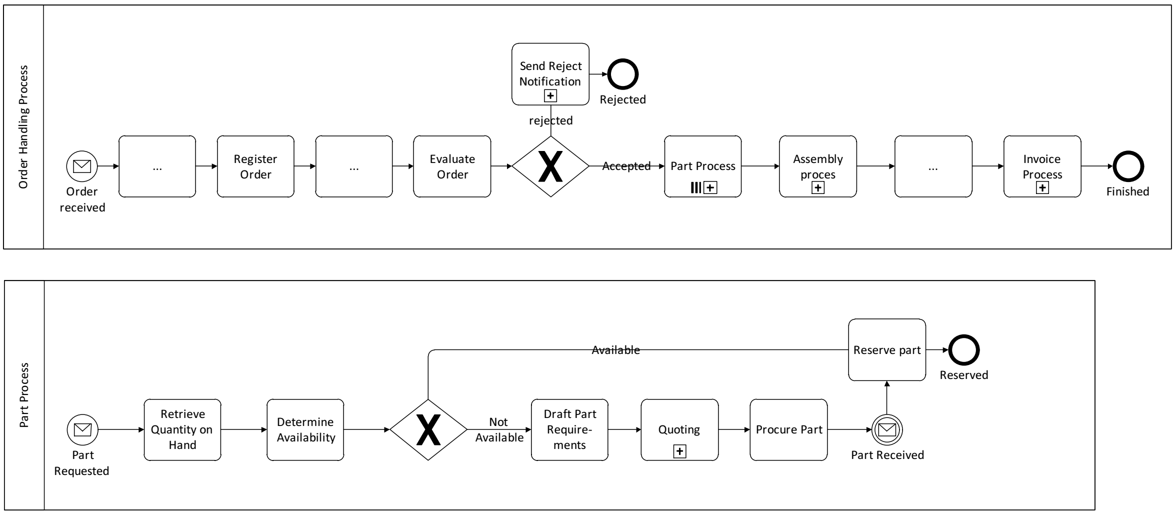


Figure 1: BPMN representations of the business process types `Order Handling` and `Part`.

complexity of the case to the available page limit, we focus on the `Order Handling Process`. Further, a set of assumptions were made to limit the complexity added in the run-time analysis (see Table 1). The process, depicted in Figure 1, describes the receipt of an order, which triggers some administrative processes or tasks (i.e., the first three tasks). Next, the order is evaluated and, if it is accepted, the parts for the order are reserved. Since a part is a different *concern* (i.e., it is located on a different aggregation level (Van Nuffel, 2011, p. 199)), it is represented in a different business process, i.e., the `Part Process`. For every part which is requested, the stock availability is checked. If a part is available, it is reserved immediately. Otherwise, quotes are requested from different suppliers, and the part is ordered from of the suppliers. When the part is received, it is reserved as well. As soon as all the parts are reserved, the bicycle is assembled, and an invoice is sent.

Our case selection was primarily motivated by the fact that these business processes already exhibit a highly fine-grained modular structure, designed to prevent the occurrence of combinatorial effects. Such effects are defined as the need to adapt multiple processes when a single change is required. For example, if the `Part Process` would be included in the `Order Handling Process`, other processes requiring parts (such as a `Repair Service Process`) would need to include these process steps as well. Consequently, a change to the `Part Process` (e.g., adding a task) would need to be applied in two different processes. The elimination of combinatorial effects has been shown to be a prerequisite to achieve systems theoretic stability (Mannaert et al., 2011). Further, it has been shown that the fine-grained modular structures needed to obtain stability in the design, also tend to reduce entropy at run-time (Mannaert and Verelst, 2009). Therefore, while the run-time perspective of entropy might require the application of additional principles, we hypothesize that the modular business process design as proposed by Van Nuffel already exhibits a low amount of entropy and constitutes a good starting point for our case study on entropy occurrence during business process execution.

4 SIMULATION SET-UP

In order to demonstrate entropy-related benefits when adopting normalized business processes (i.e., exhibiting stability and low entropy), a set of already normalized business processes should be started from. However, to our knowledge, no organizations seem to employ fully normalized business processes. Therefore, we chose to perform a simulation of the described artificial case study as they were already normalized in earlier work based on the concept of stability (Van Nuffel, 2011). In terms of data and process simulation method, a Monte Carlo experiment was conducted. Such method generally consists of a certain type of computational algorithm, employing a (high volume) repeated random sampling with the aim to obtain numerical data to mimic, for instance, the probabilities of certain real-life processes. The use of Monte Carlo experiments in the context of business processes can be noted in literature as well (Tumay, 1996). Typically, the Monte Carlo method is adopted to solve mathematical or operational problems for which no straightforward analytical solutions are within reach. While the Monte Carlo method is

Table 1: Case assumptions.

1. A part represents the most atomic unit for assembling a bike (i.e., no "sub-assemblies" exist).
2. Each bike consists out of three part types (i.e., wheels, chain wheel and frame).
3. The case organization has a fixed supplier who delivers once in a month a limited set of the most frequently used parts. However, as the organization is specialized in delivering highly-customized bikes, it frequently occurs that this monthly-based inventory appears to be insufficient or that the part is so rarely used that it is not included in the monthly inventory fill-up. When a part type is not available in stock, a quoting process takes place, in which a number of possible suppliers is asked to make a quotation for delivering the considered part. The best supplier (based on a rather complex and human-based decision) is chosen, after which a procurement order is placed.
4. For every part which is decided to be procured externally, at least one supplier will be available.
5. In case multiple parts are ordered from the same supplier, the same delivery duration for all parts is assumed.
6. A theoretically infinite amount of concurrent production capacity is assumed. Hence, no bottlenecks due to lacking manufacturing resources (i.e., place, personnel, etcetera) can occur.
7. No measuring errors are taken into account. Hence, all recorded data and possible extreme values are deemed to be correct.
8. Only accepted order instantiations are considered (i.e., only positively evaluated orders are considered in our data analysis), this means that task 3 (i.e., "Send Reject Notification") is typically not executed.

perfectly suited to generate samples of complex probability distributions, we chose to generate our data primarily for variables assumed to be normally distributed, as this distribution seems both sufficient to illustrate our entropy reasoning adequately and rather realistic when analyzing aspects like throughput time or costs related to (tasks of) a business process.

We will now successively discuss which parameters were used as an input for the generation of the data in the experiment (i.e., the "preprocessing"), the different simulation scenarios which were set-up to be able to demonstrate and analyze our entropy approach (i.e., the "simulation") and some overall observations regarding the eventually generated data (i.e., the "postprocessing"). The actual discussion and analysis of our findings will be described in Section 5.

4.1 Preprocessing

As stated earlier, we primarily assumed normally distributed variables to generate data reflecting the throughput time and costs associated with each task instantiation in an instantiated business process. Hence, for each of the task types we determined a population mean for its throughput time variable, as well as a standard deviation and an absolute minimum (to prevent throughput times from becoming negative in extreme cases). Consequently, for the task types "Register Order", "Evaluate Order", "Retrieve Quantity on Hand", "Determine Availability", "Draft Part Requirements", "Quoting", "Procure Part", "Reserve Part", "Assembly Process" and "Invoice Process", the mean, standard deviation and absolute minimum are depicted in Table 2. For all these tasks, except "Assembly Process", we assumed that the costs are proportional to the duration of the activity (as these tasks are solely relying on human activities). In such case, the cost (expressed in euro) was equal to 30 times the duration (expressed in number of individual man days). For the tasks "Receive Part" and "Assembly Process", we modeled our simulation in such way that costs were based on a distinct normal distribution as they were assumed to be equal to the purchasing costs (for "Receive Part") or related to other resources than pure manpower such as machinery (for "Assembly Process").

Further, in case a part should be ordered from a supplier, the decision regarding which supplier would be chosen was considered to be based on a rather complex and purely human decision (cf. Table 1), and the choice was hence modeled to be evenly distributed among all suppliers (i.e., each having a 20% chance for each procurement instance of a part type). We can see that each supplier has its own mean delivery time and standard deviation. One can further notice that, while the mean delivery time of supplier 5 is not spectacularly higher than that of the other suppliers, its standard deviation is. In practical business terms, this means that this supplier is the most unreliable one based on its delivery policy. Clearly, organizations aiming to offer their products or services quickly, need to have reliable and fast-delivering suppliers. Hence, for our considered bike producing company, this is deemed to be a problematic situation. The underlying question which we ask ourselves during the analysis of the case study is: to which degree can this problematic situation be observed and traced in different possible business process designs?

In summary, the parameters as depicted in Table 2 served as input variables to generate a number of simulations, discussed in the next subsection.

4.2 Simulation Scenarios

In order to simulate the processes as depicted in Figure 1, a number of $n = 100$ process instantiations was generated. Consequently, for each of the tasks t_1, t_2, t_4 (being the result of tasks t_5, t_6, t_7, t_8, t_9, t_{10} and t_{11}),

Table 2: Parameters used as input for the Monte Carlo simulation.

task		**duration (man days)**			**cost (€)**		
		mean	st. deviation	minimum	mean	st. deviation	min
Register Order	(t_1)	0.0625	0.0125	0.00625		$\sim$ time	
Evaluate Order	(t_2)	0.1250	0.0420	0.0125		$\sim$ time	
Send Reject Notification	(t_3)		not relevant			not relevant	
Part Process	(t_4)		depends on tasks 5 till 11			depends on tasks 5 till 11	
Retrieve Quantity on Hand	(t_5)	0.0500	0.0100	0.0050		$\sim$ time	
Determine Availability	(t_6)	0.0500	0.0100	0.0050		$\sim$ time	
Draft Part Requirement	(t_7)	0.2500	0.0950	0.025		$\sim$ time	
Quoting	(t_8)	2.5000	0.4000	0.2500		$\sim$ time	
Procure Part	(t_9)	0.2500	0.0800	0.1000		$\sim$ time	
Receive Part	(t_{10})		depends on supplier chosen		250	40	15
Reserve Part	(t_{11})	0.0500	0.0100	0.0005		$\sim$ time	
Assembly Process	(t_{12})	0.5000	0.2000	0.0050	250	50	50
Invoice Process	(t_{13})	0.1250	0.0420	0.00125		$\sim$ time	
Delivery time supplier 1		3	0.3	0.3			
Delivery time supplier 2		2.8	0.3	0.28			
Delivery time supplier 3		2.6	0.4	0.26			
Delivery time supplier 4		3.2	0.4	0.32			
Delivery time supplier 5		3.6	4	0.36			

t_{12} and t_{13}, a task instance duration and associated cost had to be generated for each process instantiation. This was done by associating a random number $x \in [0,1]$ with each of these task instantiations. Next, for each of these instances, the value z_t was calculated for which there is a probability of x in the normal distribution of the considered task t that a lower value is generated, so that for instance: $P(cost(t_1) \leq z_t) = x$. Hence, by using this procedure, for each of the task instantiations, a cost and throughput value is assigned based on a random sampling from their normal distribution. Also for supplier selection, a random number $x \in [0,1]$ was generated. Based on this value, supplier 1 was assigned if $x \in [0;0,25]$, supplier 2 was assigned if $x \in]0,25;0,50]$, etcetera.

Based on these cost and throughput values per task instance, a set of performance variables could easily be calculated, such as total cost price of a product instance, total throughput time for a product instance, mean cost price of a certain product type, mean throughput time of a certain product type, mean throughput time of a certain task type, etcetera. Three different case study scenarios were considered based on this initial simulation. Each time, the same set of randomly generated "basis" data was used. However, some different aggregations were performed in each of the case scenarios (cf. infra), implying a different degree of detail or granularity regarding their resulting variables/indicators and hence, as we will show, entropy. Each of these aggregations was based on one of the aggregation dimensions we proposed in De Bruyn et al. (2013), in which we illustrated the business relevance of each of them. Afterwards, for each of the considered case study scenarios, the necessary performance variables could be easily calculated by summing the respective task (instances). We will now highlight each of these case study scenarios:

Scenario 1. This scenario represents the most fine-grained variant of our case study and the registered information exactly mimics the granularity as represented in Figure 1, applying the guidelines as proposed in the work of Van Nuffel (2011). Hence, no summation on top of these fine grained tasks or their instances is made. Also, information regarding the data used for each task instantiation as well as the business process instance to which it belongs, is persisted. Consequently, no entropy should be able to occur as each of the NS entropy principles is adhered to. This scenario corresponds to an example of aggregation dimension 1 in De Bruyn et al. (2013).

Scenario 2. In this scenario two activities within the `Part Process` are combined ("Procure Part" and "Part Received") into a new task labeled t_{19}. This means that, for each process instance, no separate state is kept between the execution of these steps and, hence, no independent information on the execution of these both tasks is available for the observer. This scenario corresponds to an example of aggregation dimension 2 in De Bruyn et al. (2013). However, as this might cause information loss regarding two (arguably) relevant parts in the process, this scenario is expected to create a certain degree of entropy due to SoS and SoC violations.

Scenario 3. In this scenario, the data regarding the `Part Process` execution does not include instance specific information such as: for which

`Order Handling Process` instance was a particular `Part Process` instance carried out? Was the `Part Process` instance aimed at providing the wheels, frame or chain wheel? In case the Part was externally procured, which supplier was chosen to do so? In other words, information regarding the execution of each of the specific tasks in the business process flow is mainly available "at bulk" without any reference to the specifics of each individual instance. This scenario corresponds to an example of aggregation dimension 4 in De Bruyn et al. (2013). However, as this might cause information loss regarding (arguably) relevant information in the process, this scenario is expected to create a certain degree of entropy due to AiT and DiT violations.

4.3 Postprocessing

Some descriptive statistics of the generated data can be found in Figure 2. Obviously, these data and generated sample distributions highly approximate the population distributions as set out in Section 4.1. However, while this might initially occur to the reader as a circular reasoning, the whole point of our reasoning is exactly the fact that the observer in a realistic case is not aware of the actual population distributions and should perform his analysis solely based on the information registered during the execution of the business process instantiations. This allows to mimic the situation in which, for instance, extreme cases (e.g., high throughput or delivery time) or other problems (e.g., an unreliable supplier) are generated by the simulation model and a business analyst is aiming to trace (i.e., diagnose) the origin (i.e., microstate) of this observed fact (i.e., the macrostate).

5 FINDINGS AND DISCUSSION

Let us assume that the manager of our simulated bike selling company observes (macroscopically) that too many deliveries of requested bike orders exceed 7.5 man days in order to be delivered. This can be seen in Figure 2a en 2b, where a line is drawn at this point. Indeed, the boxplot (a) is asymmetrical and its upper whisker is rather long, indicating that some outliers or extreme values are present. Also the histogram (b) shows that at least 10 out of the 100 simulated bike order instantiations exceed this border value. The manager considers this situation harmful, as customers typically tend to file complaints starting from this moment. Therefore, he wants to find out what the cause of these extreme throughput values for the `Order Handling Process` is. In Section 4.1, we explained how we configured the parameters of our simulation model in such way that the standard deviation of the delivery time of supplier 5 is extremely large, compared to the other four suppliers. In each of the different case study scenarios, the process and the possibility to fulfill the managers' assignment to locate this problem differ. We will consider each of them consecutively.

5.1 Scenario 1: The Normalized Case

In scenario 1, all information depicted in Figure 2 (i.e., panels a till i) is (macroscopically) available for the observer for solving his problem. Indeed, information regarding all fine-grained tasks, as implied by the application of the guidelines of Van Nuffel (2011), and their instantiations is registered and available for further inspection by the observer. First, panels a and b indicate an asymmetric delivery time and a large number of outliers or deliveries exceeding the desired 7.5 man days. For instance, `Order Handling` instance 38 was indicated to have a delivery time of almost 15 man days.

In a next stage, the observer would logically opt for splitting out the different tasks in the `Order Handling Process` in order to dig down to the origin of the observed problem. Here, panel c indicates that task 4 —the "Part Process" (i.e., being the aggregation of the completion of the `Part Process` instances needed to proceed with the "Assembly Process" task)— not only represents the largest mean amount of time for the whole `Order Handling Process` but also, and probably more importantly, exhibits a very large deviation. Indeed, the histogram of panel d shows that a large amount of instances of task 4 has a throughput time less than one man day (i.e., the cases where all three parts are in stock). However, also a relatively fair amount of instances require more than 6 man days to complete, implying a large chance that the total delivery time will also exceed the target delivery time.

The next logical step would be to split up the steps constituting the `Part Process` as is depicted in panel e. One can see that primarily the tasks "Draft Part Requirements" and "Receive Part" take some time to be executed, but that the "Procure Part" task has large deviations. Having identified the most fine-grained task responsible for the extremely delayed `Order Handling Process` instances, a process design exhibiting (data) instance traceability would allow an even more detailed analysis showing the throughput time for this task, categorized by the supplier chosen when the a part is externally procured

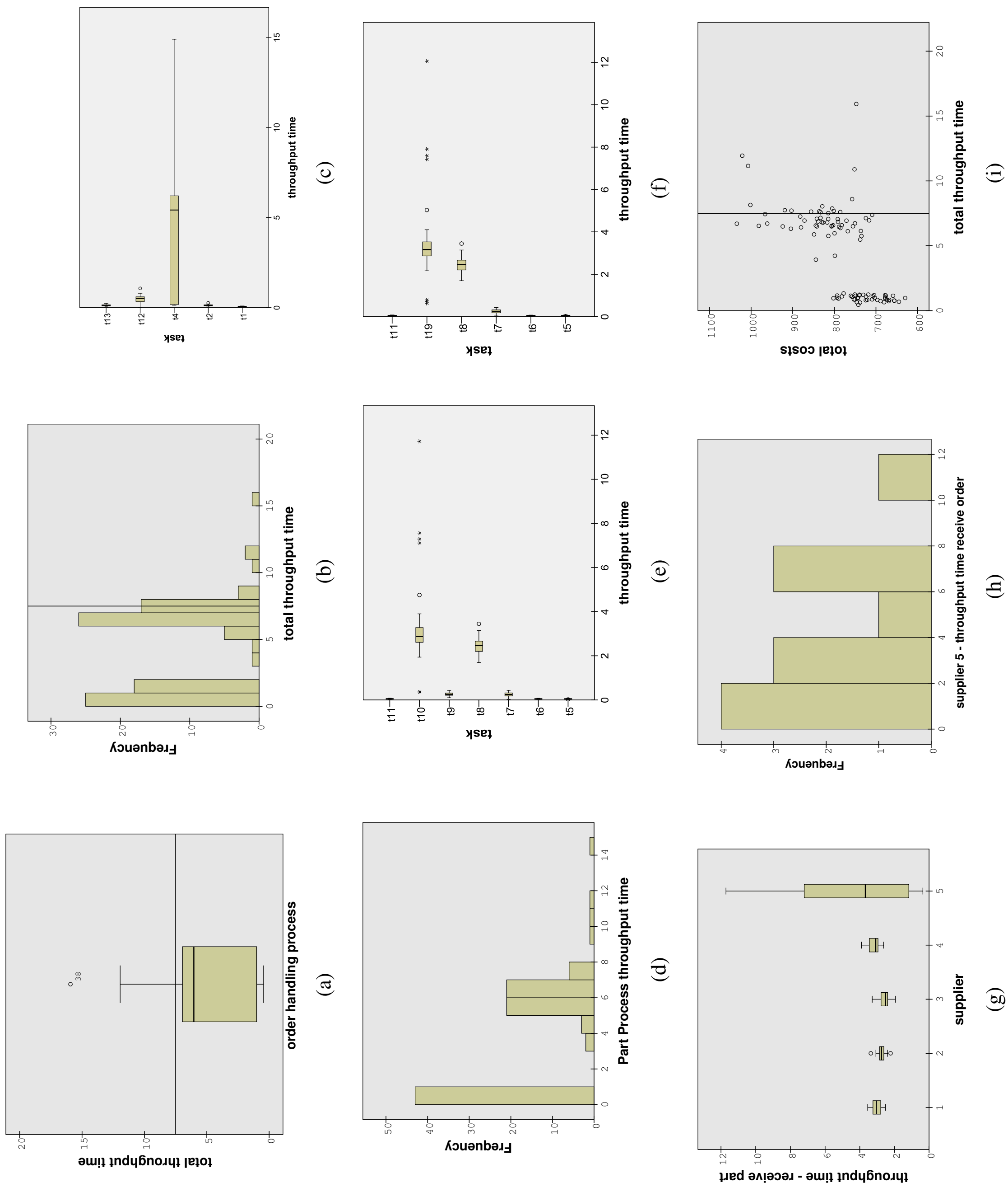


Figure 2: Some descriptive statistics of the Custom Bikes case simulation.

(i.e., panels g and h). It is only at this stage of the reasoning, that one is able to identify the root cause of the problem introduced in our simulation model (i.e., the unreliability of supplier 5 in terms of delivery time).

Consequently, both the observability and traceability of the introduced problem are rather straightforward in this scenario. Indeed, in case supplier 5 would be removed from the part suppliers portfolio, the extremely irregular delivery time would almost completely disappear, solving the originally identified problem. Hence, the multiplicity in this scenario is 1 (and entropy 0) as the macrostate (i.e., the identified problem of a set of extreme throughput times of the `Order Handling` Process) can be unambiguously related to one microstate (i.e., the configuration in which one task and its instance data (such as supplier chosen) can be attributed as causing the problematic macrostate, the other tasks being normal). This can be seen at two distinct levels: (1) at the type level of processes and tasks (i.e., identifying the process and task type responsible for the deviations in the observed total throughput time) and (2) at the instance level of processes and tasks (e.g., to trace the fact that for `Order Handling` instance 38, 11.7 man days of the total throughput time were spent while waiting for the delivery of Part 3 for which supplier 5 was chosen). Finally, from panel i, we can derive that focusing on the cost perspective would not have been sufficient to correctly diagnose this delivery related problem as many extreme deliveries have only low or moderate registered costs (e.g., the costs for `Order Handling` instance 38 only amounts to 747 €).

5.2 Scenario 2: Violation Regarding SoS/SoC

Scenario 2 represents the scenario in which a violation towards the principles SoS and SoC is introduced as information regarding two independent information units are combined, i.e., the "Procurement Part" and "Part Received" tasks. Consequently, the information (macroscopically) available for the observer is restricted to panels a, b, c, d, and f in Figure 2. The problem analysis in this scenario will proceed analogously to the procedure outlined for scenario 1, until the point that the different constituting tasks of the `Part Process` are to be examined for their respective throughput times. However, at this stage, the observer will not be able to differentiate between potential problems regarding the "Procure Part" task (e.g., do the people at the procurement department face difficulties for drafting and placing the orders, or do they have a bottleneck making each procurement wait for a certain amount of time before it can be placed?) or "Part Received" task (e.g., are there any kind of problems related to the delivery itself?). That way, one cannot be sure whether the problem is situated under ones own responsibility (i.e., the procurement department) or an external party (i.e., the supplier) and entropy clearly increases. Indeed, while the observer in this example is still able to be aware of a problem regarding the overall throughput time of the `Order Handling Process` (the macrostate), multiple (i.e., minimum 2) independent causes (i.e., tasks) can be deemed to be at the origin of this phenomenon (i.e., both procurement or delivery). As a consequence, for each `Part Process` instantiation, multiplicity equals 2, entropy rises and the traceability of the problem decreases. Consequently, in a good business process design, these information units should be separated into distinct tasks to allow for the above mentioned analyses. Interestingly, the "*Actor Task Responsibility*" guideline proposed by Van Nuffel (2011) would already recommend this separation as the procuring and the delivery of parts are carried out by different actors, in this case even from different organizational entities.

5.3 Scenario 3: Violation Regarding DiT/AiT

Scenario 3 represents represents the scenario in which a violation regarding the DiT and AiT principles is introduced. Indeed, DiT requires that the relevant data for executing a task is registered and related to the relevant state, whereas AiT implies that the execution of a task instance should be unambiguously traceable to the business process it is embedded in. This requirement is not met for the `Part Process` in this scenario. Consequently, in this scenario, the information (macroscopically) available for the observer is restricted to panels a, b, c, d, and e in Figure 2. The problem analysis in this scenario will therefore proceed analogously to the procedure outlined for scenario 1, until the point that the "Receive Part" task becomes identified as being responsible for the extreme values regarding overall throughput times of the `Order Handling Process`. However, at this stage, the observer will not be able to differentiate between possible problems related to the specifics of the different suppliers. That way, the only conclusion which can be drawn from this data is that some problems regarding the delivery of parts are present. Indeed, while the observer in this example is again able to be aware of a problem regarding the overall throughput time of the `Order Handling Process` (the macrostate), multiple (i.e., minimum 5) independent causes (i.e., suppliers) can be deemed to be at the origin of this phenomenon (i.e., is the problem supplier specific and if so, which supplier?). As a consequence, for each `Part Process` instantiation, multiplicity equals 5, entropy rises and the traceability of the problem decreases. In a good business process design, these information aspects (i.e., data) should be traceable to their corresponding task instance executions to allow for the above mentioned analyses. In fact, the "*Aggregation Level*" guideline proposed by Van Nuffel (2011), would at least suggest to separate the delivery of parts in a distinct business process of which the instances are logically linked to their "parent" business process instance. Additionally, we would recommend to register for each task instantiation the business process instance it belongs to as well as the argument data it employed for triggering its execution.

Finally, imagining the scenario in which "Part Process" is only considered as one atomic tasks, clearly constitutes an example of a violation of both the SoS/SoC and DiT/AiT design principles, thereby combining the difficulties of traceability discussed for scenarios 2 and 3 and further increasing the degree of entropy. Further, while in all three case scenarios, the observability of a problematic situation was still present, we showed in De Bruyn et al. (2013) that an increase in entropy might also lead to a lower degree of traceability. In our case example, this might be show up in case we did not include one unreliable supplier, but instead one supplier having a higher mean delivery time compared to all other suppliers in the supplier portfolio: this would lead to unnecessarily high (mean) delivery times without the manager or process owner being necessarily aware of it because no "extreme" cases or outliers would be apparent and no supplier-specific analysis could be done.

5.4 Reflections

At the end of Section 5.1, we already briefly mentioned that a focus on different perspectives when collecting data from business processes is necessary. Different stakeholders in an organization require different kinds of information which should all be considered when collecting data. For example, business process throughput optimization, cost accounting (e.g., requiring part costs and activity drivers) and manufacturing perspectives all require different data. In our case example it was for instance shown that problematic high throughput times of a process do not necessarily imply high execution costs and vice versa, as can be seen from Figure 2i. The combination of these different perspectives is not straightforward, and is not focused on in many research projects. In dedicated research domains (such as operations research or management accounting), a degree of specialist expertise is needed which does not necessarily allow insight in other fields. While we do not claim to have extensive expertise in these domains, the focus on multiple different perspectives demonstrates the emergence of issues which are specific to the *integration* of different domains. Such integration issues represent wicked problems. Even in research domains which focus on integration, it is claimed that only local optimizations are known, or that integration is studied based on one dominant perspective (Ethiraj and Levinthal, 2004).

Such integration problems can be explained clearly based on our theoretical framework. Consider the application of the first two principles presented in Section 2. These principles require the identification of the different concerns of a certain perspective, and separating them by explicitly keeping states. However, the concerns which are identified will vary for different perspectives. The design of a business process which needs to be analyzed in a business process optimization perspective will focus on concerns which influence the throughput of the process. However, when these different tasks all have the same activity driver for an Activity-Based Costing system, their separation is not required based on the application of the Separation of Concerns principles from a cost accounting perspective. For example, similar to the aggregations performed in case scenario 2, the sequence of tasks regarding the smaller administrative tasks (e.g., tasks "Register Order" and "Evaluate Order") could be combined from a cost accounting perspective, because (a) they do not need to be booked in separate accounts, since only the cost of a product as a whole is relevant; (b) they are all performed by the same type of resource (i.e., a human resource), which indicates a similar activity driver; and (c) the costs of these tasks equals "labor time" times "labor cost per time unit". From a process optimization perspective, different arguments for separating these tasks could be raised. For example, different employees can perform these tasks, which can cause different throughput times. Moreover, from an organizational diagnostic perspective, these tasks should be separated as well: this would allow an often-occurring erroneous process outcome to be traced back to a single task, or a single employee. For instance, in our Custom Bikes case, managers would typically like to be informed when the "Register Order" and "Evaluate Order" tasks have significantly different throughput and/or success rates. In contrast, concerns which need to be separated from a cost accounting perspective (e.g., tasks with different activity drivers) may not be relevant to the business process optimization perspective (e.g., when they do not add to throughput time). Additionally, similar to the aggregations performed in case scenario 3, a manufacturing or assembly viewpoint might consider the data regarding which supplier delivered a specific part as irrelevant (assuming that each of the 5 considered suppliers in our case deliver identical parts). However, we showed in our discussion that such (data) instance traceability can be relevant for other perspectives (such as business process optimization in terms of throughput time) as it was actually at the core of our simulated problem: one of the considered suppliers turned out to be rather unreliable in terms of its delivery time.

Therefore, the only way to gather data which provides sufficient insights for different perspectives simultaneously, is by separating the union of all concerns of these perspectives. Based on these "atomic" information units, different aggregations can be derived to provide the required information of each perspective. Such detailed data could be useful even for perspectives where certain concerns do not need to be separated at this moment. Consider an organization which needs to switch from a European way of reporting to an American one. Different accounts may be necessary in that case. However, the atomic information units can be re-aggregated in a different way to comply with the new regulations. Moreover, in Europe many organizations perform two sets of bookkeeping: a national and a European one. Instead of requiring (partly duplicate) data input in separate systems, such systems should be built to be able to handle the aggregation of the same atomic information units. Of course, the selection of such type of data requires a highly structured approach aiming at gathering consistent and fine-grained data. However, regarding this granularity in for instance the cost accounting field, it has been acknowledged that the "activities" identified in Activity-Based Costing (ABC) systems, are to be understood as being "composed of the *aggregation* of units of work or tasks" (Drury, 2007, p. 342). Therefore, the "activities" proposed can probably not be considered as the "atomic" information units, certainly if one's aim is to reuse these information units for deriving insight from the different perspectives mentioned above. Further, it has been reported that the collection of fine-grained data for an Activity-Based Costing system can become highly complex (Kaplan and Anderson, 2004). However, consider the structured way of designing software architectures using Normalized Systems theory (Mannaert and Verelst, 2009). Reusable building blocks for certain software functionality, called elements, describe the modular structure of all concerns from different relevant perspectives (such as remote access, security, etcetera) which need to be handled for performing, for example, a software action. This forces designers to develop software based on such fine-grained modular elements. The incorporation of a reporting concern within these software elements (typically supporting a set of business processes) could for example provide possibilities to obtain the required fine-grained data in organizations, as suggested in this paper. The design of such elements on an organizational level itself has been explored as well (De Bruyn, 2011; De Bruyn and Mannaert, 2012). To obtain such elements, reusable building blocks should be designed which perform a generic organizational action and which handle relevant cross-cutting concerns, such as logging relevant data. In our example, the part reservation process could be such a reusable element, which keeps data concerning a.o. its throughput time, and the cost of the reserved part.

6 CONCLUSIONS

In this paper, we aimed to provide evidence for entropy generation during the execution of business processes if NS theory principles are not adhered to, by performing a Monte Carlo simulation of the Custom Bikes case. Hence, this paper has several contributions. First, this paper validates in a more practical and empirical way conceptualizations proposed in earlier work. Second, we extended this conceptualization of entropy reasoning for business process modeling by releasing several simplifying assumptions (i.e., considering multiple (interacting) business processes and allowing different mean (duration or cost) parameters for each task type). Third, we elaborated on the need to split tasks and business processes up to the level of so-called "atomic tasks", which should be able to be considered as atomic from several perspectives simultaneously (e.g., cost accounting and operational business process optimization perspectives). In future research, it might be interesting to propose a set of more specific (business-oriented) guidelines to delineate business processes and their constituting tasks, as Van Nuffel (2011) did for the stability reasoning in NS, and contrast both approaches Clearly, the aim of this paper was not at all to employ complex or advanced statistical methods to perform data mining on the information traces delivered by business process execution. Instead, we consciously employed only the most basic statistical reasoning techniques to show that the identification and registration of the basic ("atomic") data regarding business process execution needs a sufficient amount of attention from multiple perspectives, before entropy reduction and relevant data analysis can occur.

ACKNOWLEDGEMENTS

P.D.B. is supported by a Research Grant of the Agency for Innovation by Science and Technology in Flanders (IWT).

REFERENCES

Boltzmann, L. (1995). *Lectures on gas theory*. Dover Publications.

De Bruyn, P. (2011). Towards designing enterprises for evolvability based on fundamental engineering concepts. In *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, volume 7046 of *Lecture Notes in Computer Science*, pages 11–20. Springer, Berlin, Heidelberg.

De Bruyn, P., Huysmans, P., Mannaert, H., and Verelst, J. (2013). Understanding entropy generation during the execution of business process instantiations: An illustration from cost accounting. In *Advances in Enterprise Engineering VII*, Lecture Notes in Computer Science. Springer Berlin Heidelberg. in press.

De Bruyn, P., Huysmans, P., Oorts, G., and Mannaert, H. (2012). On the applicability of the notion of entropy for business process analysis. In Shishkov, B., editor, *Proceedings of the second international symposium on Business Modeling and Software Design*, pages 128–137.

De Bruyn, P. and Mannaert, H. (2012). On the generalization of normalized systems concepts to the analysis and design of modules in systems and enterprise engineering. *International journal on advances in systems and measurements*, 5(3&4):216–232.

Drury, C. (2007). *Management and Cost Accounting*. South-Western.

Ethiraj, S. K. and Levinthal, D. (2004). Bounded rationality and the search for organizational architecture: An evolutionary perspective on the design of organizations and their evolvability. *Administrative Science Quarterly*, 49(3):404 – 437.

Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1):75–105.

Jacquemin, A. P. and Berry, C. H. (1979). Entropy measure of diversification and corporate growth. *The Journal of Industrial Economics*, 27(4):359–369.

Jung, J.-Y., Chin, C.-H., and Cardoso, J. (2011). An entropy-based uncertainty measure of process models. *Information Processing Letters*, 111(3):135–141.

Kaplan, R. S. and Anderson, S. R. (2004). Time-driven activity-based costing. *Harvard business review*, 82(11):131–8.

Mannaert, H., De Bruyn, P., and Verelst, J. (2012). Exploring entropy in software systems: towards a precise definition and design rules. In *Proceedings of the Seventh International Conference on Systems (ICONS) 2012*, pages 93–99.

Mannaert, H. and Verelst, J. (2009). *Normalized Systems—Re-creating Information Technology Based on Laws for Software Evolvability*. Koppa.

Mannaert, H., Verelst, J., and Ven, K. (2011). The transformation of requirements into software primitives: Studying evolvability based on systems theoretic stability. *Science of Computer Programming*, 76(12):1210–1222.

Oorts, G., Huysmans, P., and De Bruyn, P. (2012). On advancing the field of organizational diagnosis based on insights from entropy - motivating the need for constructional models. In Shishkov, B., editor, *Proceedings of the second international symposium on Business Modeling and Software Design*, pages 138–143.

Regev, G., Hayard, O., and Wegmann, A. (2012). Homeostasis - the forgotten enabler of business models. In Shishkov, B., editor, *Proceedings of the second international symposium on Business Modeling and Software Design*, pages 13–24.

Tumay, K. (1996). Business process simulation. In Charnes, J., Morrice, D., Brunner, D., and Swain, J., editors, *Proceedings of the 1996 Winter Simulation Conference*, pages 93–98. ACM Press.

Van Nuffel, D. (2011). *Towards Designing Modular and Evolvable Business Processes*. PhD thesis, University of Antwerp.

A Survey of Formal Business Process Verification
From Soundness to Variability

Heerko Groefsema and Doina Bucur
Johann Bernoulli Institute for Mathematics and Computer Science,
University of Groningen, Nijenborgh 9, 9747 AG Groningen, the Netherlands
{h.groefsema, d.bucur}@rug.nl

Keywords: Business Process Management, Verification, Model Checking, Survey.

Abstract: Formal verification of business process models is of interest to a number of application areas, including checking for basic process correctness, business compliance, and process variability. A large amount of work on these topics exist, while a comprehensive overview of the field and its directions is lacking. We provide an overview and critical reflections on existing approaches.

1 INTRODUCTION

Business process modeling helps businesses to increase the quality of their processes. Formal techniques are used to model, implement, execute, and monitor business process models. *Model checking* is a technique which verifies a given system model for compliance with a specification of interest, for various practical *goals* including ensuring basic correctness of processes, business compliance checking, and process variability. A related survey (Morimoto, 2008) provides an overview of business process checking, but does not consider compliance and variability as supported through formal verification, while in (Aiello et al., 2010), we survey variability for business processes.

While a large amount of work exists in the field of business process verification, it lacks an overview of the state of the field and its related formal verification frameworks. As such, in the present treatment, we aim to provide an overview of formal verification goals, techniques, and frameworks for business process modeling, and give critical reflections.

Frameworks aiming at the verification of business process models exist. They supports various process-specification formalisms, e.g., *imperative*, *declarative*, *event-driven*, or *artifact-centric*. In *imperative* specification formalisms, processes are modelled as sets of *tasks* or *activities*, *gates*, and *events* interlinked by *flows* or *transitions*. Each activity describes a single unit of work and the transitions describe the order between these units of work. Common notations include Business Process Model and Notation (BPMN) (OMG, 2011), Business Process Execution Language (BPEL) (Oasis, 2007), Unified Modeling Language (UML) activity diagrams (OMG, 2011), and Yet Another Workflow Language (YAWL) (Hofstede et al., 2010). Alternatively, *declarative* specification formalisms, such as (Pesic and van der Aalst, 2006), model processes without distinct flow controls which specify order between units of work. Instead, these specifications express a process model as a set of activities and a set of *constraints* over these activities, with the constraints restricting the possible inclusion and ordering of the activities. Any process behaviour not prohibited by these constraints is valid.

Event-driven specification is another approach to business process modelling. Defined in (Keller et al., 1992), Event-driven Process Chains (EPC) are directed graphs mainly consisting of events, functions (activities), and logical connectors (gates). Unlike imperative specifications, EPC do not model node ordering explicitly. Although EPC are known for their intelligible notation and simplicity, their lack of semantics is a topic of discussion (van der Aalst, 1999). Finally, *artifact-centric* specifications focus on the evolution of business entities and data. Originally proposed in (Nigam and Caswell, 2003), such specifications incorporate the notion of the lifecycle of business artifacts, such as data—which is ignored by most other specifications.

In what follows, Section 2 examines the techniques and goals of business processes verification. We classify and discuss verification frameworks in Section 3, and conclude in Section 4.

2 BUSINESS PROCESS MODEL CHECKING

Business processes verification can be based on several algorithmic techniques (and some supporting intermediary modelling formalisms), and be used for a number of goals. Next we overview these.

2.1 Verification Techniques

Petri Nets are state-transition systems used to analyze distributed systems. They are amenable to intuitive graphical notation which, unlike most of the process notations discussed in Section 1, include a mathematical definition of its execution semantics. A number of subclasses of Petri Nets have been defined, most notably the sub-class of Workflow (WF) nets (van der Aalst, 1998) which is applied as an intermediary formalism in the verification of business processes. Besides Petri Nets, a system may also be modelled as a *finite state machine* (FSM)—a directed graph of nodes and edges, with nodes representing a system state system and edges representing a change in state. Both Petri Nets and FSM-like models are then verified against a *specification* or *correctness property*.

On the other hand, generic model checkers, e.g., SPIN (Holzmann, 2004) and nuSMV2 (Cimatti et al., 1999), implement search algorithms to verify any system modelled in the model checker's *input languages*. Of these, notable is the Process Meta-Language, or Promela, used by SPIN. Business processes can be remodelled using Promela, with the Promela implementation then internally translated into an automaton and verified against a correctness property.

Correctness properties may be informal specifications, process models, or logic formulas. Informal specifications include properties defined as simple tuples or programming methods. Process models themselves can be used as correctness properties as well. In this case, the original business process model is verified to be a refinement of the correctness model. Logic properties are formulas in logics ranging from propositional logic to *deontic* and *temporal logic*. Deontic logics reason about obligations and permissions. Temporal logics include Linear Temporal Logic (LTL) and Computation Tree Logic (CTL), a branching-time logic. LTL specifies properties (e.g., the universality of a certain state property, and the order of states) over states occurring on process execution paths. CTL extends this set of temporal operators with path quantifiers, such that formulas can specify properties over branching executions. Extensions of these logics are common for process verification. They include Past-time LTL (PLTL), but also novel logics for business process specification.

2.2 Goals of Verification

Business process verification is the act of determining if a business process model is correct with regard to a set of formal correctness properties. Often, verification is automated by tools known as analyzers or model checkers. Several goals for using verification are presented in the business process literature.

The first goal is verifying basic properties such as *reachability* and *termination*. Reachability of a business activity requires an execution path to exist leading to that activity starting from the initial activities. A termination property requires that all possible execution traces terminate. Business process *soundness*, a property originally proposed in the area of Petri Net verification, is known as the combination of these two properties plus a third: the absence of related running activities at process termination (i.e., proper completion). Avoiding the deployment of erroneous processes that do not conform with these properties is obviously advantageous: *"[erroneously] designed workflow models can result in failed workflow processes, execution errors, and disgruntled customers and employees"* (Bi and Zhao, 2004).

The second goal for business process verification is business *compliance*. Compliance checks whether process models conform with specifications, which in this case can be another process model or a set of rules, such as (inter)national laws and standards. When verifying compliance, rules are often specified using a formal logic over the entities (e.g., events, activities) of the business process model. In other cases, these rules are informally specified. For example, regulations could specify that before processing a wire transfer, a bank should identify if any sanctions exist regarding the involved parties.

The third goal of verification of process models, *variability*, extends upon compliance. *"In the context of BPM, variability indicates that parts of a business process remain variable, or not fully defined, in order to support different versions of the same process depending on the intended use or execution context"* (Aiello et al., 2010). Variability aims to support different versions of the same process. This includes support of process families at design-time, when a new process *variant* can be derived from a generic process, and process flexibility or adaptability at run-time, where a generic process can be adapted. Variability can be specified in two different ways. The first, which is not in the scope of this survey, employs the use of variation points to provide different options at specific points in a process. The second, which is in

Table 1: Soundness tools and frameworks.

Framework	Formalisms: Modelling	Formalisms: Intermediate	Formalisms: Correctness properties	Tool
(van der Aalst, 1998)	Petri Net			Woflan
(Bi and Zhao, 2004)	WfMC		Propositional logic	Algorithmic
(Choi and Zhao, 2005)	WfMS			
(van Dongen et al., 2007)	EPC	Petri Net		ProM
(Fisteus et al., 2005)	BPEL4WS	CFM	LTL, CTL	SPIN, SMV
(Karamanolis et al., 2000)	WfMS	FSP		LTSA
(Koehler et al., 2002)	Imperative	FSM	CTL	nuSMV
(Masalagiu et al., 2009)	BPMN	Petri Net to TLA+	LTL, CTL (via TLA+)	TLC
(Nakajima, 2006)	BPEL	EFA to Promela	LTL	SPIN
(Weber et al., 2010)	Annotated process		Annotated process	Algorithmic
(Wynn et al., 2009)	YAWL	Petri Net		YAWL

scope, uses rules like those of compliance to specify how each version of a process should behave.

A final goal of business process verification (which we do not cover in this survey) deals with processes including multiple parties, such as business process collaborations (De Backer et al., 2009). The goal of verification includes, for example, the *compatibility* between processes, or lanes.

3 OVERVIEW OF FRAMEWORKS

The existing frameworks aim to verify business processes either at design time or at runtime. The most basic form of verification of processes deals with design-time verification of soundness properties, e.g., termination and reachability. Table 1 gives an overview of these frameworks. (van der Aalst, 1998) introduced soundness to the field of BPM by translating workflows into Petri Nets, and (Wynn et al., 2009) perfected the application by allowing Or-joins and cancelation regions. Due to this, Petri Nets are commonly used as intermediate formalisms by soundness verification frameworks, including (van Dongen et al., 2007), who use it to verify EPC. Another popular method is by translating processes into a model checker input language, e.g.: (Masalagiu et al., 2009) verify BPMN by translating it (via a Petri Net intermediate model) into the model checker input language TLA+, (Karamanolis et al., 2000) translate processes to the process algebra FSP and checks the result with the Labeled Transition System Analyzer, (Koehler et al., 2002) translate into the nuSMV input language, and (Nakajima, 2006) translates into the SPIN language Promela.

Other frameworks verify business compliance; an overview of these is given in Table 2. A dominant number of compliance frameworks focus on verifying imperative specifications such as BPMN, BPEL, EPC, and UML sequence diagrams. (Anderson et al., 2005; Arbab et al., 2009; Awad et al., 2008; Foster et al., 2003; Ghose and Koliadis, 2007; Goedertier and Vanthienen, 2006; Janssen et al., 1998; Liu et al., 2007; Ly et al., 2008; Ly et al., 2011; Nakajima, 2002) all belong in this category. Others extend declarative specifications with compliance features. In (Chesani et al., 2009), compliance is modelled based on DecSerFlow, a declarative runtime process specification, by translating it into a reactive event calculus. (Pulvermueller et al., 2010) aims at verifying the compliance of design-time EPC using an extension of CTL that differentiates between events and functions. Finally, (Deutsch et al., 2009) proposes verifying the compliance of artifact-centric processes against properties expressed in an extension of LTL.

The last set of frameworks (listed in Table 3) aim at supporting variability. Declarative variability extends upon compliance by only specifying rules over the set of tasks in a process, instead of building an imperative graph. As such, the authors of (Governatori et al., 2006) first propose a compliance framework based upon the newly proposed deontic logic FCL, then continue by extending this framework, in (Governatori et al., 2011), with goals to provide a fully declarative description. (Sadiq et al., 2005) proposes defining pockets of flexibility within imperatively specified processes to introduce design-time variability, using constraints which provide ordering and inclusion information but which are not specified using a formal logic. Formal frameworks base their declarative specifications on the temporal logics LTL and CTL. As examples, (Demeyer et al., 2010) use Finite LTL to specify fully declarative processes, (D'Aprile et al., 2011) specify declarative processes using temporal Answer Set Programming (ASP) and Dynamic LTL, (Pesic and van der Aalst, 2006) use LTL to specify flexible run-time processes, and the related work of (van der Aalst and Pesic, 2006) aims towards service flows instead. Finally, (Maggi et al., 2011) extends upon the work of (Pesic and van der

Table 2: Compliance tools and frameworks.

	Formalisms			
Framework	**Modelling**	**Intermediate**	**Correctness properties**	**Tool**
(Anderson et al., 2005)	UML Sequence Diagram	CSP	CSP	FDR
(Arbab et al., 2009)	BPMN	Reo	Automata	Vereofy/ mCRL2
(Awad et al., 2008)	BPMN	Petri Net	PLTL	LoLA/ nuSMV
(Chesani et al., 2009)	DecSerFlow		Event Calc.	Algorithmic
(Deutsch et al., 2009)	Business artifacts		LTL-FO	Algorithmic
(Foster et al., 2003)	UML+ BPEL	FSP		LTSA
(Gerede and Su, 2007)	Business artifacts		CTL-FO	Algorithmic
(Ghose and Koliadis, 2007)	Annotated BPMN	Annotated digraphs	Process effect rules	Algorithmic
(Goedertier and Vanthienen, 2006)	BPMN		PENELOPE	Prolog CLP(fd)
(Janssen et al., 1998)	AMBER	Promela	LTL	SPIN
(Liu et al., 2007)	BPEL	Pi-calculus to FSM	LTL	nuSMV
(Ly et al., 2008; Ly et al., 2011)			CRG	
(Montali et al., 2010)	LTL	ALP	ALP	SCIFF
(Nakajima, 2002)	WSFL	Promela	LTL	SPIN
(Pulvermueller et al., 2010)	EPC		EG-CTL	BAM
(Weber et al., 2008)	Annotated Process Graph			

Aalst, 2006) to provide for runtime recovery after breaking constraint compliance. (Groefsema et al., 2011) uses CTL to define process templates; processes based upon such a template are then verified for compliance with that template at design-time. Finally, (Bulanov et al., 2011) proposes Temporal Process Logic (TPL) to provide a formal mechanism supporting different gates to merge processes.

Critically, drawbacks exist in certain frameworks for compliance and variability. Some frameworks inefficiently translate the business-process model into a model checker input language, introducing a large *overhead* in the ensuing state space (e.g., a simple process of five activities and four transitions is reportedly mapped to 201 states and 586 transitions in SPIN by (Nakajima, 2002)).

Other methods for design-time verification (but not for runtime verification, which checks linear execution paths) lack good support for complex branching features of the modelling formalism (e.g., do not support parallel gates, or execution loops). To improve on this, some introduce workarounds, e.g., (Pulvermueller et al., 2010) proposes using simple variables on automata to fork exclusive paths and synchronize parallel paths. In other work (Feja et al., 2009), the same authors propose an unsound Kripke translation when dealing with parallel paths. Here, pairs of activities from different parallel paths are explicitly synchronized; in reality, however, each path should only be synchronized after a join. Other frameworks, e.g., (Sadiq et al., 2005), (Choi and Zhao, 2005), and (Groefsema et al., 2011), simply ignore exclusive, inclusive, and/or parallel paths for reasons of complexity. (Weber et al., 2008) proposes two verification algorithms in polynomial time; however, one is reported by the authors as unsound and complete, the other sound but incomplete, and both only support acyclic processes. (Montali et al., 2010) reports its verification algorithm to be unable to terminate under certain loops.

Finally, other frameworks require users to apply newly proposed or extended specification logics in order to specify correctness properties. Examples include (Pulvermueller et al., 2010), which extends CTL with the ability to differentiate between events and functions in EPC, (Governatori et al., 2006), which proposes an entirely new deontic logic, and (Bulanov et al., 2011), which proposes a new temporal process logic to allow process mergers.

4 CONCLUSIONS

Formal verification of business processes was initially proposed to check for process soundness, but lately has been deployed to also support business compliance and variability. While process soundness is well supported by frameworks based on Petri-Net formalisms, the areas of compliance and variability checking lack design-time solutions which (i) minimize the overhead of states in the formal model, and (ii) support large subsets of the business process modelling formalism, including parallel gates and process loops.

Table 3: Variability tools and frameworks.

Framework	Formalisms			Tool
	Modelling	Intermediate	Correctness properties	
(D'Aprile et al., 2011)	Temporal ASP		Temporal ASP	
(Bulanov et al., 2011)	Imperative		TPL	
(Governatori et al., 2011)	BPMN		FCL	
(Groefsema et al., 2011)	BPMN+ CTL		CTL	VxBPMN
(Demeyer et al., 2010)	Saturn	Automata	Finite LTL	Saturn Eng.
(Pesic and van der Aalst, 2006; van der Aalst and Pesic, 2006; Maggi et al., 2011)	LTL	Automata	LTL	Declare
(Rychkova et al., 2008)	BPMN + FO	Alloys	FO	Alloys
(Sadiq et al., 2005)			Informal	Chameleon

REFERENCES

Aiello, M., Bulanov, P., and Groefsema, H. (2010). Requirements and tools for variability management. In *Proc. IEEE 34th Annual Computer Software and Applications Conference Workshops*, COMPSACW '10, pages 245–250, Washington, DC, USA.

Anderson, B., Hansen, J. V., Lowry, P., and Summers, S. (2005). Model checking for E-business control and assurance. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 35(3):445–450.

Arbab, F., Kokash, N., and Meng, S. (2009). Towards using Reo for compliance-aware business process modeling. In Margaria, T. and Steffen, B., editors, *Leveraging Applications of Formal Methods, Verification and Validation*, volume 17 of *Communications in Computer and Information Science*, pages 108–123. Springer.

Awad, A., Decker, G., and Weske, M. (2008). Efficient compliance checking using BPMN-Q and temporal logic. In *Proc. 6th Int. Conf. on Business Process Management*, BPM '08, pages 326–341. Springer.

Bi, H. H. and Zhao, J. L. (2004). Applying propositional logic to workflow verification. *Information Technology and Management*, 5:293–318.

Bulanov, P., Lazovik, A., and Aiello, M. (2011). Business process customization using process merging techniques. In *Int. Conference on Service-Oriented Computing and Applications (SOCA 2011)*, pages 1–4.

Chesani, F., Mello, P., Montali, M., and Torroni, P. (2009). Web services and formal methods. chapter Verification of Choreographies During Execution Using the Reactive Event Calculus, pages 55–72. Springer.

Choi, Y. and Zhao, J. L. (2005). Decomposition-based verification of cyclic workflows. In *Proc. Third Int. Conf. on Automated Technology for Verification and Analysis*, ATVA'05, pages 84–98. Springer.

Cimatti, A., Clarke, E., Giunchiglia, F., and Roveri, M. (1999). NuSMV: a new Symbolic Model Verifier. In N. Halbwachs and D. Peled, editors, *Proc. 11th Conference on Computer-Aided Verification (CAV)*, volume 1633 of *LNCS*, pages 495–499. Springer.

D'Aprile, D., Giordano, L., Gliozzi, V., Martelli, A., Pozzato, G. L., and Dupré, D. T. (2011). Verifying compliance of business processes with temporal answer sets. In *CILC*, pages 147–161.

De Backer, M., Snoeck, M., Monsieur, G., Lemahieu, W., and Dedene, G. (2009). A scenario-based verification technique to assess the compatibility of collaborative business processes. *Data Knowl. Eng.*, 68(6):531–551.

Demeyer, R., Van Assche, M., Langevine, L., and Vanhoof, W. (2010). Declarative workflows to efficiently manage flexible and advanced business processes. In *Proc. 12th International ACM SIGPLAN Symposium on Principles and Practice of Declarative Programming*, PPDP, pages 209–218. ACM.

Deutsch, A., Hull, R., Patrizi, F., and Vianu, V. (2009). Automatic verification of data-centric business processes. In *Proc. 12th Int. Conf. on Database Theory*, ICDT, pages 252–267. ACM.

Feja, S., Speck, A., and Pulvermüller, E. (2009). Business process verification. In *GI Jahrestagung*, pages 4037–4051.

Fisteus, J. A., Fernández, L. S., and Kloos, C. D. (2005). Applying model checking to bpel4ws business collaborations. In *Proc. ACM Symposium on Applied Computing (SAC)*, pages 826–830. ACM.

Foster, H., Uchitel, S., Magee, J., and Kramer, J. (2003). Model-based verification of web service compositions. In *Proc. 18th IEEE Int. Conf. on Automated Software Engineering (ASE)*, pages 152–163.

Gerede, C. and Su, J. (2007). Specification and verification of artifact behaviors in business process models. In Krämer, B. J., Lin, K.-J., and Narasimhan, P., editors, *Service-Oriented Computing (ICSOC)*, volume 4749 of *LNCS*, pages 181–192. Springer.

Ghose, A. and Koliadis, G. (2007). Auditing business process compliance. In *Proc. 5th Int. Conf. on Service-Oriented Computing*, ICSOC, pages 169–180. Springer.

Goedertier, S. and Vanthienen, J. (2006). Designing compliant business processes with obligations and permissions. In *Proc. Int. Conf. on Business Process Management Workshops*, BPM, pages 5–14. Springer.

Governatori, G., Milosevic, Z., and Sadiq, S. (2006). Compliance checking between business processes and business contracts. In *Proc. 10th IEEE Int. Conf. En-*

terprise Distributed Object Computing, EDOC, pages 221–232. IEEE Computer Society.

Governatori, G., Olivieri, F., Scannapieco, S., and Cristani, M. (2011). Designing for compliance: norms and goals. In *Proc. 5th Int. Conf. on Rule-based modeling and computing on the semantic web*, RuleML, pages 282–297. Springer.

Groefsema, H., Bulanov, P., and Aiello, M. (2011). Declarative enhancement framework for business processes. In *Int. Conference on Service-Oriented Computing, ICSOC, LNCS 7084*, pages pp 495–504.

Hofstede, A. H., Russell, N., Aalst, W. M., and Adams, M. (2010). *Modern Business Process Automation: YAWL and its Support Environment*. Springer.

Holzmann, G. (2004). The SPIN Model Checker. A Primer and Reference Manual.

Janssen, W., Mateescu, R., Mauw, S., and Springintveld, J. (1998). Verifying business processes using SPIN. In *Proc. of the 4th Int. SPIN Workshop*, pages 21–36.

Karamanolis, C. T., Giannakopoulou, D., Magee, J., and Wheater, S. M. (2000). Model checking of workflow schemas. In *Proc. 4th Int. Conf. on Enterprise Distributed Object Computing*, EDOC, pages 170–181. IEEE Computer Society.

Keller, G., Nüttgens, M., and Scheer, A. W. (1992). Semantische Prozeßmodellierung auf der Grundlage Ereignisgesteuerter Prozeßketten (EPK). Technical Report 89, Universität des Saarlandes, Germany.

Koehler, J., Tirenni, G., and Kumaran, S. (2002). From business process model to consistent implementation: a case for formal verification methods. In *Proc. 6th Int. Conf. on Enterprise Distributed Object Computing (EDOC)*, pages 96–106.

Liu, Y., Müller, S., and Xu, K. (2007). A static compliance-checking framework for business process models. *IBM Syst. J.*, 46(2):335–361.

Ly, L. T., Rinderle, S., and Dadam, P. (2008). Integration and verification of semantic constraints in adaptive process management systems. *Data Knowl. Eng.*, 64(1):3–23.

Ly, L. T., Rinderle-Ma, S., Knuplesch, D., and Dadam, P. (2011). Monitoring business process compliance using compliance rule graphs. In *Proc. Confederated Int. Conf. On the move to meaningful internet systems - Volume I*, OTM, pages 82–99. Springer-Verlag.

Maggi, F., Montali, M., Westergaard, M., and Aalst, W. (2011). Monitoring business constraints with linear temporal logic: An approach based on colored automata. In *Business Process Management*, volume 6896 of *LNCS*, pages 132–147. Springer.

Masalagiu, C., Chin, W.-N., Andrei, S., and Alaiba, V. (2009). A rigorous methodology for specification and verification of business processes. *Formal Aspects of Computing*, 21:495–510.

Montali, M., Torroni, P., Chesani, F., Mello, P., Alberti, M., and Lamma, E. (2010). Abductive logic programming as an effective technology for the static verification of declarative business processes. *Fundam. Inf.*, 102(3-4):325–361.

Morimoto, S. (2008). A survey of formal verification for business process modeling. In *Proc. 8th Int. Conf. on Computational Science, Part II*, ICCS, pages 514–522. Springer-Verlag.

Nakajima, S. (2002). Verification of Web service flows with model-checking techniques. In *Proc. First Int. Symposium on Cyber Worlds*, pages 378–385.

Nakajima, S. (2006). Model-checking behavioral specification of BPEL applications. *Electron. Notes Theor. Comput. Sci.*, 151(2):89–105.

Nigam, A. and Caswell, N. S. (2003). Business artifacts: An approach to operational specification. *IBM Syst. J.*, 42(3):428–445.

Oasis (2007). Web Services Business Process Execution Language Version 2.0. Technical report, Oasis.

OMG (2011). Business Process Model And Notation (BPMN) Version 2.0. Technical Report 2011-01-03.

OMG (2011). OMG Unified Modeling Language^{TM} (OMG UML). Technical Report 2011-08-05.

Pesic, M. and van der Aalst, W. M. P. (2006). A declarative approach for flexible business processes management. In *Proc. Int. Conf. on Business Process Management Workshops*, BPM, pages 169–180. Springer-Verlag.

Pulvermueller, E., Feja, S., and Speck, A. (2010). Developer-friendly verification of process-based systems. *Knowl.-Based Syst.*, 23(7):667–676.

Rychkova, I., Regev, G., and Wegmann, A. (2008). Using declarative specifications in business process design. *Int. J. of Computer Science and Applications*, 5(3b):45–68.

Sadiq, S. W., Orlowska, M. E., and Sadiq, W. (2005). Specification and validation of process constraints for flexible workflows. *Inf. Syst.*, 30(5):349–378.

van der Aalst, W. M. (1999). Formalization and verification of event-driven process chains. *Information and Software technology*, 41(10):639–650.

van der Aalst, W. M. P. (1998). The application of Petri nets to workflow management. *Journal of Circuits, Systems and Computers*, 8(1):21–66.

van der Aalst, W. M. P. and Pesic, M. (2006). DecSerFlow: Towards a truly declarative service flow language. *LNCS: Web Services and Formal Methods, Volume 4184*, pages 1–23.

van Dongen, B. F., Jansen-Vullers, M. H., Verbeek, H. M. W., and van der Aalst, W. M. P. (2007). Verification of the SAP reference models using EPC reduction, state-space analysis, and invariants. *Comput. Ind.*, 58(6):578–601.

Weber, I., Governatori, G., and Hoffmann, J. (2008). Approximate compliance checking for annotated process models. In *Advances in Enterprise EngineeringProceedings of the GRCIS workshop at CAiSE*, volume 8.

Weber, I., Hoffmann, J., and Mendling, J. (2010). Beyond soundness: on the verification of semantic business process models. *Distrib. Parallel Databases*, 27(3):271–343.

Wynn, M. T., Verbeek, H., van der Aalst, W., ter Hofstede, A., and Edmond, D. (2009). Business process verification - finally a reality! *Business Process Management Journal*, 15(1):74–92.

A Method for Reengineering and Prioritizing Goal Models

Jan Willem Harmannij and Ella Roubtsova
Faculty of Informatics, Open University of the Netherlands, PO Box 2960 NL-6401 DL Heerlen, The Netherlands
j.harmannij@studie.ou.nl, ella.roubtsova@ou.nl

Keywords: Reverse Engineering, Goal Modelling, Prioritising of Goals.

Abstract: We present a method for reengineering a goal model from an existing software product, based on the ideas of GORE approaches GBRAM and KAOS. We extend the methods with application of specific ranking criteria to the goal models using Hierarchical Cumulative Voting (HCV). The reengineered goal models are used to evaluate and compare requirements for alternative goals based on the ranking criteria, thus providing input for determining the scope of new products or product versions. The method has been tested in two use cases, one of which is described here.

1 INTRODUCTION

The Belgian company Ferranti Computer Systems has developed the MECOMS™ software product, a business support system for energy and utility companies, providing among others, the Meter Data Management (MDM) and a Customer Information System (CIS). With a network of certified partners across the globe, Ferranti offers worldwide capabilities for customized implementations of the software product. These implementations are based on the standard product developed by Ferranti (Ferranti Computer Systems, 2013).

The need to develop the new software features is caused by the market changes and the demands of different stakeholders, but the budget and development capacities of the company are always limited. The MECOMS™ product organization often faces the questions: Which direction of the product development should have the highest priority and why?

Many software supplier businesses find themselves in this situation. They need a method for prioritising the activities of the product development assuming business goals. There is currently no method for reengineering goal models from existing software products, based on documentation and input from domain experts, and employing these goal models for feature selection for a next product or product version. In this paper, we propose such a method.

We started the method development with an observation that the goals from two dimensions make influence one on another: the business goals and the goals of product development. For modelling of goals in both dimensions, Goal-Oriented Requirements Engineering (GORE) (Lapouchnian, 2005) can be used. The goals in the GORE approaches are represented as trees of sub-goals refined to requirements. However, relating of two trees does not make the task of prioritizing the software development activities observable. That is why we decided to apply the GORE methods only for presentations of the software product goals and relate the business goals to criteria for prioritization. These criteria become the attributes of the software product goals. The values of criteria become the measures of priority. The choice of the criteria and the values of the criteria are fulfilled using the Hierarchical Cumulative Voting method (Berander and Jönsson, 2006) that involves participation of stakeholders. The results of prioritizing are presented in the software product goal model to lead the activities within the development team.

Paper layout: In section 2, we review the basis of goal-modelling that can be used for reengineering the goal models. In section 3, we propose a method for reverse engineering and prioritizing the goal models of a growing product. Section 4 presents the results of testing of the proposed method using two case studies taken from the MECOMS™ software product. Section 5 summarises our results and observations and concludes the paper.

2 RELATED WORK

We agree with the opinion of (Regev and Wegmann, 2005) that Goal-Oriented Requirements Engineering is an important contribution by the RE research community and provides many benefits.

Firstly, the GBRAM method (Anton, 1997) emphasizes the initial identification and abstraction of goals from various information sources, using keywords and standard questions for eliciting input from domain experts and classifying this information into concepts like goals, constraints, obstacles and requirements. GBRAM structures the goal modelling process into several phases: exploring, identifying, organizing, refining, elaborating and operationalization.

Secondly, by separating the definition of a goal and requirement, GORE gives the fundament for reasoning about the software product in hand.

Thirdly, a goal model can capture variability through the use of alternative goal refinements, separating stable information from volatile one, and making quantitative and qualitative analysis of these alternatives possible (Lapouchnian, 2005).

Goal models can be developed using a simple AND/OR tree structure, linking a goal G with sub-goals G1, G2, …, G*n*. An AND relation requires all sub-goals to be achieved, while an OR relation provides alternative options. Labelling relations with identifiers like "++" ,"+", "-" and "--", partial goal satisfaction can be modelled (Giorgini et al., 2003). There have been several GORE methods defined over the years that provide more goal modelling semantics; the most prominent being KAOS (van Lamsweerde, 2001) and i* (Yu, 1999).

Very little research is available about reengineering goal models from existing software systems. Only (Yu et al., 2005) describe a method for reverse engineering a goal model, but they focus on design recovery for undocumented software, creating goal models from legacy code using state charts and hammock graphs.

However, most software products still have requirements documented in some form, and functional experts are often willing to provide additional information. This means, that it is not necessary to rely only on legacy code reengineering the goal models. Existing documentation like requirements specification documents, user manuals, marketing brochures and product roadmaps should be used as a starting and reference point. As the domain experts often have a busy schedule, completely relying on their input is a bad strategy (John, 2006).

Prioritization of goals is not part of any of the existing GORE modelling methods.

The prioritizing is used only at the level of software product requirements. According to (Berander and Andrews, 2005), the prioritizing is often performed using a hierarchy of high- and low-level requirements, which are evaluated using methods like the Analytical Hierarchical Process (AHP) or Hierarchical Cumulative Voting (HCV). In our opinion, prioritising at the level of goals provides more information about the motivation of priorities and helps to explain the management decision.

3 A METHOD FOR REENGINEERING AND PRIORITIZING OF GOAL MODELS

In order to design our method of goal-model reengineering we have combined the useful ideas of goal-oriented approaches, and extended them with the process for goal prioritising.

Our method has two different input sorts. At the stage of initial goal model development, the input is the documentation of the software product and the approval of it by the domain experts. At the stage of prioritising, the input comes from the votes of experts. The method consists of seven phases, as shown in Figure 1.

The first four phases (1-4 in Figure 1) copy the GBRAM method (Anton, 1997), but use documentation as the input.

During the Explore phase (1), all relevant documentation is explored. This provides us with unstructured information, like the terminology used to describe functionality, core concepts, modules, etc.

In the Identify phase (2), we try to find the answers on the WHY-questions and create the first draft goal models. The draft models are created using the information obtained during the Explore phase and input from domain experts in modelling sessions.

Draft goal models are corrected by experts live during these sessions, using the KAOS modelling tool Objectiver™ (Respect-IT, 2007). One of the reasons why we have chosen to use KAOS instead other goal modelling methods is that we can easily separate the parts of goal modelling from other KAOS functionality (responsibility modelling, object modelling, operation modelling) by omitting

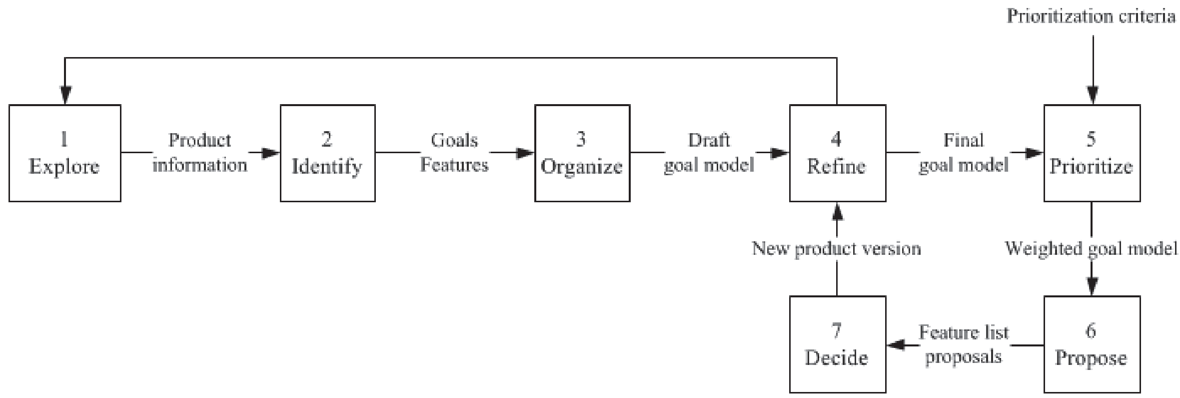


Figure 1: Method for Reengineering and Prioritizing of Goals.

relations with those parts. In other goal–oriented methods, for example in i*, modelling agent relationships and rationales is an essential activity and cannot be separated from the goal model.

An example model is visible in Figure 2. In KAOS, goals, obstacles and features are defined according to the following definitions:

- A goal (a box with thin borders in Figure 2) describes the visual part of an imaginary state of the system which can be achieved by fulfilling all sub-goals (or alternative sub-goals), or in case of the lowest-level goals, when all linked features have been completely implemented;
- An obstacle (a red box, not displayed here) describes an undesirable, but possible system state. Its sub-goals provide solutions to neutralize the obstacle.
- A feature (a thick border box) is a group of related requirements which is implemented in software (van Gurp, 2003). A feature does not describe a system state, but the required functionality for reaching that state. Features are modelled in KAOS as requirements.

The draft goal models are restructured and reorganized during the Organize phase (3). The models are separated into smaller, cross-referenced graphs, checked for completeness and consistency, and documented.

During the Refine phase (4), the organized models are presented to the domain experts and, with their input, finalized. The alternative goal refinement options are added for the next prioritising steps of our method.

The next steps of the method extend the steps of the GBRAM method. We have added these steps and an iterative cycle of their application in order to support the choice or prioritizing criteria and collecting values of those criteria.

After the Refine phase (4), the goal model of the system with alternatives of its further development is ready for the step of prioritising (5). Using a custom plugin for the Objectiver™ tool, the criteria can be added as attributes to the structures presenting goals.

First of all, the right criteria should be determined. Possible criteria include financial value, cost, urgency, readiness, and several others (Ruhe, 2011). It is highly improbable that these criteria are all equally important for the software supplier. Selecting which criteria should be used for evaluating the product features is a management decision. The input from the experts is used at this stage.

When the prioritization criteria have been determined, the feature alternatives are evaluated according to these criteria. This evaluation is the subject of a workshop session with stakeholders from the software supplier company, such as sales representatives, product managers and service managers. These stakeholders define the score of each alternative goal refinement according to the chosen prioritization criteria. We use the Hierarchical Cumulative Voting technique (Berander and Jönsson, 2006), because of its hierarchical nature that is easily combined with goal models, and its suitability for rapidly evaluating a large number of alternatives and multiple stakeholders.

The evaluation results on each criterion are normalized to a scale of 0 to 100 and can be noted as custom goal attributes in the goal models.

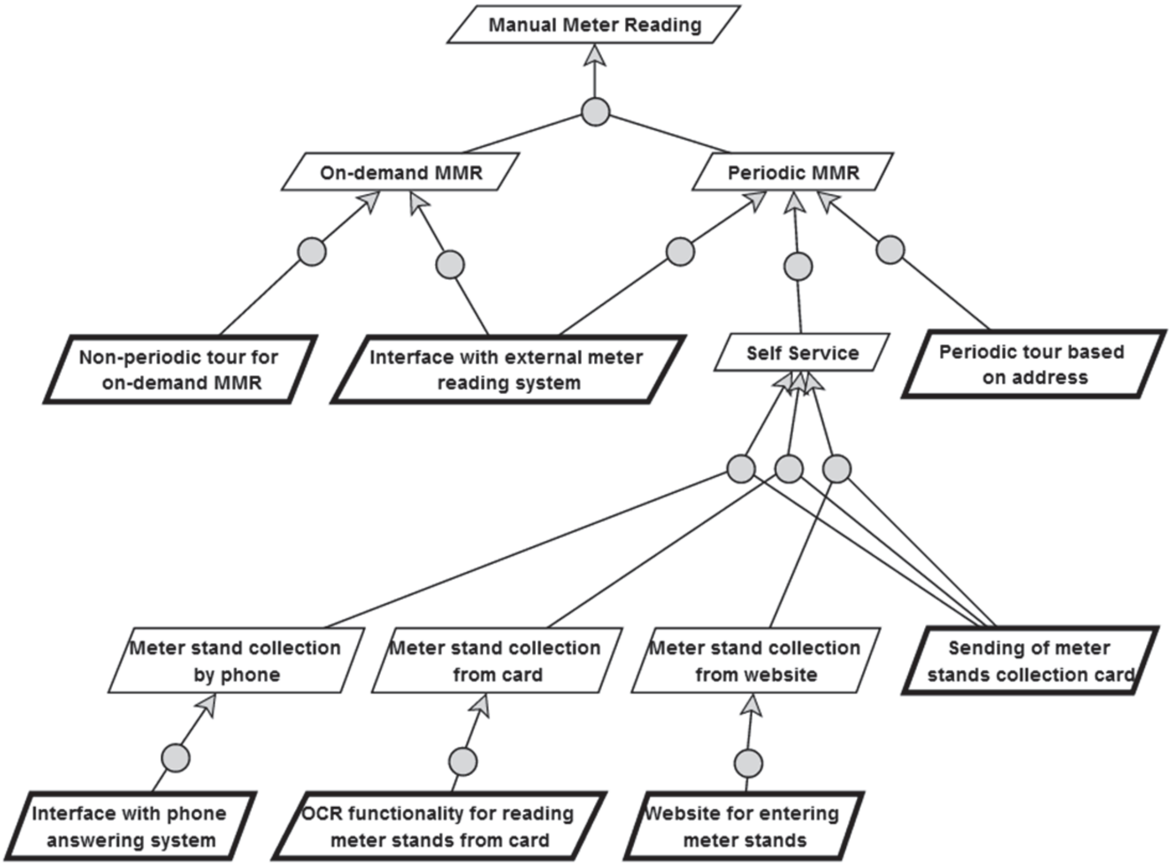


Figure 2: Part of the MDM goal model, showing "Manual Meter Reading" functionality.

The goal model is now ready to be used for the proposition of the variations that are possible for the next product instantiation or product version. Using the custom prioritization attributes from the goal model and an analyzing tool like the KAOS query language OQL, the alternative scope options can be selected and compared on the basis of the values of the prioritization criteria. This results in a document with feature list proposals and their score according to the criteria that were used for the prioritization of the goals and features.

During the final Decision phase, one of the proposals is selected by a Change Advisory Board. The model can now be refined again (back to phase 4), and a new iteration can be started.

4 APPLICATION OF THE PROPOSED METHOD

The MECOMS™ system was used in two case studies to test our method, "Meter Data Management" (MDM) and "Customer Information System" (CIS).

We describe here the MDM case, for which a partial goal model is shown in Figure 2:

1. Explore. By analyzing the documentation of the Meter Data Management (MDM) functionality, we found the concepts 'Manual Meter Reading' (MMR), 'Periodic MMR', 'On-demand MMR' and 'Self Service'. To identify goals, the documentation is analyzed by asking, "What goal(s) does this statement/fragment exemplify?" and "What goal(s) does this statement block or obstructs?"
2. Identify. The participants named additional goals and product features, which were modelled using the Objectiver™ tool on a large screen. For example, using the question "How do we provide Self Service functionality in the product?" we discovered the sub-goals "Meter stand collection by phone," "Meter stand collection from card" and "Meter stand collection from website." We then proceeded to ask the "How?" questions for the three sub-goals to formulate the features at the bottom of the model.

3. Organize. The model was then restructured into several smaller parts, and analyzed for missing goals and obstructions. These were identified by asking, "Are there alternative ways for achieving this goal?" and "Could this goal somehow be obstructed?" Also, goals that were not yet refined into features were listed. In the MDM case, the model was split into smaller parts for different functional areas. This resulted in 7 interconnected partial goal models.
4. Refine. In a second session with the domain experts, the missing concepts were recognized and added from the documentation. For example, we modelled a feature "Sending of meter stands collection card" as a prerequisite for all "Self Service" sub-goals.
5. Prioritize. Goals and features that were not yet (completely) realized, were prioritized using the Hierarchical Cumulative Voting method (Berander and Jönsson 2006). The results of the prioritization phase are presented in Table 1.
6. Propose. From the results of the prioritization session, a selection of features was proposed for inclusion in a future MECOMS™ release. This proposition was supported by information about the goals that could be achieved by implementing these features, and the HCV scores that were assigned by the stakeholders.
7. Decide. A decision about the scope of the next MECOMS™ release by the Change Advisory Board has not yet been made. This means the case study has not been fully completed yet. However, this last phase is only an endorsement by the CAB of our proposal, and does not add more information than what we already have.

The finalized MDM goal model consists of 37 goals, 4 obstacles and 29 features, modelled in 7 partial (interconnected) goal models. Figure 2 shows a fragment of the goal model, concerning Manual Meter Reading (MMR) functionality. Periodic meter readings can be read into the system by an interface with an external meter reading system, periodic and non-periodic tours, or through one of the several self-service methods. The self-service goal-refinement alternatives are the meter stand collection by phone, mail-in card, or company website. In order to achieve these goals, we need respectively an automated phone-answering system, OCR (Optical Character Recognition) software, or a self-service website. For all three alternatives, a prerequisite is that first a notification is sent to the customers: this is modelled as the feature "Sending of meter stands collection card".

These goal refinements provide alternative solutions for achieving the high-level "Periodic MMR" goal.

Table 1: Score of the MDM features as a result of the Hierarchical Cumulative Voting method.

Feature	Score
Interface with the phone answering system	6,70
OCR functionality for reading meter stands from card	4,54
Sending of meter stands collection card	20,96
Website for entering meter stands	19,66
Periodic tour based on the address	29,19
Non-periodic tour for on demand MMR	18,96

Some observations

The models created for the MECOMS™ products were much larger than anticipated. Still they only cover mostly high-level system goals, and all requirements have been grouped into features in order to keep the models to a reasonable size.

We also found that goal refinements are often a complex combination of AND and OR relations, and sometimes a XOR-like goal-refinement relation was needed but was unavailable or would heavily complicate the model. This resulted in workarounds and, in a few cases, omissions in the models.

We found that there are no suitable concepts for modelling maintenance goals in the current version of the KAOS modelling tool Objectiver™ (Respect-IT, 2007). We modelled these goals as regular (achievement) goals.

The criteria that were used to prioritize the goals and features in the MDM and CIS case studies were determined by the stakeholders. What we didn't expect, was that they opted to use different criteria for goals and for features. The reason for this is that features can be evaluated on a much more detailed level than goals. In our case, the goals were only subjected to the criterion "Ability to sell", while the features were evaluated on six different criteria.

5 CONCLUSIONS

As with any production process, software development requires support for project planning, monitoring the current state of the business product and the next steps of its development at the strategic level. The goal models of a product are instruments supporting the product assessment and planning the steps of its evolution.

However, the goal models of a real product are usually large. In order to be observable the goal models need to have a mechanism of labelling the goals with priorities and filtering the goals with high priority.

In this paper, we have presented a method for building the goal models labelled with criteria measuring priority. Our method includes an expert approach to selecting the labelling criteria related to business goals and collecting the values of the criteria attributed to each goal and associated features.

There are different methods for selecting labelling criteria. Therefore, one of the directions for future research would be goal prioritization based on conflicting criteria. Future work could also focus on the tool support for building the goal models labelled with priorities and filtering the goals having the highest priority. The procedures of assessing the values of the criteria attributed to each goal and filtering could be used periodically and become a valuable mechanism for management of software product-development teams.

ACKNOWLEDGEMENTS

We would like to express our gratitude to Prof. Dr. Rob Kusters at the Open University of the Netherlands for his comments, suggestions and support.

We thank Dr. Robert Darimont, CEO of Respect-IT S.A., for providing us with a research license and customized plugin for the Objectiver™ modelling tool.

REFERENCES

Anton, A. 1997. Goal Identification and Refinement in the Specification of Software-Based Information Systems. Ph.D. thesis, Department of Computer Science, Georgia Institute of Technology.

Berander, P., and Andrews, A., 2005. Requirements Prioritization. In *Engineering and Managing Software Requirements*, 69-94: Springer.

Berander, P. and P. Jönsson, 2006. Hierarchical cumulative voting (HCV) – prioritization of requirements in hierarchies. *International Journal of Software Engineering and Knowledge Engineering, 16(06): 819-849.*

Ferranti Computer Systems. MECOMS™ Product Overview. http://www.mecoms.com/product (accessed March 13, 2013).

Giorgini, P., Mylopoulos, J., Nicchiarelli, E. and Sebastiani, R., 2003. Reasoning with goal models. *Conceptual Modeling—ER 2002*, 167-181.

John, Isabel, 2006. Capturing Product Line Information from Legacy User Documentation. In *Software Product Lines*, edited by Timo Käköla and JuanCarlos Duenas, 127-159: Springer Berlin Heidelberg.

Lapouchnian, A., 2005. Goal-Oriented Requirements Engineering: An Overview of the Current Research.

Regev, G. and A. Wegmann, 2005. Where Do Goals Come From: The Underlying Principles of Goal-Oriented Requirements Engineering. *Proceedings of the 13th IEEE International Conference on Requirements Engineering,* 253-362: IEEE Computer Society.

Respect-IT, 2007. *A Kaos Tutorial*: Objectiver.

Ruhe, Günther, 2011. *Product Release Planning: Methods, Tools and Applications*: Auerbach Publications.

Gurp van, J., 2003. On the Design & Preservation of Software Systems. PhD dissertation, Computer Science Department, University of Groningen, Groningen.

van Lamsweerde, A., 2001. Goal-Oriented Requirements Engineering: A Guided Tour. In *IEEE International Conference on Requirements Engineering*, 249. Toronto, Canada.

Yu, E., 1999. Strategic Modelling for Enterprise Integration. In *Proceedings of the 14th World Congress of the International Federation of Automatic Control, July 5-9, 1999*, 5-9. Beijijng, China.

Yu, Y., Y. Wang, J. Mylopoulos, S. Liaskos, A. Lapouchnian, and J.C.S. do Prado Leite, 2005. Reverse Engineering Goal Models from Legacy Code. *Requirements Engineering, 2005. Proceedings. 13th IEEE International Conference on*, 363-372: IEEE Computer Society.

The Determination of Service Couplings for Non-disruptive Systems

Thunchira Thongmee, Hiroto Suzuki, Takahiro Ohno
Graduate School of Creative Science and Engineering, Waseda University, Tokyo, Japan
thunchira@moegi.waseda.jp, hiroto-suzuki@aoni.waseda.jp, ohno@waseda.jp

Udom Silparcha
School of Information Technology, King Mongkut's University of Technology Thonburi, Bangkok, Thailand
udom@sit.kmutt.ac.th

Keywords: Service Oriented Architecture, Degree of Coupling, Non-disruptive.

Abstract: Business continuity becomes an important feature for business these days. It is difficult to accept unplanned downtime due to the unavailability of the service, which mostly is unavoidable. The unavailable of the service is the situation such as service failure, unplanned downtime, etc. When such situation occurs, the system will need to look for some solution that will allow the system to continue. A simple but effective solution is to provide at least one alternative path of services that performs similar functions to the usual services which become unavailable. Therefore, switching between service paths is crucial in any reliable system. Service quality of a system or task can be determined with various factors. One of the most common factors is service coupling. This study will analyse the changes in couplings upon service switching, which in turn reflect the overall quality of the system.

1 INTRODUCTION

Service technology is widely adopted by business these days. One of the important features to determine the service quality is couplings among services or modules. Couplings refer to the dependencies between two or more services, which normally exist in software applications (Erl, 2008). There are loose couplings and tight couplings, which imply how much the modules are dependent on each other. Such levels of dependencies can be described as the "degree" of coupling between services, where a loose coupling contributes a lower degree of coupling than a tight coupling. The degree of coupling can be used to determine the availability, reliability, performance, usability, discoverability, and maintainability of the services within a system (Xu et al., 2006). The degree of coupling also identifies possible risks, and the estimated cost for development and maintenance (Yang et al., 2005). In general, the lower degree of coupling implies the better system design. Thus, there are many researches focus on the degree of coupling in order to be able to minimize the degree of coupling in their tasks.

In a service-oriented environment, any service interruption may cause some loss in business due to the business disruption. Therefore, for business continuity, a highly available system should consider providing at least one alternative service for each critical one (Wang et al., 2009). That means when a critical service becomes unavailable, there should be some other service that can provide the equivalent functionality. The unavailability of the service indicates the situations such as the service down, resource full (unable to allocated), or when the communication lost occurs. In this study, we discuss about the service switching between the critical service and the alternative services, by focusing on the analysis of its impact on the degree of coupling within the same task and among different tasks within a system. By analyzing such an impact, a system designer will have a more proper insight of how reliable the system truly is.

2 SERVICE COUPLINGS

There are a number of researches in the literature that have proposed techniques to determine the quality of a system design (Yang et al., 2005; Rich et al., 2012; Wang et al., 2009; Gui and Scott, 2006; Allen and Khoshgoftaar, 1999). Among the proposed techniques, system component coupling or dependency is one of the most widely adopted techniques (Yang et al., 2007). There are two kinds of dependencies among system components, inter-component and intra-component dependencies. The system components can be viewed at different levels. From the business point of view, a system component can be a single business process that interacts with other business processes. At the system architecture level, a system component is a service that provides specific functions for a business process. Such a service usually requires some interactions with other services. Finally, at the object-oriented implementation level, a system component is a class that works hand in hand with other classes. Therefore, the degree of coupling can be determined at different levels depending on the perspectives. In this paper, we focused at the system architecture level, where services are the major component of concern. How much a service is dependent on another service is described as the degree of coupling. The system with a higher degree of coupling, i.e. there are more dependencies among its services, is considered to be more complex than the system with lower degree of coupling.

There are a lot of researches in literature intended to measure the degree of coupling (Xu et al., 2006; Yang and Temporo, 2007; Wang et al., 2009; Gui and Scott, 2006; Allen and Khoshgoftaar, 1999; Saxena and Kumar, 2012). Our proposed concepts are applicable for any kind of couplings. However, for simplicity, we only concern with direct couplings (Yang et al., 2005). A direct coupling is a relationship, between two services. There are several kinds of such a relationship, including a method call, communication between services, etc. (Saxena and Kumar, 2012). The amount of dependency between two services can be obtained by the number of relationships between them. Therefore, the overall degree of coupling of a task or a system is the summation of the amounts of dependency of all possible pairs of services within its scope. The degree of coupling of a task is described in equation (1).

$$DC_{task} = \sum_{i,j=1}^{n} dp(i,j), i \neq j \qquad (1)$$

where

DC_{task} is the degree of coupling of the task,

$dp(i,j)$ is the amount of dependency services i and j,

n is the number of services within the task.

3 SERVICE SWITCHING

For system continuity, a reliable system should be designed such that every critical service should have some "backup" or alternative service that can provide similar functionalities. In case of uncertain circumstances may occur with a critical service also called the primary service becomes unavailable, some business process will not be able to continue or have a difficulty to continue its execution. In this case, the service path will have to switch from the primary path to the alternative path. In the other words, the prepared alternative service will be used in place of the primary service. Such uncertain circumstances may be due to service unavailable such as failures, data corruption or human errors (Beecher et al., 2011). In turns, the alternative service may also be required by some other services which are possibly parts of the same task and even worse if it is required by services of some other tasks. In addition, the primary path is usually the best choice in a good system design. Being forced to execute an alternative path to maintain its function may result in an inevitable increasing complexity of the system. We define service switching into two types which are switching services within a task, and switching services across tasks. Before we discuss in greater details of such types of service switching, there are some conventions to be noted as follows.

- Primary and alternative services can provide equivalent functionalities but they are not exact the same service.
- Primary and alternative services may have different degrees of coupling associated with them.
- Any service may be served by one or more services. To complete a task a number of services may be utilized in parallel and/or sequential orders.

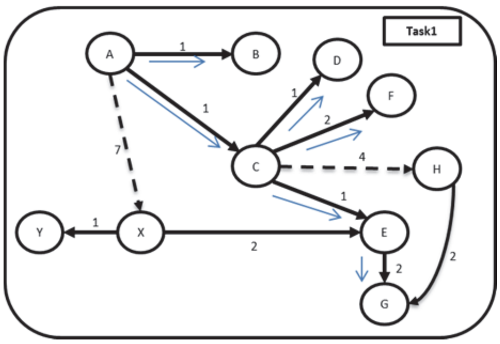


Figure 1: This primary path to complete Task1.

Table 1: The primary path to complete Task1.

Path	Degree of Coupling
A → B	1
A → C	1
C → D	1
C → E	1
C → F	2
E → G	2
Total Degree of Coupling	8

3.1 Switching Services within a Task

When a critical service needed for a task becomes unavailable, the task will switch to execute a predefined alternative service in order to complete the task. In this case, the service switching will affect only the specific task, and no others.

To demonstrate the service switching within a task, an example system is considered with an execution task, referred to as Taks1. The task has its primary path set to services A, B, C, D, E, F, and G. Services C and E are critical services with X and H being their alternative services, respectively. The overall service dependencies of Task1 are shown in Figure 1.

The overall degree of coupling of the primary path of Task1 is calculated as shown in Table 1.

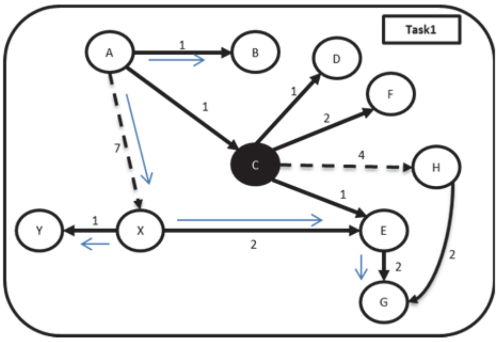


Figure 2: Task1 after service switching occurs.

Table 2: The alternative path to complete Task1.

Path	Degree of Coupling
A → B	1
A → X	7
X → E	2
X → Y	1
E → G	2
Total Degree of Coupling	13

The primary path's degree of coupling of Task1 is 1+1+1+1+2+2 = 8. We can assume for simplicity that the execution order of these services is not significant. Once service C becomes unavailable as shown in Figure 2, Service A has to execute service X in order to continue its work for Task1. By switching from service C to service X the degree of coupling has become 1+7+2+1+2 = 13 as shown in Table 2.

3.2 Switching Services across Tasks

It is quite common that different task may require same services. Switching services across tasks happens when a service component, that serves a usual task, is occupied by a different task from the usual one due to some circumstance that makes such a switching occurred. The result of such service switching can then cause a chain reaction on another task that usually utilizes the service. This phenomenal is demonstrated in Figure 3.

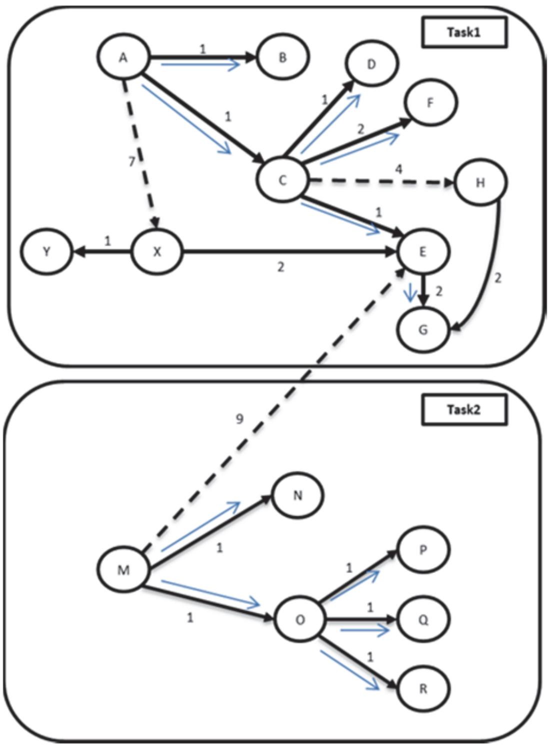


Figure 3: Service path of Task1 and Task2.

In this system, there are two separated tasks running simultaneously, i.e. Task1 and Task2. Task1 is similar to the one described in the previous discussion, such that it requires services A, B, C, D, E, F, and G with the degree of coupling 8 for its primary path. Task2 consists of services M, N, O, P, Q, and R. Task1's service E presents as an alternative of Task 2's service N. The degree of coupling of the primary path of Task2 is 1+1+1+1+1

= 5. In a normal situation, each task executes its service according to its primary path. Once, for some reason, service N becomes unavailable. In this case, the service switching occurs. Service N will be switched to its alternative which is service E, which depends on service G. Task1's degree of coupling has now increased from 8 to 13 as discussed earlier. Task2, at the same time, has increased its overall degree of coupling from 5 to 16 due to the switching across tasks. Since both Task1 and Task2 belong to the same system, the total degree of coupling of this system has increased from 8+5 = 13 to 13+16 = 29. The increased degree of coupling of a system implies that it becomes more vulnerable.

3.3 Determining the System True Degree of Coupling

In the two sections above, we discussed about the couplings within and across tasks. When a task within the system needs to switch from a primary service to the alternative services, the degree of coupling usually increases. That is because when we design software, we usually select the best path, i.e. the path that has the lowest overall degree of coupling, to be our primary path, and some other path with higher degree of coupling to be an alternative one. Therefore, once a service switching occurs, the degree of coupling usually increases.

Since the degree of coupling can be used for software quality measurement, considering only the degree of coupling of the primary path may no longer be accurate because it is not guaranteed that the primary path will always be available, i.e. a service switching occurs and the degree of coupling will change. On the other hand, considering only alternative path is equally unfair. Thus, the system designer should take account for the degrees of coupling in both primary and all alternative paths. Equations (2) - (4) suggest the proper degree of coupling of a system using the probability as the weight.

$$DC_{system} = \sum_{p=1}^{m} DC_{task_p} \quad (2)$$

$$DC_{task_p} = P(M_p)C(M_p) + C(A_p) \quad (3)$$

$$C(A_p) = \sum_{x=1}^{y} [P(alt_{px})C(alt_{px})] \quad (4)$$

,where

DC_{system} is the degree of coupling of the system,

DC_{task_p} is the direct coupling of $task_p$ within the system,

$P(M_p)$ is the probability that the primary path will be used in task p,

$C(M_p)$ is the total dependency along the primary path,

$C(A_p)$ is the total degree of coupling along the alternative path,

$P(alt_{px})$ is the probability that the alternative path x will be used in task p,

$C(alt_{px})$ is the total dependency along the alternative path x.

For a simplest case, a system consists of two separated tasks; say T_a and T_b, each of which relies on its main service path. Suppose the degrees of coupling primary paths of T_a and T_b are 8 and 11, respectively. An alternative path is provided for each main task, such that the alternative path of T_a has 14 degree of coupling, which the one for T_b has 17 degree of coupling. The probability that the main service path in T_a to become unavailable is 5%, and 10% for T_b. Then the overall degree of coupling of this system will be calculated as shown below. Then $P(M_a)$ = 0.95, $P(M_b)$ = 0.90, $P(A_a)$ = 0.05, $P(A_b)$ = 0.10, $C(M_a)$ = 8, $C(M_b)$ = 11, $C(A_a)$ = 14, and $C(A_b)$ = 17.

$$\begin{aligned} DC_a &= P(M_a)C(M_a) + P(A_a)C(A_a) \\ &= (0.95)(8) + (0.05)(14) \\ &= 8.3 \end{aligned}$$

$$\begin{aligned} DC_b &= P(M_b)C(M_b) + P(A_b)C(A_b) \\ &= (0.90)(11) + (0.1)(17) \\ &= 11.6 \end{aligned}$$

$$\text{Thus,} \quad \begin{aligned} DC_{System} &= DC_a + DC_b \\ &= 8.3 + 11.6 \\ &= 19.9 \end{aligned}$$

From above example, we consider the degree of coupling from both primary and alternative paths.

4 CONCLUSIONS

This study proposes a technique that can be used to determine the quality of software systems through the overall degree of couplings, particularly for non-disruptive systems. The system continuity can be achieved by providing alternative service paths to critical ones, so that the system may switch the execution to the alternative services once the main services become unavailable.

It is quite natural that the degree of couplings of the alternative path is higher than the main path because of the system design that should minimize the degree of coupling. Since switching the service paths may increase the overall degree of coupling of a system, it is essential that the system designer should look into this issue when determines the true quality of the system.

REFERENCES

Allen, E., Khoshgoftaar, T., 1999. Measuring coupling and cohesion: an information – theory approach. The 6th International Symposium on Software Metrics.

Beecher V. et al., 2011. Oracle database high availability overview 11g R2. Oracle Database Documentation Library.

Erl, T., 2008. SOA: principles of service design. Prentice Hall.

Gui, G., Scott, P., 2006. Coupling and cohesion measures for evaluation of component reusability. The 2006 International on Mining Software Repositories.

IBM Global Service, 2012. IBM index reveals key indicators of business continuity exposure and maturity. IBM Global Technology Services.

Rich, K. et al., 2012. Oracle data guard concepts and administration 11g R2. Oracle Database Document Library.

Saxena, V., Kumar, S., 2012. Impact of coupling and cohesion in object-oriented technology. Journal of Software Engineering and Application.

Wang, J. et al., 2009. Service evaluation in SOA: Towards business/IT alignment. The 10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking, and Parallel/Distributed Computing.

Xu, T., et al., 2006. Service oriented dynamic decoupling metrics. The 2006 International Conference on Semantic Web and Web Service.

Yang, H. et al., 2005. Detecting indirect coupling. The 2005 Australian Software Engineering Conference.

Yang, H., Temporo, E., 2007. Measuring the strength of indirect coupling. The 2007 Australian Software Engineering Conference.

Managing Business Model Objectives through Platform Strategies
A Case Study of the Google Android Ecosystem

Karthik Jayaraman
Department of informatics, University of Oslo, Norway
karthikj@ifi.uio.no

Keywords: Software Ecosystems, Competing Values Framework, Android, Information Infrastructures, Platforms.

Abstract: The goal of this research is to study how organizations achieve and balance their conflicting organizational objectives with the help of dynamic platform strategies. This is done by analyzing the Android platform where the translation of the organizational objectives of the platform controller Google into its platform strategies is examined through a series of cases. The analysis is done through the lens of the competing values framework where changing organizational goals of platform controllers are mapped and understood through the enactment of their platform strategies.

1 INTRODUCTION

Cross disciplinary research in the fields of evolutionary psychology, biology and neuroscience has led to the understanding that people are driven by four biologically determined needs that can be used to describe all human behavior. These are the drives to bond, to learn, to acquire, and to defend (Lawrence and Nohria 2002). Like human beings, an organizations business model should propogate objectives such as creating new innovation, facilitating collaboration, controlling the evolution of a product or platform and compete successfully in the industry (Quinn et al 2010). These values are not at harmony often and can be conflicting most of the time. The dominance of certain values over others and the conflict in values are determined by organizational goals and various forces in the industry (Porter 1979).

Companies like Google that play the role of a platform controller have to constantly adapt their organizational values to survive in the market place. Android is among the fastest evolving platform and is competing with other platforms to become the dominant design. Innovativeness of the platform has led to the attraction of an early install base, which is critical to the success of these platforms. The organizational value creation through innovation is a key value to drive growth. To become the dominant design and increase the pace of innovation companies have transitioned their strategy from closed products to platform centric ecosystems. This model of dealing with complexity and accelerating innovation by building an ecosystem is achieved through the process of open innovation; a strategy where firms use external as well as internal ideas and internal and external paths to market, as they look to advance their technology (Chesbrough, 2006).

Once an organization decides to make its platform available to entities outside its boundary, it creates a software ecosystem (Bosch 2009). Due to the global diaspora of knowledge workers, the knowledge and skill sets required to create new innovation is often not present within the firm's boundaries; hence firms leverage the global talent pool for exploration activities through open innovation. Hence innovation in mobile information infrastructure ecologies such as Android is no longer the sole responsibility of the platform owners but a shared responsibility of various stakeholders such as app developers, handset manufacturers, content providers and mobile carriers. The design and evolution of these information infrastructure ecologies impact both the platform owners and the various stakeholders who are part of the ecology. Hence the organizational value of collobration by managing stakeholder interests is one of the key determinants of the success of these platforms (Selander, Henfridsson and Svahn 2010).The concept of generativity is the system's capacity to produce unanticipated change through unfiltered contributions from broad and varied audiences

(Zittrain 2008). The generative ability of mobile platform’s such as Android to create new functionality through interfaces such as API's is a lead determinant of the platform becoming the leader (Koski & Kretschmer 2007) by attracting an early installed base. Using generativity to drive the organizational value of competition can assist in platforms becoming a dominant design.

The goal of this paper is to analyze how platform controllers like Google acheive their organizational objectives to innovate, control, compete and collaborate through their platform strategies and how they manages strategic tensions between platform creation and control, and simultaneous collaboration and competition with various stakeholders in the ecosystem. Four cases each representing an organizational objective is analyzed with the help of competing values framework and information infrastructure theory and various generalizations are drawn from the discussions. This research aims to contribute to the existing field of organizational strategy, information infrastructures, platforms and ecosystems research.

2 THEORETICAL PERSPECTIVE

The invention and rapid adoption of large complex systems such as mobile platforms and the Internet poses several challenges, which can be effectively addressed by learning from existing well established large scale infrastructures such as railroads and highways. Information infrastructures are complex systems that are shared, continuously evolving, open for interconnections, based on standards and heterogeneous installed bases (Hanseth and Monteiro 1998). The study of Information Infrastructures promotes the understanding of the design and interaction of manyInformation Systems and components that interact with each other to produce a functioning infrastructural backbone.

Mobile platforms such as the iOS and Android exhibit the traits of information infrastructures and are a sum of their parts as they contain various separate systems such as the kernel, modules, interfaces and apps and hence requires a holistic perspective for analysis. The control of the evolution of Information Infrastructures is often distributed and negotiated due to their complexity, hence a key challenges in the design of information infrastructures is in dealing with negotiations between control and generativity in the evolution of the Information Infrastructure. The concept of generativity is the system’s capacity to produce unanticipated change through unfiltered contributions from broad and varied audiences (Zittrain 2008). Some of the key drivers of generativity are leverage, adaptability, ease of mastery, accessibility and transferability. Leverage of a generative system indicates how extensively a system leverages a set of possible tasks to create value for the user.

The greater the functionality of a system the greater is its ability to produce change and be generative. Adaptability of a generative system is determined by how easy it is to extend or modify the system to broaden its use. Leverage and Adaptability are closely linked in the case of mobile information infrastructures. Ease of mastery determines the easiness for different types of users to understand, adopt and adapt the technology. The concept of accessibility is determined by ease of access to a technology. Barriers to accessibility are factors such as expense of producing and hence consuming the technology, taxes, regulations associated with its adoption or use, and the secrecy its producers adopt to maintain scarcity or control (Zittran 2008). Transferability indicates the easiness with which the changes in the technology can be communicated. An information infrastructure is considered fully transferable if the adaptation of the technology by highly skilled users can be easily communicated to a user with lesser skills and know-how of the technology.

2.1 Competing Values Framework

The competing values framework (CVF) is a strategic analysis framework that is useful in understanding organizational strategy and its effectiveness (Quinn et al 2010). The framework also assists in the recognition of guidelines that can assist in the management of relationships, congruencies, and contradictions among the various aspects of organizations (Quinn & Cameron 1983). Organizational strategy with the help of CVF can be analyzed through four quadrants each denoting a value of the firm. CVF describes that organizations are structured around two basic opposing needs; the need for flexibility and autonomy versus the need for control and stability; and the focus on internal concerns versus responsiveness to the external environment. These values represent competing assumptions of the firms beliefs and strategy. Being successful in the collaborate quadrant entails creating and sustaining commitment and cohesion. Collaboration deals with open communication, which entails a deep understanding of the concerns of various stakeholders. Collaborators perform both exploration and exploitation activities with stakeholders within and outside the boundaries of the firm. Collaborators manage intra and inter organizational conflict and promote innovation (Quinn et al 2010). The community created through colloboration shares beliefs, competencies, vision and values. Being successful in the control quadrant entails establishing and maintaining stability and continuity. The management of the control quadrant deals with compliance of rules and regulations. Control quadrant ensures performance, efficiency and effectiveness. Compete quadrant deal with the compete actions of the organization. This deals with improving productivity and profitability of the organization.

The understanding of the external environment is crucial for planning, goal setting and designing work processes in this quadrant. The strategies are driven by aggressive competition, markets changes, profits and speed. Stakeholders in this quadrant must constantly manage performance through objectives and use iterative mechanisms to quickly initiate or cease initiatives. The goals of the create quadrant are to rapidly create the necessary innovation, adapt to change and acquire the necessary support. The key skills required in this quadrant are identifying trends, differentiating from the competitor, encouraging new ways of thinking, starting new ventures, extrapolating emerging opportunities and promoting innovation. The quadrants of the competing values framework represent tensions that organizations face in creating and managing their strategies. Organizations possess varying degrees of the values represented in the quadrants.

3 METHOD

This research is based on a detailed case study of the Google Android platform. The study of how platform controllers achieve their organizational objectives through their platform strategies and manage strategic tensions in the ecosystems requires a case that provides different perspectives on the mentioned aspects of study to compare and contrast the various values in the competing values framework. The organizational and platform strategy processes of Google and their impact on the ecosystem can be seen as extreme cases (Yin 2009). The studied cases are paradigmatic of some phenomenon of interest (Gerring 2007). To highlight the challenges involved in managing platform strategies, a case study approach of (Gerring 2007; Yin 2009) was followed with a specific focus of studying organizational and platform strategies of Google and the Android ecosystem.

3.1 Data Collection

The data for the case studies in this research is based on various sources such as documents, website interviews and nonparticipant observations in communities. Secondary data available on the internet is a source of diverse, abundant and rich data material that exceeds the diversity of data that can be collected from direct interviews (Romano et al 2003; Ghazawneh and Henfridsson 2011).The data that were collected represents official press releases from Google related to Android, relevant messages from Android mailing lists that describe issues related to platform strategies and generative mechanisms were identified.

The collected data described various issues related to organizational values and platform strategies and how the stakeholders reacted to the implementation of the various strategies. The study of the field of Platform based ecosystems requires large volumes of data to understand the relationship between the different actors, their actions and the motivations for their actions (Ghazawneh and Henfridsson 2011).The secondary data for this research is representative of the period between January 2009 and January 2013, where some of the key issues described in this research became a cause of major concern with the growth and proliferation of the Android ecosystem. Data from multiple sources can assist in generating various generalizations and help in improving data quality (Soy 1997).

3.2 Data Analysis

This collected data was then analyzed with the help of Romano et al's (2003) methodology for analyzing web based qualitative data. The Romano et al' method is based on a three-step approach to data collection and analysis of Internet-based qualitative data, namely: elicitation, reduction and visualization. In the first step, elicitation, specific terms that are of interest to this research from the framework such as "Android collaboration", "Android innovation", "Android ecosystem control" were fed into popular search engines such as Google and Bing and specialized online search engines such as board tracker and omgili that track discussions in mailing lists and the resulting data was captured in a QDA tool. Online observations on the evolution of the

Android is aimed at collecting relevant data from natural settings, hence non-participant observations in mailing lists and discussion boards discussing Google's Android strategy provided useful data for this research. The elicited data was then saved in a Qualitative Data Analysis (QDA) tool. Once the data was stored in the tool, some of the key themes in the data were identified. This was done with the help of word frequency analysis and a thorough literature review. Word frequency analysis provides a complete list of all the words that occur in the collected data material and the number of times they appear in the text. The analysis of the collected data through word frequency counts helped in deriving inferences about the subjects of importance. The result of the word frequency analysis and literature review was the discovery of some of the key themes in the data such as"licensing", "open handset alliance" and "release cycles".

An initial set of codes were then created to structure the data. The elicitation process led to the creation of a large research data set, some of the key concepts relevant to this research were identified such as platform control, agility in release intervals, open handset alliance. These concepts were then investigated in detail.In the second step, reduction, the large data set that was built during the elicitation process was reduced to fit the identified themes. Some of the key concepts identified in the elicitation step such as "licensing", "release process", "app blocking" and "fragmentation" were further investigated. The reduced data was further coded to identify the sequence of events and actors based on the themes that were discovered and evolved from the literature review and investigation of the initial data set. In the visualization step the various organizational objectives and their acheivement through platform strategies and information infrastructure mechanisms are identified and summarized the discussion section. The strategies are summarized as four cases. The visualization process also led to the identification of the key factors that drove the organizational strategy in a mobile software ecosystem, which were further expanded upon and analyzed in section 5 of this research.

4 FINDINGS

From the time of release of first Android based phone to today, Google has been able to rapidly scale its platform, user base and other ecosystem components. One of the key drivers of Google's growth is its ability to manage its strategic initiatives around innovation, collaboration, quality management and using the platform and its ecosystem components as a way to compete in the market place. In the below section, four key cases from the collected data representing the above mentioned strategies are described.

Case I: Platform licensing decisions

Google adopted the Apache licensing scheme during the release of the Android platform to create provide access to the code base for a wide variety of audience. Andy Rubin, the head of the Android project at Google describes the reason for adopting the apache license as

"We built Android to be an open source mobile platform freely available to anyone wishing to use it. In 2008, Android was released under the Apache open source license and we continue to develop and innovate the platform under the same open source license -- it is available to everyone at: http://source.android.com. This openness allows device manufacturers to customize Android and enable new user experiences, driving innovation and consumer choice."

Case II: Design of the open handset alliance

The open handset alliance is a networked model of collaborative innovation where Google engages with various stakeholders in the ecosystem. The open handset alliance assists stakeholders such as device manufacturers, content providers, semiconductor companies and operators in adopting the Android platform. Andy Rubin the head of the Android project at Google commented

"Despite all of the very interesting speculation over the last few months, we're not announcing a Gphone. However, we think what we are announcing -- the Open Handset Alliance and Android is more significant and ambitious than a single phone. In fact, through the joint efforts of the members of the Open Handset Alliance, we hope Android will be the foundation for many new phones and will create an entirely new mobile experience for users, with new applications and new capabilities we can't imagine today."

Case III: Managing fragmentation

The flexible licensing scheme of Android allowed for the platform to be modified in many different ways and did not require the modifications to be contributed back to the creators of the platform. This led to the forking of the project into various derivatives. Chris Roland a commentator on Android describes the issue of fragmentation as

"Fragmentation of a operating system is nothing new one need only look at what has happened with Linux to see a great example of how this can occur. Fedora and Ubuntu are both Linux, they both have a Linux kernel, and are for the most part compatible with each other, but not entirely. Applications designed for one will not easily run on the other unless recompiled. Android is the same, but worse. Not only do developers have to contend with different versions of Android between 2.x, 3.x and now 4.x, but not all Androids of the same major version are the same developers have to contend with OEM customisations and issues as well.".

Case IV: Platform release processes & leveraging ecosystem components

The Android team uses an agile way of product development release. Since its beta release in November 2007, Google has launched over thirty versions of the Android OS. Andy Rubin the head of the Android project describes the agile release strategy as a way to compete in the market

"We were at a feverish place post 1.0. 1.0 felt to me more like an 0.8 - it was pushed out for Christmas. We subsequently got it up to the spec that the industry expected it to be. We saw a rapid release cycle to basically catch up with the industry, and now I feel pretty much caught up. So any new releases aren't going to be catch-up releases, they're going to be releases that are focused on innovation. "

5 DISCUSSION

Based on the findings in the previous section, it can be observed that Google tries to balance conflicting values and organizational interests during the evolution of the Android platform. In the create quadrant one of the key challenges is to facilitate the necessary innovation and adapt to changes in the industry. Organizations have to cultivate strategies that facilitate flow of innovation and new ideas that can help the organizations innovate.

For the successful evolution of a platform, it must balance the introduction of new artifacts, processes, and actors but in the same time offering the flexibility to support scaling and further evolution of the platform (Tilson, Sorensen and Lyytinen 2011). As seen in case 1 in the findings section, the open licensing adopted by Google for its Android platform enables users access to the code base enabling them to modify and change the platform in new innovate ways, it also allowed commercial use of the platform as the licensing scheme allowed for the modification of the platform without having to commit the changes back to the platform controllers. One of the challenges in the create quadrant is the bootstrap problem. A platform's value is realized when a large number of users use a platform, hence platform controllers have to find ways to attract early users to use the platform. This can be difficult as platform controllers have to often address the needs of these early users before having a complete design of their platform (Hanseth and lyytenen 2010). The flexible licensing, killer apps and tools such as the SDK allowed for early users to understand the technical architecture and implement the necessary changes to adopt the platform and make it useful. As seen in case 3 in the findings section, the challenges of the control quadrant have to do with the quality aspects of the strategy. Complimenters of a platform need to be governed in a way where both platform controllers and complimenters create and extract value from the platform. The openness aspects of the create quadrant helped bootstrap the platform by attracting early users but also lead to the forking and fragmentation of the platform and its ecosystem. End users and competing organizations fork the platform and thereby split the platform resources and its ecosystem. To control the platform effectively, the platform owner constructs new platform designs, secures platform control through agreement , increases knowledge heterogeneity through distribution channels, and counteracts foreign boundary resources designed to infringe on the platform (Ghazawneh and Henfridsson 2010). Forks of the Android platform do not receive the latest updates of the platform released by Google, which might lead to critical security and quality issues.Due to the emergence of dozens of forked versions of the Android platform, the creation of an additional clause in the Android SDK's terms of service enabled Google to control and act on third parties that fork the platform.Due to the phenomenon of increasing returns, the more a standard is diffused, the greater its value.

A platform standard can also be used as a strategic tool to decide whether a particular product is compatible with competitor's products (Katz and Shapiro1995). Forking can create incompatible versions and could hamper the long term sustainability and quality of the platform and its ecosystem (Krogh and Spaeth 2007). Google created the Android compatibility definition document , which details the software and hardware requirements to comply to the specifications of a compatible Android device.

Once a complimentor or a competitor creates an Android compatible product they will have to use the compatibility test suite as an aid to compatibility during the development process.Organizations such as Google have to thus balance their quest for platform growth with enforcing quality and standardization in the ecosystem. Companies that are part of an industry segment are encapsulated in an ecosystem of suppliers, customers, partners, competitors, suppliers of substitute products or services and potential new entrants that can challenge the status quo (Popp and Meyer 2010). Hence colloborating with various stakeholders is a key organizational value that determines the success of a platform controller. The values of the collaborate quadrant are determined by actions such as fostering collective effort, building cohesion and teamwork and managing conflicts in the ecosystem.
The collaboration is driven by a shared aims, values and expertise. One of the key challenges of a mobile platform is the challenge of install base cultivation. A rapid establishment of a large installed base can help companies become the platform standard of choice (Besen and Farrell 1994). Rival firms seek to exceed consumers' expectations as the size of the installed base determines the choice of platform standard, hence industries with network effects wittiness an intense early competition as a quick early lead can determine the outcome of the race and the ones that follow the platform leader eventually end up becoming obsolete.

As seen in case 2 in the findings section, Google created a collaborative model of innovation through the open handset alliance, where complimentors in the ecosystem could work with Google in shaping the Android platform. The networked model of innovation enabled various stakeholders such as device manufacturers, content creators and others to provide inputs to the innovation process and build a platform for shared vision and innovation, which assists in the deployment of Android on various devices contributing to the build up of a rapid install base. Stakeholders could customize new releases of the platform to suite their needs and colloborating in the design and development process enables an easier adoption of the platform and assists in planning ahead.When information infrastructures such as Android grow in the creation phase due to bootstrapping and aspects such as openness, the platform controllers have to deal with unforeseen and diverse demands and manage these demands in the design and technical architecture of the platform. This is defined as the adaptability problem of information infrastructures (Hanseth and lyytenen 2010). The open handset alliance acts as a forum to evolve complex relationships with the various stakeholders, who are some times direct competitors. The adaptations that a platform undergoes through a series of negotiations in forums like the open handset alliance are determined by socio-technical motivations and can be understood by analyzing the significant changes that a platform undergoes, which are often infrequent, discontinuous, and intentional (Tilson, Sorensen and Lyytinen 2011). Keystones or platform controllers like Google have to balance two key activities to be successful, the first is to create value within the ecosystem and the second is to share the value with other participants in the ecosystem. Unless a keystone finds a way of doing this efficiently, it will fail to attract or retain members (Iyer et al 2006). Hence the open handset alliance helps cultivate install base by acting as a forum for creation of new innovation and facilitation of Android compatability and drives platform standardization through ensuring a consistent experience for developers, manufacturers and consumers.

The core competence required by an organization in the compete quadrant is product differentiation through rapid innovation, agility and lockin mechanisms. Innovation in a platform ecosystem is characterized by simultaneous competition and cooperation (Walley 2007), where the relationships between the members are actively shaped by the stakeholders. In such ecosystems, the competitive position of a firm participating in the ecosystem is measured through its relationships to other participants in the ecosystem (Selander, Henfridsson and Svahn 2010). As seen in case 4 in the findings section, while Google and Apple directly compete against each other through their platforms, they colloborate and leverage each others platform for their ecosystem components. Apple uses Google maps as its maps and navigation system but Google released a version of its maps app with diminished capabilities for the iOS version thereby creating a locking in end users to the Android platform.

The existence of strong network effects and increasing returns promotes a single or a small set of platform standards being adopted in an industry (Bekkers and Martinelli 2010). Hence agility through short release cycles to gain market share by a rapid buildup of install base is key to becoming the dominant design. Google uses short release intervals to introduce new innovations in the platform, while most of these releases are backward compatiable some versions are not and this leads to fragmentation of the platform. Hence balancing competing organizational values of colloboration which is enacted through sharing and networked innovation with the organizational value of competition which is enabled through agilie innovation and lockin's is crucial for platform controllers in becoming the dominant design.

6 CONCLUSIONS

The goal of this research was to analyze how organizations such as Google achieve their objectives through the enactment of platform strategies and manage strategic tensions in the ecosystem. Based on the discussions in the previous section, it is evident that in order to deal with the changes in the industry and counter market forces, organizations such as Google have to balance radical new innovation with incremantal sustaining innovation through managing change that is new, innovative, unique, and transformational with small incremental change that drives efficiency, predictability, and continuity through their open innovation strategy. This can help platform controllers become the dominant design, but to sustain their posistion, platform controllers have to also balance their short term need for speed and agility with a long term focus on developing their ecosystem by controlling the pace of innovation.

Hence being dynamic and balancing the various organizational values over time can yield desirable results. The strength of the platform lies in the diversity of the stakeholders participating in its ecosystem, when engaging in an ecosystem the organizational value collaboration takes greater importance. Platform controllers have to constantly shape their relationship with the various stakeholders to be successful and shape it in a way that everyone in the ecosystem benefits from the platform.More research is needed to corroborate many of the findings in this research. One of the caveats is that the analysis is restricted to the Andorid ecosystem.

While this restriction has assisted in improving our understanding of how platform controllers deal with competing organizational values through their platform strategies, we would require more cases where the findings can be tested. A task for future research would be to conduct longitudinal studies on various mobile and non-mobile software platforms and analyze platform controllers balance their organizational values over the lifecycle of a platform.

REFERENCES

Baldwin, C.Y., Clark, K., Design Rules: The Power of Modularity. Cambridge, MA: MIT Press.

Besen, SM., Farrell, J. (2007). "Choosing How to Compete: Strategies and Tactics in Standardization." Journal of Economic Perspectives, 8(2): 117–131. Walley, K., Coopetition: An Introduction to International Studies of Management and Organization, 37(2), 11-31.

Bekkers, R., Martinelli, A. (2010). The interplay between standardization and technological change: A study on wireless technologies, technological trajectories, and essential patent claims, No 10-08, Eindhoven Center for Innovation Studies (ECIS) working paper series.

Bosch, J. (2009). From software product lines to software ecosystems. In Proceedings of the 13th International Conference on Software Product Lines (SPLC).

Chesbrough, H., Vanhaverbeke, W., West,J. (2006). Open Innovation: Researching a New Paradigm. Oxford: Oxford University Press, pp. 1-12.

Clark, D.D. (1988). The design philosophy of the DARPA Internet protocols. Computer Communications Review, 18 (4): 106–114..

Ghazawneh, A., Henfridsson, O. (2011). "Micro-Strategizing in Platform Ecosystems: A Multiple Case Study" . ICIS 2011 Proceedings. Paper 3.

Gerring, J. (2007). Case study research: Principles and practices. Cambridge, UK: Cambridge University Press.

Hanseth, O., Monteiro, E., (1997), Understanding Information Infrastructures. Manuscript: http://www.ifi.uio.no/~oleha/Publications/bok.pdf, Accessed May 5, 2012.

Hanseth, O,.Lyytinen, K. (2010). Design theory for dynamic complexity in information infrastructures: the case of building internet . JIT. Journal of information technology (Print). ISSN 0268-3962.

Henfridsson, O. (2009). Mathiassen, L., Svahn, F., "Reconfiguring Modularity: Closing Capability Gaps in Digital Innovation," Viktoria Institute, Sweden . Sprouts: Working Papers on Information Systems, 9(22).

Iyer, B., Lee, C.-H. & Venkatraman N. (2006). Managing in a "Small World Ecosystem": Some Lessons from the Software Sector. California Mgmt. Review, 48(3): p. 28—47.

Katz, M., Shapiro, C. (1995).Network Externalities, Competition and Compatibility. American Economic Review.

Koski, H, Kretschmer, T. (2007). "Innovation and Dominant Design in Mobile Telephony," Industry &

Innovation, Taylor and Francis Journals, vol. 14(3), pages 305-324.
Krogh, G.V., Spaeth, S. (2007). The open source software phenomenon: Characteristics that promote research, The Journal of Strategic Information Systems, Volume 16, Issue 3, Pages 236-253.
Lawrence, P. R., & Nohria, N. Driven. (2002). How Human Nature Shapes Our Choices, San Fransisco: Jossey Bass.
Porter, M.E. (1979.)"How Competitive Forces Shape Strategy", Harvard Business Review
Popp, K.,M., Meyer, R. (2010). Profit from Software Ecosystems: Business Models, Ecosystems and Partnerships in the Software Industry. Norderstedt, Germany: BOD. ISBN 3-8391-6983-6.
Quinn, R., Cameron, K. (1983). Organizational life cycles and shifting criteria of effectiveness: Some preliminary evidence. Management Science, 29, 33-41.
Quinn, R.E., Faerman, S.R., Thompson, M.P., McGrath, M.,St. Clair, L.S., (2010). Becoming a Master Manager: A Competing Values Approach, 5th Edition, Wiley, ISBN 978-0-470-28466-7,
Romano, N. C., Donovan, C., Chen, H., & Nunamaker, J. F. (2003). "A methodology for analyzing web-based qualitative
data," Journal of Management Information Systems, (19:4), pp. 213-246..
Sambamurthy, V., A. Bharadwaj, V. Grover. (2003). Reconceptualizing the role of information technology in contemporary firms. MIS Quart. 27(2),237–264.
Selander, L., Henfridsson, O., Svahn, F. (2010). " Transforming Ecosystem Relationships in Digital Innovation ". ICIS 2010 Proceedings. Paper 138.
Soy, S. K. (1997) "The case study as a research method,"University of Texas, Graduate School of Library
http://www.gslis.utexas.edu/~ssoy/usesusers/l391d1b.htm [Accessed on 1 Oct 2012].
Tilson, D., Sørensen, C., Lyytinen, K. (2011). The Paradoxes of Change and Control in Digital Infrastructures: The Mobile Operating Systems Case. ICMB 2011: 26-35.
Yin, R.K. (2009). Case Study Research: Design and Methods, Thousand Oaks: SAGE Publications.
Yoo, Y., Henfridsson, O., Lyytinen,K. (2010). Research Commentary---The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research. Info. Sys. Research 21, 4 (December 2010), 724-735.
Zittrain, J. (2008). The Future of the Internet - And How to Stop It., Yale University Press, Penguin UK/Allen Lane; Oxford Legal Studies Research Paper No. 36.

Soft Reservations
Uncertainty-aware Resource Reservations in IaaS Environments

Seyed Vahid Mohammadi[1], Samuel Kounev[2], Adrian Juan-Verdejo [1] and Bholanathsingh Surajbali[1]
[1]*CAS Software A.G, CAS-Weg 1-5, Karlsruhe, Germany*
[2]*Institute for Program Structures and Data Organization, Karlsruhe Institute of Technology, Karlsruhe, Germany*
vahid.mohammadi@cas.de, kounev@kit.edu, {adrian.juan, b.surajbali}@cas.de

Keywords: Cloud Computing, IaaS, Resource Reservations.

Abstract: Modern Infrastructure-as-a-Service (IaaS) provides flexible access to data center resources on demand in an elastic fashion to meet the highly variable workload requirements of cloud applications. Cloud providers aim to provision resources as efficiently and as quickly as possible to their consumers. However, the lack of information about the hosted applications and their workloads makes it hard for cloud providers to anticipate the future resource demands of their customers so that they can plan the capacity of their infrastructure. Cloud providers can receive arbitrary requests for allocating resources on-the-fly in a completely unpredictable manner. Given this unpredictability, it may happen that providers might not be able to provision the requested resources quickly enough, or in the worst case, they might ran out of capacity and may not be able to satisfy their customers resource demands. To address these concerns, in this paper we propose a new resource reservation mechanism, based on the concept of soft reservations, addressing the issue of uncertainty and lack of information concerning the expected future customer workloads and corresponding resource demands. The proposed resource reservation mechanism makes it possible for cloud providers to better plan the capacity of their infrastructure and continuously optimize the placement of virtual machines on physical nodes thus improving the infrastructure cost and energy efficiency. It also takes into account the uncertainty of resource demand estimations and enables proactive online capacity planning resulting in cost benefits for both cloud providers and cloud customers.

1 INTRODUCTION

Cloud Computing is emerging as a new computing paradigm providing cloud consumers (henceforth called consumers) with on-demand access to data center resources by integrating computing, storage and networking platforms in a transparent manner. One of the major factors for the success of cloud computing is its elasticity property and the pay-per-use pricing strategy.

Elasticity is one of the major essential properties of the cloud paradigm providing the ability to deal with load variations by automatically provisioning / deprovisioning resources on-the-fly to match the current demand, i.e., adding more resources during high load periods and consolidating the resources to fewer nodes when the load decreases.

Ideally, this implies that the amount of resources such as CPU, main memory and network bandwidth are assigned and utilized in an optimal manner. For a system deployed in a pay-as-you-go cloud environment, such as Infrastructure-as-a-Service (IaaS), elasticity is critical to minimize operating cost while ensuring acceptable performance during high load periods. It allows consolidation of the system to consume less resource and thus minimize the operating costs during periods of low load while allowing it to dynamically scale up as the load increases.

This elastic scaling is typically implemented using virtualization technology where consumers deploy their applications packaged in virtual machines (VMs) on a virtualized infrastructure. Each VM hosts a complete software stack (operating system, middleware, application components) and instances of the VM can be dynamically added or removed based on the load variation. This fine-grained allocation is referred in the literature as on-the-fly elasticity (Vijayakumar et al., 2010).

However, complex workload patterns highly affect elasticity. Indeed, time-varying workload intensities are already challenging to handle in todays Internet systems. Workloads can vary for multi-tier applications by orders of magnitude within the same business day which makes it hard for the cloud providers (referred as providers) to optimally allocate VMs. Through awareness of workload changes, providers can more effectively overcome this challenge with less difficulty in the future.

Consumers are in the best position to predict how their workloads would change over time. However, the separation of cloud providers (referred as providers) and consumers hinders the former in having access to such information. Providers do not have direct access to the applications running inside the hosted VMs. Therefore, they cannot predict the applications future workload needs and consequently their future resource demands. Similarly, consumers do not have access to the hardware infrastructure where VMs are deployed. Therefore, they cannot anticipate the effects of sharing resources with third-party applications deployed by the provider on the same virtualized infrastructure. Because of this lack of information, consumers can only specify the type and amount of resources (e.g., number of CPUs) that should be allocated to their VMs by means of resource reservations communicated to the provider.

Due to above described information gap, cloud providers are not in the position to predict the future resource demands of consumers so that they can plan the capacity of their infrastructure accordingly. Cloud providers can receive arbitrary requests for allocating resources on-the-fly in a completely unpredictable manner. Given this unpredictability, it may happen that providers might not be able to provision the requested resources quickly enough, or in the worst case, they might ran out of capacity and may not be able to satisfy the consumers resource demands. The latter may lead to violations of Service Level Agreements (SLA) leading to loss of customers and reputation for both the cloud providers and cloud consumers. As a result, in order to guarantee SLAs, cloud consumers are forced to reserve more resources than they actually need resulting in over-provisioning and associated over-subscribed costs.

We propose a new reservation mechanism in order to protect consumers and providers from the cost overhead incurred due to over-provisioning. In this reservation mechanism, consumers can issue pre-reservations, referred to as soft reservations and then claim the pre-reserved resources by issuing normal reservations, referred to as hard reservations, if they actually end up needing them. Soft reservations capture the estimated amount of resources that will be required by a consumer at a given future point of time as well as the probability of actually needing these resources. This approach also aims at closing the information gap between consumers and providers by supplying a communication mechanism to exchange the relevant information for both parties.

The proposed approach comprises mechanisms to exploit the exchanged information in a beneficial way for both consumers and providers. Consumers would be able to use low level information about utilization of physical resources to better estimate their actual resource demands for running their services at the desired Quality of Service (QoS) level. Meanwhile, providers would be able to exploit the information about the expected future resource demands of their consumers to better plan the capacity of their infrastructure and continuously optimize the mapping of logical to physical resources resulting in lower data center operating costs and energy consumption.

The rest of the paper is organized as follows: Section 2 describes our resource reservation approach. We describe types of information we need to exchange between consumers and providers in Section 3. In Section 4, we survey the previous works in this area. Our end-to-end envisioned approach is summarized in Section 5. Finally, we summarize the paper in Section 6.

2 RESOURCE RESERVATIONS

Cloud providers provide on-demand access to scalable computing, storage and networking resources over a wide-area network. Consumers are able to deploy the VMs required to satisfy their SLAs with their customers (SaaS end-consumers). Consumers may dynamically ask for resources by placing resource reservations with the provider to match their varying workloads and respective resource demands. Once a consumer submits a request to the provider the request will be accepted, provided that enough resources are available. Otherwise, it would be rejected or some other options could be offered, i.e., a counter-offer (Lu et al., 2011) may be made. If the request is accepted, the provider will need to find a mechanism to satisfy the request. Consumers take into account several aspects such as amount, level of granularity, validity period, certainty, and provisioning intervals when making reservations. For example, a reservation could look like:

> "I need 10 nodes, each with 1 GB of memory, right now" or "I need 4 nodes, each with 2 CPUs and 2GB of memory, from 2pm to 4pm tomorrow".

The latter describes the amount of required resources for a specified time window (from 2 to 4 pm), whereas the former requests them immediately. Consumers

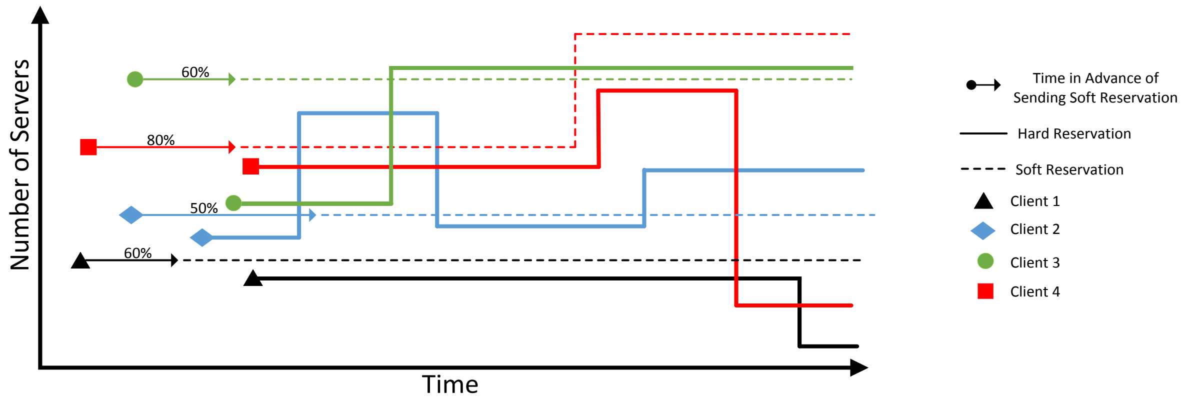


Figure 1: Snapshot of reservations received in infrastructure.

would normally not be able to predict in advance the exact amount of resources they will require due to the uncertainty about their future workloads and resource demands. However, consumers should normally be able to indicate how confident they are about the provided estimation. The latter can be done in different ways, e.g., by specifying the probability of actually requiring the reserved resources or by providing a cumulative distribution function of the amount of required resources.

2.1 Hard and Soft Reservations

As mentioned earlier, the actual resource utilization will only be known at runtime, so it is not possible for consumers to anticipate the real resource consumption in advance. In traditional resource reservation mechanisms, consumers have to pay for the requested resources even if they do not end up using them which is not aligned with the pay-per-use model in cloud computing.

Even though consumers may not be able to anticipate their exact resource demands, they would normally be able to approximately estimate the expected resource consumption based on workload forecasts and performance predictions with some level of certainty (Herbst et al., 2013). In the envisioned approach, consumers pre-reserve the forecasted required resources in long term through soft reservations. In these reservations, they basically specify how much resource they will need, in what time span, and how certain they are about their estimation.

The issuing of a soft reservation does not grant a consumer the requested resources. Nevertheless, once a consumer becomes more certain about his resource demands (near to the point of actual usage) traditional on-the-fly reservations (henceforth called hard reservations) can be issued to claim the resources that were previously pre-reserved through the corresponding soft reservations. If a consumer does not claim the required resources by means of hard reservations, the resources will not be allocated.

Figure 1 illustrates a snapshot of hard and soft reservations received by an IaaS provider and the time in advance of sending them by consumers over a period of time.

Consumers can choose only between using on-the-fly hard reservations or also taking advantage of the new soft reservations that allow them to book in advance resources from provider for a given time horizon (e.g., minutes, hours) in order to account for the perceived risk that a workload surge may occur. Informally, soft reservations will act as a form of insurance for consumers about obtaining resources at lower costs when needed provided that a correct bid for their future resource demands is communicated in advance to providers, whereas more expensive hard reservations (that are not preceded by a previous soft reservation) will be used to obtain unanticipated capacity that is required to process the current workload.

Soft reservations cater a win-win solution for both consumers and providers. For consumers, soft reservations will be much cheaper than hard reservations, since they only offer the right to obtain a set of future resources within a certain amount of time if they turn out to be required. If these resources are truly allocated at some point in the future, the consumer will have to pay additional compensation, but they will still save money compared to the on-the-fly hard reservations that would otherwise have to be used as workloads vary. If not allocated, the soft reservations would have instead merely served as an insurance policy for the consumer against high resource provisioning costs. However, the pricing model should provide a policy to avoid oversubscribed unclaimed soft reservations.

Similarly, providers will utilize the information provided through soft reservations as a basis for online capacity planning driving infrastructure management decisions. Upon observing changes in hard reservations, providers would dynamically allocate new capacity using standard mechanisms such as provisioning of new servers previously in stand-by mode. Providers can then use heuristics to optimize the placement for the new servers taking into account the currently active hard and soft reservations. Such heuristic algorithms should prioritize placements that improve the Total-Cost-of-Ownership (TCO) of the infrastructure.

Our proposed approach envisions different levels of soft reservations. Currently, four different dimensions are considered in order to define the softness level:

Provisioning interval of a soft reservation refers to the amount of time in which the softly-reserved resources are guaranteed to be provisioned if they end up being requested by issuing a respective hard reservation. The smaller the provisioning interval, the faster resources are guaranteed to be provisioned if they are claimed.

Validity period represents the validity time frame of a soft reservation. A reservation for one month would normally have higher importance and more implications for capacity planning than a reservation only for one day.

Time in advance represents the time in advance of sending a soft reservation before the desired period of its validity begins. A soft reservation for next week would normally have higher priority and more implications for capacity planning than a soft reservation with validity period beginning after one month.

Level of uncertainty refers to the estimated probability that the reserved resources will actually end up not being needed.

All four dimensions influence the degree of softness of soft reservations which in turn would normally influence the price for placing them. The softer reservations are, the cheaper their price would normally be expected to be.

Figure 2 illustrates the problems arises without soft reservations and benefits of the proposed hard and soft reservation compare to traditional mechanism. In this example, the provisioning interval and the level of uncertainty are fixed. The arrows depict the points in time at which soft reservation are submitted. One issue with the traditional reservation mechanism is the potential delay between the arrival of hard reservations and the actual provisioning of the requested resources (dashed red box on the left side of the first diagram). Another problem which might occur without soft reservations is that the provider might not be able to provision all of the requested resources (dashed red box on the right side of each diagram). The soft reservations help to address these two issues by enabling the provider to plan the capacity of the infrastructure such that all requested resources can be provided in time. The extent to which the softly reserved resources are guaranteed to be provisioned when placing a hard reservation depends on the four dimensions of the softness level explained above. Both exemplary

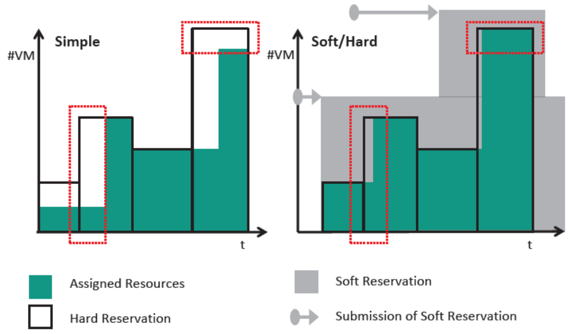


Figure 2: Example of hard and soft reservations.

problems described above are rooted in the inability of providers to anticipate what resources will be required by consumers in the future and thereby plan their capacity accordingly. The envisioned hard and soft reservation mechanism will tackle this problem by making it possible for consumers to communicate their estimated future resource demands.

3 INFORMATION EXCHANGE BETWEEN CLOUD PROVIDERS AND CONSUMERS

As mentioned earlier, the separation of cloud providers and consumers hinders providers to optimize their placement algorithms for provisioning IaaS resources. Moreover, the effects of sharing resources with third-party applications deployed in the same virtualized infrastructure are hidden from consumers. Hence, the exchange of information between the two stakeholders is beneficial for both. Providers can estimate the consumers future resource requirements and plan the allocation of VMs to Physical Machines (PM) improving the resource provisioning time and making significant cost savings at both ends. Moreover, information about the actual measured hardware utilization will allow consumers to only reserve and pay for the resources that are actually needed and to run their services at the service level required to fulfil the end-customers' (SLAs).

We identify two types of information which needs to be exchanged by consumers and providers, namely: (1) infrastructure monitoring-data supplied by the provider and (2) SLAs containing resource reservations placed by consumers.

3.1 Infrastructure Monitoring Data

Consumers suffer from the lack of direct access to the physical infrastructure level which is necessary to accurately monitor their resource consumption. Therefore, in order to enable consumers to accurately monitor and predict their future resource usage, providers must supply information and monitoring data about the physical infrastructure.

To provide such information, the provider needs to know the accuracy level and granularity of the measurement data required by each consumer. It is also important to know for how long historical data about resource utilization should be stored on the provider side in order to prevent infinitely growing large log files.

3.2 SLAs Containing Resource Reservations

Cloud providers normally do not have direct access to the applications and services running inside the VMs deployed on their infrastructure.

To bridge this gap, the proposed resource reservation mechanism offers a means for consumers to supply information about their expected future resource requirements based on workload forecasts and performance predictions. Such information should logically be part of the SLAs established between the consumers and providers.

The SLAs we are considering would not only cover classical metrics such as service response time and throughput, but also provide a powerful protocol for placing resource reservations, cancelling existing, or changing them. To realize these consumers need to know what types of resources are available for reservation and at what level of granularity they can be reserved (differentiating between general provider offerings, static agreements about the maximum allocations that could be provided to a consumer, and the availability of resources for possible reservation at a particular point in time).

4 STATE-OF-THE-ART

A vast amount of research exists in the literature on resource reservations in grid computing a summary of which can be found in (Rani et al., 2011). In cloud computing, advance reservations are an active area of research. In (Chaisiri et al., 2009), the authors present a stochastic integer program algorithm that works in an environment with multiple cloud providers. They propose an optimal virtual machine placement (OVMP) algorithm to minimize the total costs of reservations and on-demand resource provisioning. However, this approach does not consider any insurance policy for consumers allowing them to obtain their required resources at a cheaper price.

In (Mark et al., 2011), the authors address this problem in the same environment (Chaisiri et al., 2009), but they take different approach for handling future demands. They apply three different prediction algorithms (i.e., simple Kalman filter, double exponential smoothing, and Markov prediction) to predict the demands of customers, i.e., they use past usage history as a basis for forecasting future demands. In their approach, resource predictions takes place on the provider side while in ours, consumers are responsible for estimation of their future resource requirements.

Similarly, (Lu et al., 2011) provides a solution for the resource reservation problem in IaaS providers with limited resource capacity by which they are able to realize the feasibility of individual requests from consumers. If they are not able to satisfy the requests they will be able to provide an alternative offer by shifting requests in time (backward and forward) to fulfil them rather than refusing them. In their solution fragmentation in virtual resources is controlled and tried to be avoided. Resource requests are SLA-based and reservations take place during SLA negotiation. They utilize computational geometry for advanced reservation of resources.

Haizea [1] is a resource manager ("resource scheduler") software component which allows consumers to request resources from a computational resource. Haizea uses leases as a basic resource provisioning abstraction. A lease is "a negotiated and renegotiable agreement between a provider and a consumer, where the former agrees to make a set of resources available to the latter, based on a set of lease terms presented by the resource consumer". In (Sotomayor et al., 2009), designers of Haizea, present a model for predicting various runtime overheads involved in using virtual machines, which efficiently support advance reservations. They extend Haizea to use this new model in its scheduling decisions, and use it with the OpenNebula virtual infrastructure manager so the scheduling decisions will be enacted in a Xen cluster.

In (Wang et al., 2011), the management of QoS

[1] http://haizea.cs.uchicago.edu

in the presence of resource reservations in cloud environments is investigated. In order to guarantee QoS in the near future and maximize the total revenue of the resource provider, resource reservation requests should be accepted selectively. The decision is made based on the analysis of the possible achieved QoS after resource configuration.

In (Diaz et al., 2011), three types of resource requests which are rejected because of finite number of (PM)s and to the variability of VM resources utilization is identified. (1) Immediate Rejection (IR): If there is not enough available capacity in any PM. (2) Resources Allocation Rejection (RAR): resources are allocated, and VMs are already hosted in PMs. However, due to the variability of VM workload, the sum of resources utilized by VMs hosted in the same PM can exceed its capacity. Therefore, one or more VMs must be suppressed to free the resources in PM. (3) Total of Rejections: It is the sum of IR and RAR ratios. They propose a new concept of Resource Over-Reservation (ROR) as a mean to reduce RARs. The basic idea is to pre-reserve additional resources in order to stick at the load variation. The authors found a trade-off between the IR and RAR as a value for ROR to keep percentages of total request rejections low.

To summarize, in all of these works traditional resource reservations were considered as an input to the algorithms for placement of VMs and none of them consider the gap of information between providers and consumers to address resource reservations in cloud IaaS environments.

5 APPROACH

Our contribution is new ”soft” resource reservation mechanism for consumers. As we described in Section 2.1, consumers can issue their soft reservations based on their expected resource demand in long term with some level of certainty. Typically this is close to the actual resource consumption point, when they become certain about their resource requirements, they can claim their softly reserved resources through hard reservations.

After receiving soft and hard reservations, providers can use them to continuously improve the quality of virtual machine placement decisions and to plan the capacity of their infrastructure. To be effective, soft reservations should be coupled with a pricing model that reduces risk and TCO for both providers and consumers (Rizou and Polyviou, 2012).

We also propose the importance of data sharing between consumers and providers which was described in Section 3. This sharing of information will help both parties to characterize and dynamically react to unexpected changes of usage patterns of services and systems.

Figure 3 summarize our end to end envisioned approach. Consumers automatically adapt the amount of resources requested from providers based on the dynamic performance models which will be automatically calibrated at run-time. These predicted resources will be translated to soft and hard reservation and will be sent to providers. Providers adapt the mapping of requested logical resources to physical resources in a dynamic way and strictly accounting for TCO. One possibility to realize the online forecast-

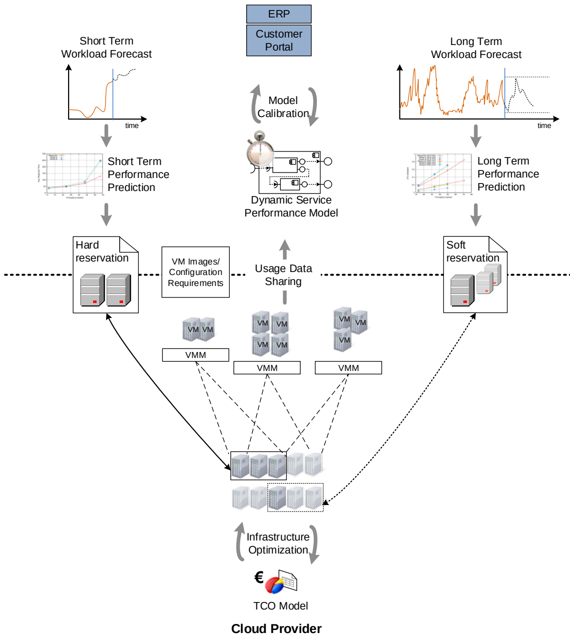


Figure 3: Online system performance model.

ing and performance predicting process is using the Descartes Meta Model (Huber et al., 2012). Within this approach, consumers and providers will individually work towards achieving better performance and TCO, thus promoting a more efficient use of data center resources at reduced cost. Particularly, consumers will automatically adapt the amount of resources requested from providers based on dynamic performance and TCO models which will be automatically maintained, calibrated, composed and evaluated at runtime (Kounev et al., 2011). This will produce continuously TCO optimization.

6 CONCLUSIONS

In this paper, we introduced our research roadmap through the definition of soft and hard reservations.

This mechanism enables consumers to communicate their workload forecasts in long term to providers by means of soft reservations. Consumers can claim the softly reserved resources when they become certain about their estimated resource requirements. Soft reservations act as an insurance policy that guarantees that the consumer will receive the softly reserved resources with cheaper price once they are requested through hard reservations. Similarly, providers have to send hardware utilization data (with respect to privacy of other customers) to the consumer. By means of these types of reservations, consumers would reserve and pay for the amount of resources they use and will receive their requested resources faster. Providers would be able to estimate the amount of resources they should provide in any point in time. Therefore, they would be able to manage their resources more efficiently by keeping PMs off or turn off PMs. In our future work, we intend to develop algorithms on the provider side to handle these reservations. These algorithms will cater for determining the expected required capacity at a given point of time in the future. Furthermore, our algorithms will identify future changes in capacity needs and will optimize the on-the-fly placement of VMs taking into account the costs of different reconfiguration options.

ACKNOWLEDGEMENTS

The research presented in this paper has been supported by the European Union within the FP7 Marie Curie Initial Training Network "RELATE" [2].

REFERENCES

Chaisiri, S., Lee, B.-S., and Niyato, D. (2009). Optimal virtual machine placement across multiple cloud providers. In *Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific*, pages 103 –110.

Diaz, F., Doumith, E., and Gagnaire, M. (2011). Impact of resource over-reservation (ror) and dropping policies on cloud resource allocation. In *Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on*, pages 470 –476.

Herbst, N. R., Huber, N., Kounev, S., and Amrehn, E. (2013). Self-Adaptive Workload Classification and Forecasting for Proactive Resource Provisioning. In *Proceedings of the 4th ACM/SPEC International Conference on Performance Engineering (ICPE 2013), Prague, Czech Republic, April 21–24*.

Huber, N., Brosig, F., and Kounev, S. (2012). Modeling Dynamic Virtualized Resource Landscapes. In *Proceedings of the 8th ACM SIGSOFT International Conference on the Quality of Software Architectures (QoSA 2012), June 25–28, 2012, Bertinoro, Italy*. Acceptance Rate (Full Paper): 25.6%.

Kounev, S., Brosig, F., and Huber, N. (2011). Self-Aware QoS Management in Virtualized Infrastructures (Poster Paper). In *8th International Conference on Autonomic Computing (ICAC 2011), June 14–18, 2011*, Karlsruhe, Germany.

Lu, K., Roblitz, T., Yahyapour, R., Yaqub, E., and Kotsokalis, C. (2011). Qos-aware sla-based advanced reservation of infrastructure as a service. In *Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on*, pages 288 –295.

Mark, C., Niyato, D., and Chen-Khong, T. (2011). Evolutionary optimal virtual machine placement and demand forecaster for cloud computing. In *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*, pages 348 –355.

Rani, B., Venkatesan, R., and Ramalakshmi, R. (2011). Resource reservation in grid computing environments: Design issues. In *Electronics Computer Technology (ICECT), 2011 3rd International Conference on*, volume 4, pages 66 –70.

Rizou, S. and Polyviou, A. (2012). Towards value-based resource provisioning in the Cloud. In *2012 IEEE 4th International Conference on Cloud Computing Technology and Science*.

Sotomayor, B., Montero, R., Llorente, I., and Foster, I. (2009). Resource leasing and the art of suspending virtual machines. In *High Performance Computing and Communications, 2009. HPCC '09. 11th IEEE International Conference on*, pages 59 –68.

Vijayakumar, S., Zhu, Q., and Agrawal, G. (2010). Dynamic resource provisioning for data streaming applications in a cloud environment. In *Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science*, CLOUDCOM '10, pages 441–448. IEEE Computer Society.

Wang, X., Xue, Y., Fan, L., Wang, R., and Du, Z. (2011). Research on adaptive qos-aware resource reservation management in cloud service environments. In *Services Computing Conference (APSCC), 2011 IEEE Asia-Pacific*, pages 147 –152.

[2]http://www.relate-itn.eu/

Generating a Tool for Teaching Rule-based Design

Stef Joosten and Gerard Michels

Department of Computer Science, Open University The Netherlands, Valkenburgerweg 177, Heerlen, Netherlands

{stef.joosten, gerard.michels}@ou.nl

Keywords: Business Rules, Requirements Engineering, Software Generation, Computer Science Education.

Abstract: Software generation from a formal requirements specification has enabled a small research team to develop a tool set for educational design exercises and didactical research. This approach was needed to obtain a development environment, which is responsive to changing requirements due to maturing didactics and future research questions. We use Ampersand, a rule-based design methodology, to specify and generate the tool set called the Repository for Ampersand Projects (RAP). RAP is being used in a course on Ampersand for master students of computer science and business management. Analytic tools have been interconnected to RAP to obtain analytics about student activities in RAP. So, Ampersand is both the subject of teaching and research as well as an asset used to develop and maintain RAP. In this paper we present how RAP has been generated with Ampersand and reflect upon the value of this design choice.

1 INTRODUCTION

The authors have developed a Repository for Ampersand Projects (RAP) to facilitate teaching Ampersand as well as research on that subject. Ampersand is a methodology to design information systems and business processes as a formal collection of rules. RAP is a tool set for Ampersand design exercises, which features rule-based interfaces to connect analytic tools in a way that preserves semantics of data. The analytic tools use semantic data in RAP to produce unambiguous measurement results for our research. The objective of our research is to understand how to teach Ampersand to master students of computer science and business management. The first study using data from RAP has been accepted for publication (Michels and Joosten, 2013).

This paper reflects on our choice to use Ampersand to generate RAP from functional requirements. Ampersand was a prerequisite in this choice c.q. the question is whether rule-based design has brought us closer to our research objective. The answer is positive. By using Ampersand we have obtained a development environment for RAP, that enables us to respond to changing requirements in a controlable and timely fashion. We can adopt new didactical insights in the exercise tool and facilitate our current and future studies with unambiguous analytics from RAP. Thus, this paper presents the fundament of our environment to study and improve the teaching of Ampersand i.e. the generation of RAP with Ampersand.

Section 2 provides a background on using rules as requirements. Section 3 introduces Ampersand as a rule-based approach on a formal language to design information systems. Section 4 describes the generic and specific issues of generating RAP with Ampersand e.g. rule-compliant processes, customized data access. We conclude with a reflection on our accomplishments with RAP up to date and expectations for the future.

2 RULES AS REQUIREMENTS

This paper argues that a rule-based design approach can reduce the gap between the owners of requirements (end users, patrons, auditors, etc.) and the design of information systems, by giving compliance guarantees to these owners and by giving the appropriate tools to requirements engineers. The Business Rules Community (Ross, 2003b) has argued since a long time that natural language provides a better means for discussing requirements than graphical models (e.g. UML (Rumbaugh et al., 1999)). This vision is the fundament of profound assets like the Business Rules Manifesto (Ross, 2003a), the Business Rules Approach (Ross, 2003b), the SBVR standard (Object Management Group, Inc., 2008) and RuleSpeak (Business Rule Solutions, LLC., 2013). Rule-based design goes beyond requirements by formalizing business rules and using them as requirements to partially automate information system development.

Rule-based design uses the word *business rule* to denote a formal representation of a business requirement. Business rules have been represented in many different ways. Prolog (Clocksin and Mellish, 1981) is an early rule-based approach that uses Horn-clauses as a means to write computer programs. Widespread are also Event-Condition-Action (ECA) rules of active databases, such as (Dayal et al., 1988; Paton and Díaz, 1999; Widom, 1996), which represent successful results of earlier research into functional dependencies between database transactions in the seventies and eighties. Expert systems and other developments founded in ontology (Gruber, 1993; Berners-Lee et al., 2001) can be regarded as ways to represent business processes by means of rules. Approaches based on event traces, such as Petri Nets (Reisig, 1985) and ARIS (Scheer, 1998), have dominated the 90's and are still popular to date (Green and Rosemann, 2000). The information systems community is aware (e.g. (Ram and Khatri, 2005)) that mathematical representations of business rules can be useful. For example, Date's criticism of SQL for being unfaithful to relational algebra (Date, 2000) advocates a more declarative approach to information system design, and puts business rules in the center of the design process.

This paper argues that business requirements are sufficient to generate a functional prototype of an information system. The word 'sufficient' suggests that requirements engineers need not communicate with the business in any other way. This suggestion is seriously flawed. Models are still useful, but their role changes. In Ampersand, models are artifacts, preferrably produced automatically, that document the design. If desired, a requirements engineer can avoid to discuss these artifacts (data models, etc.) with the business, but they are available and undeniably useful as documentation in the design process. Ampersand shifts the focus of the design process to requirements engineering, because a larger part of the process is automated.

Controlling design processes directly by means of business rules has consequences for requirements engineers, who will encounter a simplified design process. From their perspective, the design process is depicted in figure 1. The main task of a requirements engineer is to collect rules to be maintained. These rules are to be managed in a repository (RAP). From that point onwards, a first generator (G) produces various design artifacts, such as data models, process models etc. These design artifacts can then be fed into another generator (an information system development environment), that produces the actual system. That sec-

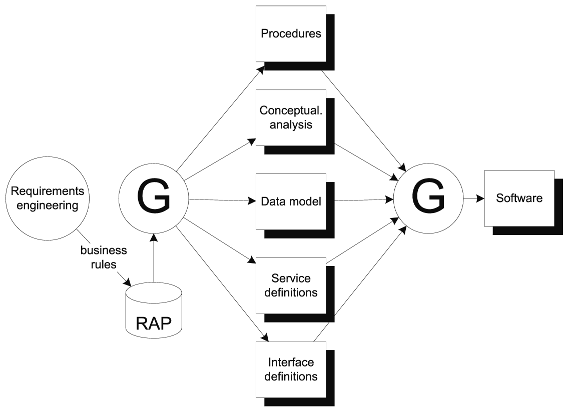


Figure 1: Rule-based design process (engineer).

ond generator is typically a software development environment, of which many exist and are heavily used in the industry. Alternatively, the design can be built in the conventional way as a database application. A graphical user interface on the repository and generator will help the requirements engineer by storing, managing and checking rules, to generate specifications, analyze rule violations, and validate the design.

From the perspective of an organization, the design process looks like figure 2. At the focus of at-

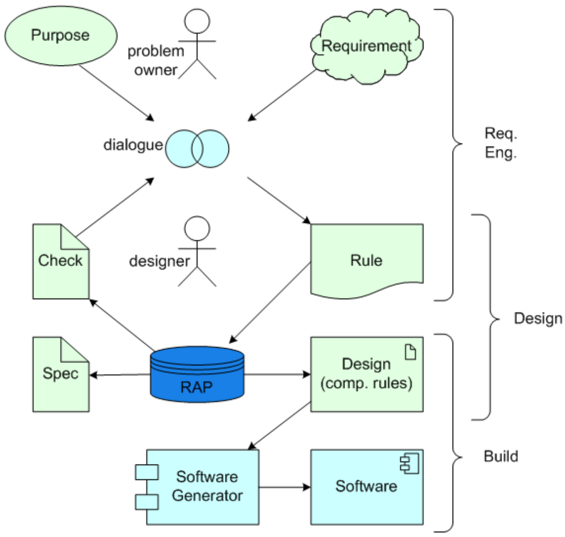


Figure 2: Rule-based design process (organization).

tention is the dialogue between a problem owner and a requirements engineer. The former decides which requirements he wants and the latter makes sure they are captured accurately and completely. The requirements engineer helps the problem owner to make requirements explicit. Ownership of the requirements remains in the business. The requirements engineer can tell with the help of his tools whether the requirements are sufficiently complete and concrete to make a buildable specification. The requirements engineer sticks to the principle of one-requirement-one-rule, enabling him to explain the correspondence of the specification to the business.

3 AMPERSAND

The purpose of Ampersand is to have the right interaction with stakeholders to define the right business rules and represent them unambiguously. Ampersand looks at business rules not only as an agreement between parties, but uses them as functional requirements to automate an information system as well. That is: these rules are to be maintained by all parties at all times and the IT must support that. Maintaining these rules is done either by people (of any party) or by computers.

A requirements engineer using Ampersand defines a suitable relational information structure to define rules upon. A meaningful specification of rules in Ampersand can be incomplete or even contain no business rule at all. In that matter, Ampersand clearly differs from more common relational specification languages like Alloy (Jackson, 1999). No business rules on an information structure represent ultimate freedom when using that information structure in business processes. This kind of freedom is explicitly useful in experimental contexts like the educational design exercises in RAP. For example, a student can first define an information structure and add rules one-by-one while studying the impact of adding another rule. More practical reasons to omit a business rule are disagreement between stakeholders, irrelevance of the rule, or unawareness of the rule.

Ampersand uses a relation algebra (Maddux, 2006) as a language in which to express business rules. Relation algebras have been studied extensively and are well known for over a century (Schröder, 1895). The use of existing and well described theory brings the benefit of a well conceived set of operators with well known properties.

The Ampersand syntax consists of constant symbols for (business) concepts, (business) elements, relations and relation operators. Relation terms can be constructed with relations and relation operators.

Let $\mathbb{C}$ be a set of concept symbols. A concept is represented syntactically by an alphanumeric character string starting with an upper case character. We use A, B, and C as concept variables.

Let $\mathbb{U}$ be a set of atom symbols. An atom is represented syntactically by an ASCII string within single quotes. All atoms are an element of a concept, e.g. `'Peter'` is an element of `Person`. We use a, b, and c as atom variables.

Let $\mathbb{D}$ be a set of relation symbols. A relation symbol is represented syntactically by an alphanumeric character string starting with a lower case character. For every $A, B \in \mathbb{C}$, there are special relation symbols, I_A and $V_{A\times B}$. We use r, s, and t as relation variables.

Let $\neg$, ${}^{\sqcup}$, $\sqcup$, and $\fatsemi$ be relation operators of arity 1, 1, 2, and 2 respectively. The binary relation operators $\sqcap, \Rightarrow$ and $\equiv$ are cosmetic and only defined on the interpretation function.

Let $\mathbb{R}$ be the set of relation terms. We use R, S, and T as variables that denote relation terms.

$\mathbb{R}$ is recursively defined by

$$I_A, V_{A\times B}, r, \neg R, R^{\sqcup}, R \sqcup S, R \fatsemi S \in \mathbb{R}$$

provided that $R, S \in \mathbb{R}, r \in \mathbb{D}$, and $A, B \in \mathbb{C}$

An Ampersand script for context $\mathfrak{C}$ is a user-defined collection of statements where

- $RUL \subseteq \mathbb{R}$ is a collection of relation terms called rules.
- REL is a collection of $r : A \sim B$ for all $r \in \mathbb{D}$ such that $A, B \in \mathbb{C}$. The instances of REL are called relation declarations.
- POP is a collection of $a\ r\ b$ such that $a \in A, b \in B$ and $(r : A \sim B) \in REL$.

The relation declarations define the conceptual structure and scope of $\mathfrak{C}$. Relation elements define facts in $\mathfrak{C}$. POP is called the population of $\mathfrak{C}$. Rules are constraints on that population.

The interpretation function $\mathfrak{I}(R)$ defines the semantics of a relation term R. This function interprets relation terms based on POP and a relation algebra $\langle \mathbb{R}, \cup, \bar{\ }, ;, \smile, \mathbb{I} \rangle$ where $\mathbb{R} \subseteq \mathcal{P}(\mathbb{U})$. All relation symbols used in a relation term are either declared by the user in REL or I_A or $V_{A\times B}$. So, the relation algebra on $\mathbb{R}$ is configured by the user through REL. The interpretation of all relation symbols in $\mathbb{D}$ is completely user-defined through POP. Thus, given some REL and some POP, $\mathfrak{I}(R)$ determines whether some relation holds between two elements.

Given some $\mathfrak{C}$, the interpretation function of relation terms is defined by

$$\text{relation} \quad \mathfrak{I}(r) = \{\langle a,b \rangle \mid a\ r\ b \in POP\} \tag{1}$$

$$\text{identity} \quad \mathfrak{I}(I_A) = \{\langle a,a \rangle \mid a \in A\} \tag{2}$$

$$\text{universal} \quad \mathfrak{I}(V_{A\times B}) = \{\langle a,b \rangle \mid a \in A,\ b \in B\} \tag{3}$$

$$\text{complement} \quad \mathfrak{I}(\neg R) = \overline{\mathfrak{I}(R)} \tag{4}$$

$$\text{converse} \quad \mathfrak{I}(R^{\sqcup}) = \mathfrak{I}(R)^{\smile} \tag{5}$$

$$\text{union} \quad \mathfrak{I}(R \sqcup S) = \mathfrak{I}(R) \cup \mathfrak{I}(S) \tag{6}$$

$$\text{composition} \quad \mathfrak{I}(R \fatsemi S) = \mathfrak{I}(R) ; \mathfrak{I}(S) \tag{7}$$

(the interpretation of the mentioned cosmetic relation operators)

$$\text{intersection} \quad \mathfrak{I}(R \sqcap S) = \mathfrak{I}(\neg(\neg R \sqcup \neg S)) \tag{8}$$

$$\text{implication} \quad \mathfrak{I}(R \Rightarrow S) = \mathfrak{I}(\neg R \sqcup S) \tag{9}$$

$$\text{equivalence} \quad \mathfrak{I}(R \equiv S) = \mathfrak{I}((R \Rightarrow S) \sqcap (S \Rightarrow R)) \tag{10}$$

Relations need to have a type to be more practical for requirements engineers (Michels et al., 2011). This way, only a relation term $R \in \mathbb{R}$ with a type $\mathfrak{T}(R)$ has an interpretation. A relation term without a type $\mathfrak{T}(R)$ is said to have a type error or to be (semantically) incorrect. Compilers for Ampersand scripts must reject relation terms with a type error. A requirements engineer might call a relation term without a type *nonsense*.

Van der Woude and Joosten (van der Woude and Joosten, 2011) have enhanced the typing function with subtyping of concepts. Subtyping is useful to confront two different, but comparable concepts without being rejected as a type error. For example, a requirements engineer can model the concept `Client` and `Provider` to be comparable over a more general concept. This might make sense when a client can be a provider as well. A course book on Ampersand (Wedemeijer et al., 2010) also discusses alternative patterns in Ampersand to model apparent subtypes of concepts.

4 GENERATING RAP

RAP includes first and second generator functions and the repository (RAP) of figure 1. The generator functions are part of a command-line tool called the Ampersand compiler. A web application of RAP for design exercises provides access to the repository and generator functions. The Ampersand compiler is manually coded in Haskell. Source and binary files of the Ampersand compiler are freely available via `wiki.tarski.nl`. The web application and repository are generated with the compiler from an Ampersand script for RAP. RAP and its full script are freely available at request.

We claim that the Ampersand compiler can generate a compliant information system from business requirements in an Ampersand script. Early versions of the Ampersand compiler could already generate a trivial, but compliant system. Such a trivial system consists of a relational database, a web interface, and a rule engine. The database model contains a table of two columns for each relation declaration in the script, to hold its population. An initial population for the database is checked to be free of violations. The web interface provides a user with basic data functions for the database. A rule engine checks all the rules before committing changes to the database in order to remain compliant with the rules c.q. the requirements. The current version of the Ampersand compiler takes rules into account and uses simple syntactic Ampersand constructs to derive more practical system components. Examples of Ampersand constructs for practical enhancements are: an interface construct to customize data access; attributes to customize violation handling; and plain text attributes to attach meaning or purpose to formal elements for traceability. In the next subsection we address a generic issue: How do business rules yield a compliant business process? In the remaining subsections we describe how a system, RAP, could be generated, which has shown to be sufficiently practical for studies on and design exercises of a course at the Open University. We use examples from the Ampersand script for RAP.

4.1 Compliant Processes

Whenever and whereever people work together, they connect to one another by making agreements and commitments. These agreements and commitments constitute the rules of the business. The role of information technology is to help *maintain* business rules. This is what compliance means. If any rule is violated, a computer can signal the violation and prompt people or trigger a computer to resolve the issue. This can be used as a principle for controlling business processes. For that purpose two kinds of rules are distinguished: rules that are maintained by people and rules that are maintained by computers. A rule maintained by a computer is an automated activity within a business process.

Since all formalized rules (both the ones maintained by people and the ones maintained by computers) are monitored, computers and persons together form a system that lives by those rules. This establishes compliance. Business process management (BPM) is also included, based on the assumption that BPM is all about handling cases. Each case (for instance a credit approval) is governed by a set of rules. This set is called the *procedure* by which the case is handled (e.g. the credit approval procedure). Actions on that case are triggered by signals, which inform users that one of these rules is (temporarily) violated. When all rules are satisfied (i.e. no violations with respect to that case remain), the case is *closed*. This yields the controlling principle of BPM, which implements Shewhart's Plan-Do-Check-Act cycle (often attributed to Deming) (Shewhart and Deming, 1939).

Assume the existence of an electronic infrastructure that contains data collections, functional components, user interface components and whatever is necessary to support the work. An adapter observes the business by drawing digital information from any available source. The observations are fed to a rule engine, which checks them against business rules in a repository. If rules are found to be violated, the rule engine signals a process engine. The process engine distributes work to people and computers, who take appropriate actions. These actions can cause new signals, causing subsequent actions, and so on until the process is completed. This principle rests solely on rules, yielding implicit business processes which directly follow from the rules of the business. In comparison: workflow management derives actions from a workflow model, which models the procedure in terms of actions. Workflow models are built by modelers, who transform the rules of the business into actions and place these actions in the appropriate order to establish the desired result.

4.2 Rule-based Structure of RAP

RAP is an application on a repository of Ampersand scripts. If RAP was just a storage for scripts, then its Ampersand structure could remain limited to only one concept $Script \in \mathbb{C}$; no rules; no relations. However, we want RAP to have functions to facilitate certain activities. For those functions, concepts, relations and rules need to be added to the script of RAP.

We have used RAP as:

- a design exercise tool for students. A student user of RAP has a facility to upload a script to RAP, a structured view on a script in RAP, and access to a few compiler functions to run on a script in RAP. The second release of the exercise tool anticipates on more roles than students, which are teachers, advanced students, and requirements engineers. The second release introduces rule-guided editing of a script in the structured view;
- a data source for measurements to study student behaviour;
- a data storage for measurement results to use in the exercise tool for learning.

No additional structure was needed in the script of RAP to use RAP as a data source for measurements.

If the results of measurements need to be stored in RAP, then we need at least a univalent relation from the source of the measure to the result. For example, the structured view on a script includes the counters of rules, relations and concepts, which are measurements needed for a design activity called cycle chasing. Each counter is a functional relation in the script of RAP e.g. $count\_rules : \texttt{Script} \rightarrow \texttt{Number}$.

Most of the script of RAP relates to student activities in the exercise tool. The upload facility is a handcoded component of RAP, and is thus excluded from the script of RAP. The structured view follows the syntactic structure of a script as implemented by the script parser of the compiler. The compiler including parser is coded in Haskell, which is a functional programming language. The functional structure of a script in Haskell could easily be transposed into a relational structure in Ampersand, because a relation is less strict than a function. The view also contains derivates from a script like conceptual diagrams, a diagnosis on the typing function(Michels et al., 2011), or a report of rule violations. For example, the source concept of a relation declaration is a functional relation in the script of RAP, $source : \texttt{Concept} \rightarrow \texttt{Declaration}$. In the Haskell code of the compiler, the data structure *Declaration* has an attribute of type *Concept* to hold its source concept. A compiler function on a script is implemented as a functional relation from a script to a web location where the compiler output is accessible. For example, the script of RAP contains a relation to generate a functional specification from a script, $gen\_fspec : \texttt{Script} \rightarrow \texttt{FSpec}$.

4.3 Custom Data Access

An interface construct exists in the Ampersand language to access the population of relations by means of relation algebraic expressions. An interface is a tree of labeled expressions where each node connects a parent R and child S with a composition operator $R \fatsemi S$. The interface provides an easy way to configure data access with the full power of the relation algebra. The relation algebra also has restrictions, for instance numerical calculations are not possible. Attributes can be set to customize data access like who may use an interface and which relations in an interface can be altered with that interface.

We demonstrate how to configure an interface by an example from RAP. The following interface is used to generate a web page for students to view a context.

```
-1- INTERFACE "CONTEXT" FOR Student:I[Context]
-2- BOX ["name" : ctxnm
-3-     ,"PATTERNs" : ctxpats
-4-     ,"concepts" : ctxcs
-5-     ,"ISA-relations" : ctxpats;ptgns
-6-     ,"relations" : ctxpats;ptdcs
-7-     ,"RULEs" : ctxpats;ptrls]
```

The text elements in double quotes are labels. The text after a colon is an ASCII-encoded relation algebraic expression. The first line defines a root node with the identity relation of a concept *Context*. This means that the interface can be used to view or alter relations of any instance of the domain of the identity relation of *Context*. This interface is a view-only interface, because the attribute to grant access to edit certain relations in the interface is not set. The optional FOR-attribute restricts usage of this interface to a *Student* user only. Each instance of *Context* represents a context $\mathfrak{C}$, that could have been parsed from a script by the compiler. The box of line 2 to 7 connects six sibling nodes to the root by means of a composition operator. The relation *ctxnm* : `Context`→`ContextName` holds the relation of user-friendly names for a $\mathfrak{C}$. The relation *ctxpats* : `Context` ∼ `Pattern` holds the relation to patterns in the $\mathfrak{C}$. Patterns are syntactic containers for rules, declarations and ISA-relations i.e. *ptrls* : `Pattern` ∼ `Rule`, *ptdcs* : `Pattern` ∼ `Declaration`, and *ptgns* : `Pattern` ∼ `Gen`. The relation *ctxcs* : `Context` ∼ `Concept` holds the relation to concepts in the $\mathfrak{C}$.

4.4 Configure Layout

The layout of a web page generated from an interface is configured in Cascading Style Sheets (CSS). These web pages contain labels and textual data.

Little efforts were needed to customize the default style sheets to suit RAP. We experienced that solutions to improve the user experience and comfort are more difficult to implement in a generated page than in a hand coded page. But when such a solution is implemented on a good design, then you can easily take advantage of it in any page.

For example, RAP needs to display non-textual data like images and handlers to execute parameterized software functions. A view on a concept is invented that allows the requirements engineer to compose non-textual HTML elements based on instances of a concept. For example instances of the concept *ConceptualDiagram* are urls to actual images, which are embedded in an HTML element to display images. Likewise, all software functions on data in RAP are encapsulated by the only manually coded web page of RAP (index.php). RAP has a concept *G* refering to the G in figure 1, which contains software functions to compose HTML links to execute software like compiler functions. The Ampersand code below configures a view on an instance of G.

```
VIEW G:
 G(PRIMHTML
    "<a href='../../index.php?operation="
  ,operation          ,PRIMHTML "&file="
  ,applyto;filepath  ,applyto;filename
  ,PRIMHTML "'>"      ,functionname
  ,PRIMHTML "</a>")
```

PRIMHTML becomes actual HTML on the generated web page. Functional relations with the instance of G as a source fill the parameters of the url. The relation *operation* : `G`→`Operation` relates instances of G to an operation number. The relation *applyto* : `G`→`Script` relates instances of G to the script to which the operation of G must be applied. The relations *filepath* : `Script`→`FilePath` and *filepath* : `Script`→`FilePath` relate the script to a location on a file system. The relation *functionname* : `G`→`String` relates instances of G to a textual element to click on in order to execute a software function.

4.5 Manage Measurements for Research

The most prominent advantage of using RAP for measurements in our experience is the documentation and qualitity of input data. The quality of data in RAP is high, because data is embedded in a formally defined information structure. The information structure of data has a clear and compliant relation with business requirements, such that the purpose and meaning of data can be documented and verified easily.

Measurements do not need to be part of the script of RAP to use RAP as a data source for measurements. This makes it easy to adapt to new measurements, because no development efforts are required. We do have added a univalent relation for each of our measurements for studies to the script of RAP. Adding a univalent relation is a small effort with hardly any impact on RAP. A small advantage is gained because measurements can be documented and managed in a structured way. We have experienced control over our measurements in their implementatation and use.

Measurements have been implemented as extensions on the compiler and in spreadsheets.

5 CONCLUSIONS

In this paper we have presented an application of rule-based design to obtain a development environment for RAP.

Does RAP fulfill its purpose to facilitate teaching Ampersand as well as research on that subject? We have shown that the functional requirements for RAP can be formalized by means of an Ampersand script, such that RAP can be generated. Section 4 describes all the details of how RAP has been constructed. About 50 students per year have used RAP to complete a design exercise which defines 80% of the final grade of a master course on rule-based design. A first study has been accepted for publication

(Michels and Joosten, 2013), which uses analytics from RAP based on 52 students who have used RAP in the period between April 2010 and May 2011. This publication reports on six hypotheses based on those analytics in order to explore the possibilities to study student behaviour with RAP. From the above observations we conclude that:

- RAP has been generated with Ampersand;
- RAP is sufficiently practical for teaching and research.

For further research we have planned to identify didactical requirements for the exercise tool for students. Current analytics in RAP will be taken into account. The objective is to upgrade the 'sufficiently practical exercise tool' to a 'teaching exercise guide for students'.

Has the application of Ampersand brought us closer to understand how to teach Ampersand to master students of computer science and business management? With the development environment of RAP we created, we are ready to adopt RAP to new didactical insights and produce unambiguous analytics. Our first study has produced preliminary results on how to teach Ampersand. Thus, our research has progressed due to the application of Ampersand because:

- with Ampersand we have created a responsive and controlable environment for research;
- with Ampersand we could produce unambiguous analytics to show that meaningful research with RAP is feasible (Michels and Joosten, 2013).

The next step in research is to define didactical studies on how to teach Ampersand, which are based on analytics in RAP.

REFERENCES

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*, 284(5):34–43.

Business Rule Solutions, LLC. (2013). RuleSpeak. Retrieved Februari 27, 2013, from `http://www.rulespeak.com`.

Clocksin, W. F. and Mellish, C. (1981). *Programming in Prolog*. Springer.

Date, C. J. (2000). *What Not How: the Business Rules Approach to Application Development*. Addison-Wesley Longman Publishing Co., Inc., Boston.

Dayal, U., Buchmann, A. P., and McCarthy, D. R. (1988). Rules are objects too: A knowledge model for an active, object-oriented database system. In Dittrich, K. R., editor, *OODBS*, volume 334 of *Lecture Notes in Computer Science*, pages 129–143. Springer.

Green, P. F. and Rosemann, M. (2000). Integrated process modeling: An ontological evaluation. *Inf. Syst.*, 25(2):73–87.

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220.

Jackson, D. (1999). A comparison of object modelling notations: Alloy, UML and Z. Technical report, Retrieved Februari 27, 2013, from `http://people.csail.mit.edu/dnj/publications/alloy-comparison.pdf`.

Maddux, R. (2006). *Relation Algebras*, volume 150 of *Studies in logic*. Elsevier, Iowa.

Michels, G. and Joosten, S. (2013). Progressive development and teaching with RAP. Accepted at CSERC'13.

Michels, G., Joosten, S., van der Woude, J., and Joosten, S. (2011). Ampersand: Applying relation algebra in practice. In *Proceedings of the 12th conference on Relational and Algebraic Methods in Computer Science*, Lecture Notes in Computer Science 6663, pages 280–293, Berlin. Springer-Verlag.

Object Management Group, Inc. (2008). Semantics of Business Vocabulary and Business Rules (SBVR), v1.0. Technical report, Retrieved Februari 27, 2013, from `http://www.omg.org/spec/SBVR/1.0/PDF`.

Paton, N. W. and Díaz, O. (1999). Active database systems. *ACM Comput. Surv.*, 31(1):63–103.

Ram, S. and Khatri, V. (2005). A comprehensive framework for modeling set-based business rules during conceptual database design. *Inf. Syst.*, 30(2):89–118.

Reisig, W. (1985). *Petri Nets: an Introduction*. Springer-Verlag New York, Inc., New York, NY, USA.

Ross, R. G. (2003a). The Business Rules Manifesto. Retrieved Februari 27, 2013, from `http://www.businessrulesgroup.org/brmanifesto.htm`.

Ross, R. G. (2003b). *Principles of the Business Rule Approach*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Rumbaugh, J., Jacobson, I., and Booch, G., editors (1999). *The Unified Modeling Language reference manual*. Addison-Wesley Longman Ltd., Essex, UK, UK.

Scheer, A.-W. W. (1998). *Aris-Business Process Frameworks*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2nd edition.

Schröder, E. (1895). *Algebra und Logik der Relative*. Vorlesungen über die Algebra der Logik (Exakte Logik) / Ernst Schröder. Teubner.

Shewhart, W. and Deming, W. (1939). *Statistical Methods from the Viewpoint of Quality Control*. Dover Books on Mathematics Series. Dover Publications, Incorporated.

van der Woude, J. and Joosten, S. (2011). Relational heterogeneity relaxed by subtyping. In *Proceedings of the 12th conference on Relational and Algebraic Methods in Computer Science*, Lecture Notes in Computer Science 6663, pages 347–361, Berlin. Springer-Verlag.

Wedemeijer, L., Joosten, S., and Michels, G. (2010). *Rule Based Design*. Open Universiteit Nederland, 1st edition.

Widom, J. (1996). The starburst active database rule system. *IEEE Trans. on Knowl. and Data Eng.*, 8(4):583–595.

The Development of an Imitation Model of a Multi-tenant Database Cluster

Evgeny Boytsov and Valery Sokolov
Department of Computer Science,Yaroslavl State University, Yaroslavl, Russia
{boytsovea, valery-sokolov}@yandex.ru

Keywords: Databases, SaaS, Multi-tenancy, Imitation Modelling.

Abstract: This paper deals with the main principles forming the foundation of an imitation model of a multi-tenant database cluster — the concept of reliable and easy to use data storage for high load cloud applications with thousands of customers, based on ordinary relational database servers. The main architectural principles of cloud applications are discussed; some statistics about real multi-tenant cloud application is given. This statistics is interpreted to detect the key characteristics of a flow of queries. Basing on this interpretation, the architecture of the imitation model and the process of its development are described.

1 INTRODUCTION

The development of a cloud application throws a lot of architectural challenges to software engineers. One of such challenges is the problem of organizing the storage of data in the cloud with the requirement of high performance, fault-tolerance and reliable tenants' data isolation from each other. At the moment these tasks are usually solved by designing an additional layer of application logic at the level of application servers. Such a technique is discussed in a lot of specialized papers for application developers and other IT-specialists (Chong et al., 2006a/b, Candan et al., 2009). There are also some projects of providing native multi-tenancy support at the level of a single database server (Schiller et al., 2011). This paper is devoted to the alternative concept of a multi-tenant database cluster which proposes the solution of the above problems at the level of a data storage subsystem. In particular, the key characteristics of the query flow are being studied basing on the statistics of an existing multi-tenant cloud application. The correlation between various parameters of a tenant and its query flow are studied. The process of the development of cluster imitation model is described and basic principles of the model architecture and mode of operation are highlighted.

2 THE ARCHITECTURE OF THE MULTI-TENANT DATABASE CLUSTER

A multi-tenant database cluster (Boytsov and Sokolov, 2012) is a concept of data storage subsystem for cloud applications. It is an additional layer of abstraction over ordinary relational database servers with a single entry point which is used to provide the isolation of cloud application customers' data, load-balancing, routing of queries among servers and fault-tolerance. The main idea is to provide an application interface which has most in common with the interfaces of traditional RDBMS (relational database management system). At the moment a typical scenario of interaction with the cluster from the developer point of view is seen as the following:

```
Connect(TenantId,ReadWrite/ReadOnly);
SQL-commands
Disconnect();
```

A multi-tenant cluster consists of a set of ordinary database servers and specific control and query routing servers.

The query routing server is a new element in a chain of interaction between application servers and database servers. This is the component application developers will deal with. In fact, this component of the system is just a kind of a proxy server which hides the details of the cluster structure, and whose main purpose is to find an executor for a query and route the query to him as fast as possible. It makes a decision based on the map of a cluster.

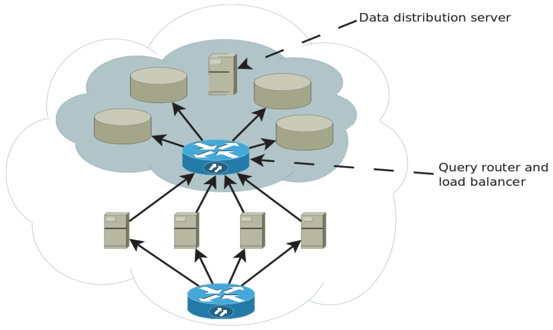


Figure 1: Multi-tenant database cluster architecture.

It is important to note that a query routing server has a small choice of executors for each query. If the query implies data modification, there is no alternative than to route it to the master database of a tenant, because only their data modification is permitted. If the query is read-only, it can also be routed to a slave server, but in the general case there would be just one or two slaves for a given master, so even in this case the choice is very limited.

The data distribution and load balancing server is the most important and complicated component of the system. Its main functions are:

- initial distribution of tenants data among servers of a cluster during the system deployment or addition of new servers or tenants;
- management of tenant data distribution, based on the collected statistics, including the creation of additional data copies and moving data to another server;
- diagnosis of the system for the need of adding new computing nodes and storage devices;
- managing the replication.

This component of the system has the highest value since the performance of an application depends on the success of its work.

3 ANALYSIS OF EXISTING APPLICATION

Analysis of existing applications and their mode of operation is the first thing to study when designing an imitation model. In the context of the multi-tenant cluster theme the most interesting question is the characteristics of the query flow, since this component has the greatest impact on the results obtained during the modelling. As the multi-tenant cluster is a queuing system, the Poisson flow of events is a good basic model of a query flow. The key points to explore are:

1. intensity distribution of incoming query flows among clients;
2. presence or absence of dependency between an average time of query execution and characteristics of the client which this query belongs to;
3. characteristics of a customer base;
4. characteristics of customer base changes over time.

Since questions 1 and 2 have a significant impact on the distribution of queries between servers thus making a decisive contribution to the assessment of the efficiency of load balancing across the cluster as a whole, they are very important. The answer to the fourth question will allow us to adequately simulate the dynamism inherent to all cloud systems and therefore offer an effective long-term data management strategy.

There are many factors that possibly can affect parameters of a client query flow. At the initial stage of the study it was decided to take the size of the data that the client stores in the cloud as its key characteristic. The relationship between this parameter and the intensity of the query flow or an average time of query execution has been studied. The following assumptions seemed to be reasonable:

1. the most of client schemas are approximately of the same size, but there are also significant (but rare) variations in both directions;
2. client query flow intensity is directly dependent on the size of client data (the greater data the client has, the more often they are accessed);
3. the query execution time is directly dependent on the size of client data (the greater data the client has, the more data are accessed by the average query, thus its execution time increases);
4. client data size and activity smoothly change over time.

The verification of the above assumptions has been performed on the basis of statistics and logs of the existing multi-tenant cloud application. This application is the online service that provides an electronic flow of documents and accounting. The diversity of offered services leads to the diversity of possible scenarios of interaction between a client and the application, thus making a complicated query flow. The application uses Postgres SQL server as its primary data storage. All management stuff is performed by a set of specialized services and routers. Currently, the cluster consists of about database 60 servers and continues to grow. The statistics being investigated is incomplete and rather inaccurate due to the way it is collected. There is a dedicated service that is used to collect this statistics. It “wakes up” every 24 hours and processes logs of all application servers within the cluster, making conclusions about average values of key characteristics. Since this service is only able to analyse the entire client’s remote procedure call, which can consist of several SQL queries, it can not provide accurate data about the amount of queries and their execution time. Despite this fact the statistics allows to find out some notable trends since the most of remote procedure calls consist of a single SQL query.

At first, the graph of clients’ data size distribution was built to better understand the characteristics of the application’s user base, that currently consists of about 40 000 clients. The most significant part of this graph is shown in Figure 2.

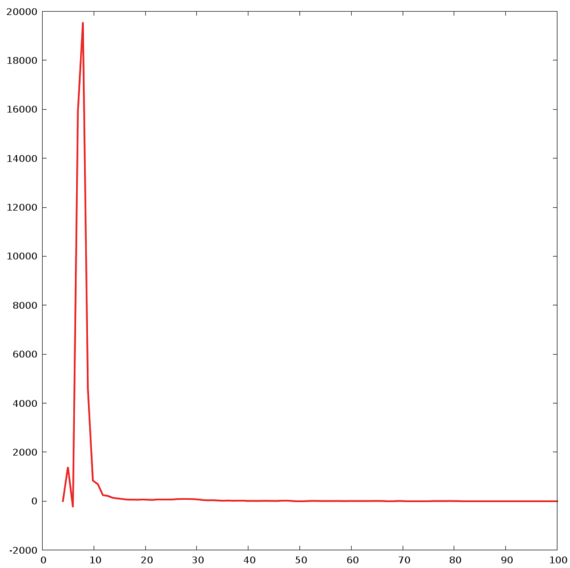


Figure 2: Client data size distribution.

The values on the horizontal axis correspond to the size of client’s database schema in megabytes. The values on the vertical axis correspond to the amount of clients with that data size.

Here we can see that most clients of the application being studied (about 37000 of 40000) have a schema with 7-8 megabytes of data. The curve of clients data size distribution basically corresponds to the curve of lognormal distribution This means, that if we want to visualize dependency between the client data size and some other characteristics, it is better to use the logarithmic scale.

To verify the correctness of assumptions 1 and 2, the data set containing 40 000 records with the format (client data size, total amount of queries for a week) was built and analysed. The graph that visualizes this data set is given below in Figure 3. The values on the horizontal axis correspond to the size of client data. The horizontal axis has the logarithmic scale and every next interval is about 1.5 times longer than the previous one. The values on the vertical axis correspond to the average amount of queries from clients that fall into the corresponding interval.

The analysis of the above data set confirmed assumptions 1 and 2 about the relationship between the data size and the client activity: the coefficient of correlation between the size of the client's schema and the number of client queries is about 0.7, which indicates a fairly strong correlation between these two values.

At the next stage the assumption about the dependency between the average query execution time and the client data size was studied. Another data set with the format (client data size, average query execution time for the last 24 hours) with 7500 records was built to confirm or reject this assumption. According to that data set, the coefficient of correlation between these two values equals to 0.03, that corresponds to a very weak dependency or its absence. This allows to throw away Assumption 3 as not confirmed.

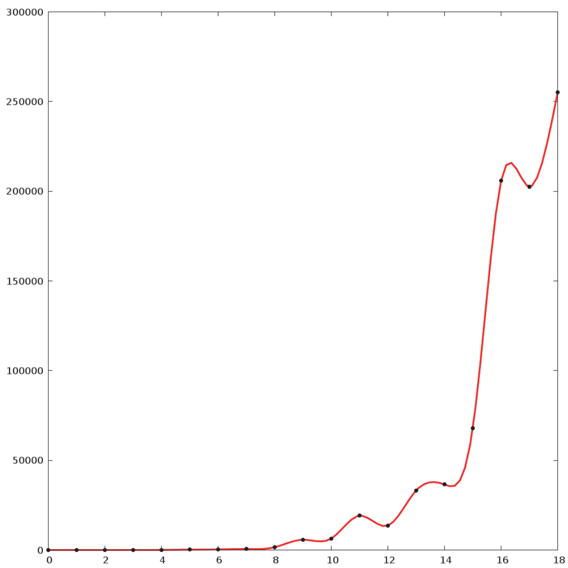


Figure 3: Dependency between the data size and the amount of queries.

The above data set also allowed to build a graph of average query execution time distribution (Figure

4). The values on the horizontal axis correspond to an average query execution time in milliseconds, and the values on the vertical axis correspond to a number of clients with that average query execution time. The resulting curve basically corresponds to the curve of lognormal distribution.

Another interesting point about the cloud application is the dynamics of users base grow.

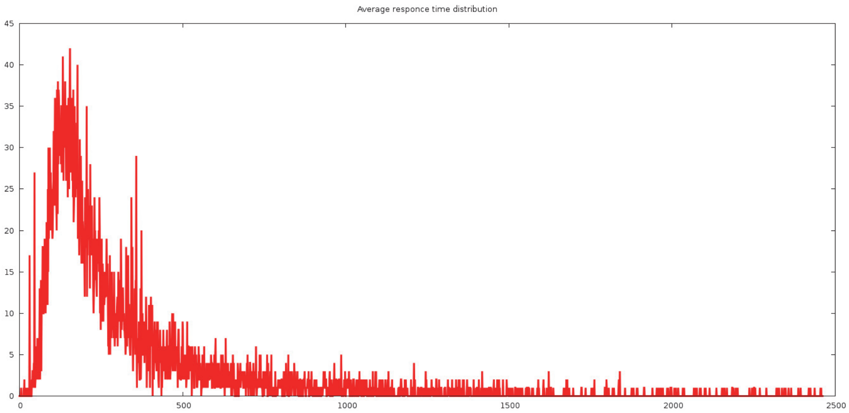


Figure 4: Query execution time distribution.

Unfortunately, the application studied is currently not very mature and goes through a phase of initial accumulation of a customer base. The dynamics of the total data size within the cluster is given in Figure 5, where the values on the horizontal axis correspond to a number of weeks and the values on the vertical axis correspond to the total size of client data in megabytes.

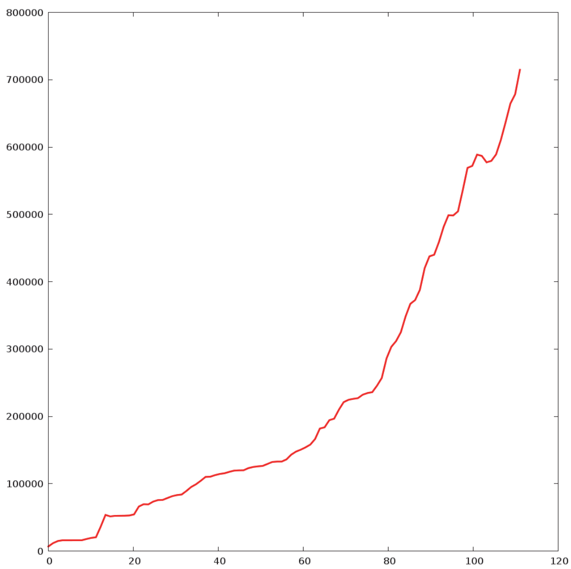


Figure 5: Total data size dynamics.

4 THE IMITATION MODEL OF THE MULTI-TENANT CLUSTER

The imitation model of the multi-tenant database cluster has been developed according to the analysis of the collected data. The model is a GUI application that runs under Linux OS. It was developed using Qt framework and C++ programming language.

The model works in virtual discrete time from one event to another. There are the following main entities that can produce events:

1. a query generator, including queries for creating of new tenants;
2. database servers inside of the cluster – query execution finish events;
3. a generator of failures – a special entity, which is used to simulate failures of cluster components.

Models of other cluster subsystems are implemented according to an "observer" design pattern. That means that they are notified by the model core about all occurring events and are able to generate derivative events (for example, data replication queries after finishing the execution of primary data modification query at the client's master server). A data distribution algorithm, in particular, is one of such observers that allow it to collect some statistics and analyse cluster performance for further optimization of its operation.

The Poisson flow of events is the basic model used for the query generator. The flow intensity can be set in two ways: as a static value, representing the amount of queries per unit of model time or as a dynamic value depending on the amount of clients (for example, 0.05 queries per client per unit of model time). When the generator operates in the second mode, the total intensity of the flow is equal to the product of the total amount of clients by the value set that is it changes with a rise of the clients base. Every new query is supplied with a weight coefficient that affects the query execution time by the server. According to the above statistics these coefficients are distributed by the lognormal law and distribution parameters can be changed during the generator setup.

Besides, the model is able to simulate server's failures and a denial of service. For that purpose there is a special entity – the failure generator. Every server inside the cluster has a non-negative probability of its failure per unit of model time. If this probability is not equal to zero, then sooner or later the server stops to serve queries for some time. The time of server's malfunctioning is determined by the distribution with parameters set by the user. This feature allows to research the cluster for fault-tolerance.

A lot of graphical reports on the cluster operation can be generated by the model (an average response time for the last 100 queries, a queue size at each server and a total queue size, a query flow distribution between servers and so on). These reports can be used to monitor a state of the model and to analyse trends in operation of the cluster.

There is also a special mode of mass parallel simulation to collect statistics about characteristics of algorithms tested. When this mode is used, the predefined configuration of the cluster and the profile of query flow is tested in combination with the offered algorithms of query routing and data distribution. There are several special stop conditions that are used to indicate the completion of modelling (a period of time, a specified queue size at any server, a specified total queue size at the level of the entire cluster). Many identical experiments run in parallel. A complete statistics about the state of the cluster and the combination of the used algorithms are stored in resulting files by the model.

5 CONCLUSIONS

Some primitive algorithms of data distribution and query routing were developed for the model (the balancing of data size or the amount of clients per server). The early experiments with the model were devoted to analysing the efficiency of these algorithms. The study has shown that the usage of primitive strategies sometimes leads to the formation of query queues at some servers inside the cluster even when the actual intensity of the flow is less than the theoretical throughput. This effect is caused by a burst of client activity for some period of time. The formation of queues leads to a degradation of application performance from the point of view of the client whose data are stored by these servers. The above fact means that primitive strategies should not be used in the real production environment. Further study will be devoted to the model improvement and identification of key factors that influence the efficiency of the proposed cluster control system.

REFERENCES

Chong, F., Carraro, G. (2006a). *Architecture Strategies for Catching the Long Tail*. Microsoft Corp. Website.

Chong, F., Carraro, G., Wolter, R. (2006b). *Multi-Tenant Data Architecture*. Microsoft Corp. Website.

Candan, K.S., Li, W., Phan, T., Zhou, M. (2009). *Frontiers in Information and Software as Services*. In Proc. of ICDE, pages 1761-1768.

Schiller, O., Schiller, B., Brodt, A., Mitschang, B. (2011). *Native Support of Multi-tenancy in RDBMS for Software as a Service*. In Proc. of the 14th International Conference on Extending Database Technology.

E.A. Boytsov, V.A. Sokolov (2012). *The Problem of Creating Multi-Tenant Database Clusters.* In Proc. of SYRCoSE Conf., pages 172-177.

SPECIAL SESSION ON e-HEALTH SERVICES AND TECHNOLOGIES (EHST)

Health Information Services using Finger Plethysmogram

Mayumi Oyama-Higa[1], Tiejun Miao[2] and Shigeki Takada[3]
[1]Chaous Technology Research Lavolatory, 6-26-5 Seta, Otsu, Shiga, Japan
[2]TAOS Institute Inc, 1-7-8 kaigann,Minatoku, Tokyo, Japan
[3]Kwansei Gakuin University 1-1-155 Uegahara, Nishinomiya, Hyogo, Japan
mhiga@chaotech.org, info@itaos.org, takada@kwansei.ac.jp

Keywords: Vital Signs, Mental Wellness, Mental Disorders, Autonomic Nerve Balance, Pulse Chaos, Nonlinear Dynamics, Information Complexity, Plethysmograms.

Abstract: The goal of this research is to develop a system that stores and displays visual analysis of measurements taken from pulse waves at the fingertip so that anyone can check their mental state including past information at any time and at any location. Furthermore, there is also the need to fully utilize the power of digital networks so that people and the people surrounding them are aware of such mental states.

1 INTRODUCTION

Up to now, it was said that living organisms must maintain homeostasis to maintain life. It was also thought that living organisms are capable of maintaining homeostasis due to the automatic control systems of the negative feedback mechanism. When applying vital signs that are directly associated with our lives such as the heart rate, respiration, blood pressure and body temperature to the concept of homeostasis, it can be said feedback is triggered to compensate any disturbance that causes certain values to deviate from their normal values, and the more stable these values are the more efficient the control systems of the living organism are functioning. But the heartbeat of a healthy person, for example, is never constant even if the person is in a relaxed state of mind. On the contrary, it fluctuates quite irregularly (heart rate variability). This also applies to the respiration, blood pressure, body temperature, etc. In fact, we know that there is less fluctuation of the heart rate among the elderly and individuals with medical conditions. The same can be observed in pulse waves. For this reason, there were continuous reports in the field of physiology around the mid-1980's indicating the possibility that such fluctuations including heart rate and brain waves are chaotic. Because no new knowledge could be gained when using the conventional method of linear analysis to analyze chaotic fluctuations, there was the need to analyze chaotic fluctuations using nonlinear methods. Recent advancements in computer processing speed and visualization capabilities have allowed us to analyze nonlinearly the chaotic properties of vital signs. Such technologies have opened new doors of understanding concerning information that was treated as error or simply unknown in the past to actually contain information that we wanted to know most.

Psychologists in the past have tried to examine the mental state of people through trial and error by asking a series of questions as there was no way to read a human mind. But if accurate information can be obtained by directly measuring vital signs and performing nonlinear analysis, such information should be greatly effective in the field of psychology that relies on the rules of thumb. Those involved in brain research conducted large-scale experiments such as directly inserting electrodes into the brain or taking video images based on the principle that all information can be found in the brain. However, we can now check with relative ease the various states of the mind by examining the pulse waves at the fingertip which contains information of the central nervous system. We are now able to assess the state of the sympathetic and parasympathetic nerves from pulse waves taken from finger plethysmogram. Combined with the information gained from nonlinear analysis of pulse waves we can also obtain other types information such as moods, etc. Such understanding was gained through many psychological and biological tests. In recent years, there has been an increase in the number of suicides resulting from depression as well as people causing social problems as the result of becoming mentally "high." What kind of mental state are they in? We believe that a measurement device that allows us to check the various mental states of ourselves would contribute, to a certain degree, a safer and peaceful society. We also believe the need to develop a system so that people can check themselves in order to handle major issues in the increasingly complex human society such as how to rejuvenate people mentally incapable of fitting into society in the aging society where one in every four will be 65 years old or older, how to detect and deal at an early stage the bullying of children that has become a serious problem of communal life, etc.

2 MEASUREMENT AND CHAOS ANALYSIS OF VITAL SIGNS

2.1 Vital Signs and Chaos

From the day we are born to the day we die, we humans continuously emit vital signs that fluctuate dynamically. Complex fluctuations are everywhere, including macroscopic fluctuations of life activities, the fluctuations of the heart and blood pressure, and the microscopic fluctuations on a molecular level. Such fluctuations, however, are neither constant fluctuations created mechanically nor fluctuations that are completely random. Living organisms fluctuate chaotically. Often times, chaos and random are interpreted to be the same. Unlike random, however, chaos has deterministic rules as shown in the diagram below. There are a number of ways to check whether a certain fluctuation is random or chaotic. One of these methods is to draw an attractor. Fig. 1 shows one of the methods for checking whether fluctuating data is random or chaotic. In the natural world, random and chaotic fluctuations exist outside constant, regular fluctuations. And it has already been established that pulse waves possess chaotic properties.

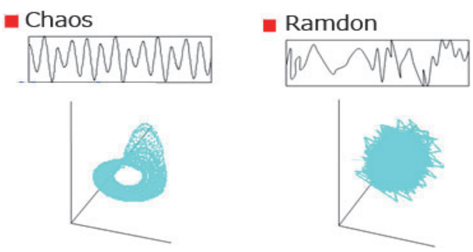


Figure 1: Difference between chaos and random shown using an attractor.

2.2 Measuring Pulse Waves from the Fingertip

As shown in Fig. 2, pulse waves from a finger is taken by measuring the increase and decrease of hemoglobin flowing through the capillaries at the fingertip using an infrared sensor and then converting the obtained analog information to digital data for use in calculation. Other than a fingertip, the sensor can also take measurements from an earlobe or even a toe. However, the sensitivity of the left-hand fingertip is especially suitable for measurement to synchronize with the blood flow from the heart.

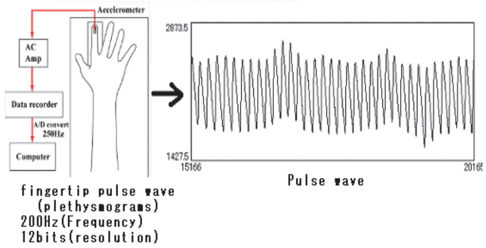


Figure 2: Measuring fingertip pulse waves.

2.3 Chaos Attractor and Analysis of LLE (Largest Lyapunov Exponent)

In order to create an attractor from fingertip pulse waves (hereafter just "pulse waves") , embedding dimensions d and delay time (time delayed for embedding) τ must be determined using Takens' embedding theorem. A good attractor cannot be drawn unless an appropriate value of τ is selected. If τ is too small, the value before delaying time τ and the value after delaying time τ will be almost the same, and the values will no longer be independent as the correlation is too strong. If τ is too large, phase relation information is lost as there will be no statistical correlation. Hence, there is the need to select the optimal delay time. Delay time is determined by continuously calculating nonlinear average mutual information (cross-correlation coefficient and delay time) to first find the smallest value of τ.

Next, embedding dimensions d is found by incrementally increasing the number of dimensions starting from two using the G-P algorithm (correlation dimension method) until number of correlations within the attractor stops increasing. The trajectories of an attractor fluctuate along with time. Such fluctuation is referred to as the largest Lyapunov exponent, or LLE. In our research, we conducted various psychological experiments by focusing our attention on the LLE. In order to assess what kind of information can be obtained from LLE, we created a mathematical model and conducted an experiment using general anesthesia. As a result, we were able to verify that LLE contains information of the central nervous system.

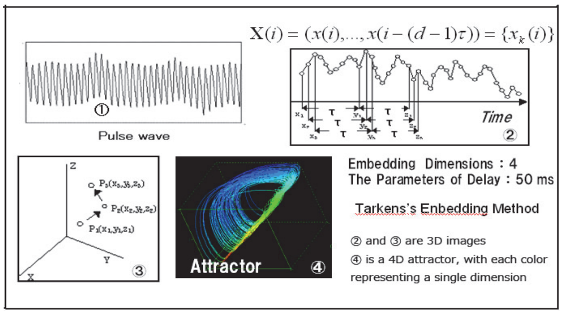


Figure 3: Method of creating an attractor from pulse wave data.

Fig. 4 shows the method for finding the largest Lyapunov exponent by calculating the fluctuations of the trajectories of an attractor over time.

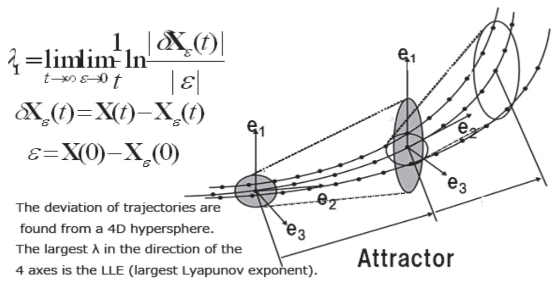


Figure 4: Finding the largest Lyapunov exponent (LLE).

The LLE representing the instability of trajectories of an attractor is found by calculating the LLE from the attractor structured by 3,500 points, delaying 200 points, calculating the LLE structured by the next set of 3,500 points and repeating the process until pulse wave data ends. 43 Lyapunov exponents are calculated from one minute of measurement data consisting of 12,000 points. One LLE is calculated in the first 17 seconds and then one every second thereafter. The average LLE found from the total time of measurement and standard deviation are used as assessment values in analysis.

2.4 Information That Can Be Acquired from Pulse Waves

We have discussed that LLE gain be obtained from nonlinear analysis of pulse waves. But there was also the need to check what that information was telling us. We conducted a simulation using a power spectrum by synthesizing waves that incorporate various conditions including blood pressure, heart rate and respiration transmitted from multiple parts obtained from the biological model shown in Fig. 5 and pulse waves. When running a simulation by entering formulas for the central nervous system, we found that the waveform of the mathematical model resembles the waveform created in the measurement test. This suggests that the mathematical model of pulse waves contain information of the central nervous system.

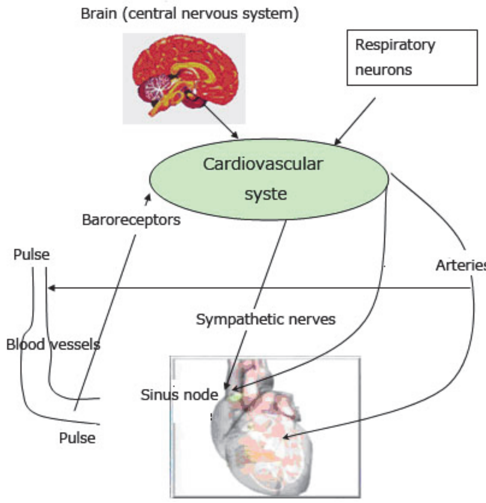


Figure 5: Mathematical model for simulating pulse waves.

The mathematical model was also verified in the experiment which examined the state of LLE during general anesthesia. If LLE contains information of the central nervous system, there should be a drop in the LLE when inducing general anesthesia. Fig. 6 shows the state of LLE during general anesthesia from the start to the end of surgery of a patient diagnosed with rectal cancer. Although the fluctuation of LLE does not drop to zero since the heart is moving, there is a gradual decrease in the LLE at the start of general anesthesia. During general anesthesia, LLE drops drastically. And upon recovery, LLE starts to rise again.

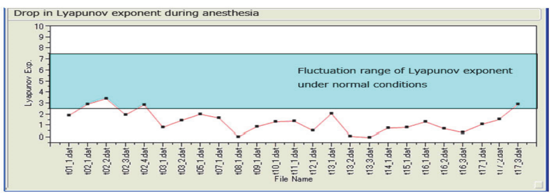


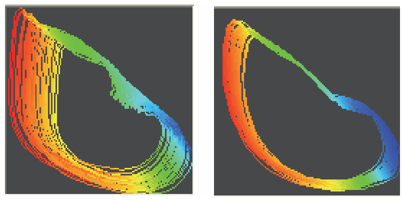
Figure 6: Changes in LLE during general anesthesia.

We were able to verify from the mathematical model and the general-anesthesia experiment that pulse waves are affected by the central nervous system.

3 RELATIONSHIP BETWEEN LLE AND COGNITIVE PSYCHOLOGY

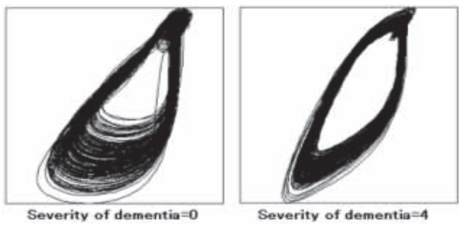
Up to now, changes in the mental state of a human mind was never assessed numerically using biological information. Focusing on LLE obtained from nonlinear analysis of pulse waves, we conducted various experiments to study the relationship between LLE and dementia of the elderly, LLE and communication skills from view of the ADL index, LLE and error rate during work, LLE and daily variations of an employee as well as the cumulative fatigue index, LLE and changes in fluctuation over time between ages zero and five, LLE and the effects of a mother's affection on children, etc. The results have allowed us to gain understanding that LLE is closely associated with the things we humans need to maintain a healthy state of mind including external adaptation capabilities regarding the environment and society, flexibility of the mind, self-motivation and harmony. LLE that defines the fluctuation of the trajectories of an attractor can be defined as chaotic fluctuations. In other words, a continuously low LLE, or prolonged state without fluctuation, can be defined in everyday life as a drop in the power to adapt to the outside world. On the contrary, continuously high LLE and large fluctuation suggest continuous extreme tension or stress, preventing one from maintaining a healthy mental state. For human beings, a healthy state is a state with constant fluctuation. We also believe that human emotions cause change in the fluctuation. Let's look at a healthy mental state in contrast to physical immune strength. Normally, physical immune strength is said to be vital to maintain health. We human beings need physical immune strength to maintain our health. Drop in the immune strength can lead to various illnesses. In order to prevent this drop in physical immune strength (vitality), we eat carefully, rest, take medications and exercise to build up tolerance. On the other hand, what state defines the mind as healthy? Mental strength, such as the ability to communicate actively, motivation to live and the ability to tolerate the drastic changes in the outside world, is something extremely vital for the survival of mankind. If this is mental immune strength (vitality), there was no way to examine it using a scientific approach. Although mental immune strength is related to the vitality of human beings, it is basically a state of high or low and strong or weak. The mental immune strength is flexible and fluctuates constantly. A healthy state of mind is the ability to flexibly adapt to external changes with fluctuation. In other words, fluctuation of the LLE over time is critical to maintain mental health.

Fig. 7 shows an attractor of a mentally healthy person and an attractor of a depressed patient. Notice that the fluctuation of the attractor of the depressed patient is extremely small. Fig. 8 shows an attractor of a patient with dementia. Both attractors were drawn using data taken from an elderly. It is clear that the fluctuation decreases as the severity of dementia increases.



Attractor of healthy subject | Attractor of depressed patient

Figure 7: Attractors of a healthy person and a depressedpatient (30-second measurement).



Severity of dementia = 0 | Severity of dementia = 4

Figure 8: Elderly data.

A continuously high state of LLE can also be observed during daily life when exposed to extreme tension or stress. A mentally healthy person can naturally relax after continuous exposure to extreme tension. This is because such person can lower the LLE to restore the fluctuation to a natural state. A person suffering from depression or an elderly with advanced dementia, on the other hand, will show a continuously low state of LLE. External adaptation is impossible in such state. In such case, there is the need for the person to examine his or her own changes in the LLE, learn from past states of LLE and allow the LLE to fluctuate by discovering methods that are effective or communicating with surrounding people. It is important to know yourself before proceeding with hospitals and medical treatment.

4 RELATIONSHIP BETWEEN LLE AND MENTAL HEARTH

4.1 Analyzing the Severity of Dementia and Communication Skills using Chaos Analysis of Pulse Waves Taken from the Elderly

Subjects: Measurements were taken from 179 patients (male: 40, female: 139) at three nursing homes in Shiga Prefecture.

Measurement period: August – November 2003.

Measurement method: Three measurements of three minutes each were taken using finger plethysmogram. Measurements were taken while maintaining the subjects in a relaxed state in a room set at 25°C. Prior to the measurement of pulse waves, the maximal blood pressure, minimal blood pressure, pulse and body temperature were taken.

Index: The relationship with LLE calculated from pulse waves was examined by utilizing data indicating the severity of dementia in five stages determined by a physician and ADL (3-level assessment) data consisting of seven items created by the care taker.

Results: Significant relationship was observed between LLE and severity of dementia, as well as between LLE and communication skills.

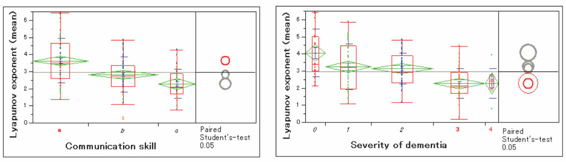


Figure 9: Relationship between LLE (vertical axis) and communication skills (left graph), and between LLE and severit of dementia (right graph). (Communication skills: 3 levels of a, b, and c; severity of dementia: 0 – 4).

The graph on the left shows significant drop in the LLE as the level of communication skills decreases. The graph on the right shows significant drop in the LLE as the severity of dementia progresses.

Fig. 10 shows the results of measurements taken nine months following the first set of measurements. Results varied from patients having higher LLE than the first time to those with less LLE. From the results, we were able to confirm that the value of LLE fluctuates constantly. However, the patient that passed prior to the second measurement had the lowest LLE among the patients during the first measurement. Is this an indication of something significant? The results are deeply concerning.

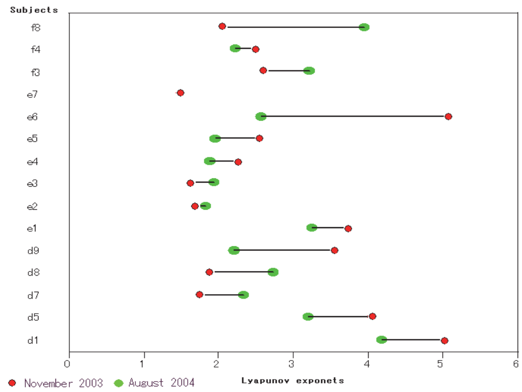


Figure 10: Results of LLE measurements taken nine months after.

4.2 Relationship between Changes in LLE of Children and the Mother's Affection

Subjects: 242 children between zero and five years of age at daycare centers in Osaka and Himeji.

Measurement period: January 2004 – March 2005.

Measurement method: Two measurements of a minute each were taken using finger plethysmogram. Measurements were taken while maintaining the subjects in a relaxed state in a room set at 25°C.

Results: The LLE of children between zero and five is lower at the age of three when compared to the other ages. The results of verification show a significant relationship with a probability of 0.05%.

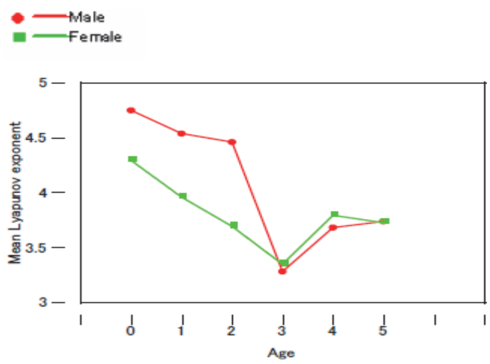


Figure 11: Changes in LLE of children by age (242 children).

The diagram show that the LLE is at its highest at the age of zero, followed by one and two, with three having the lowest value among all ages. The myth of the first three years has raised a question about the age of three as a global theme. It is highly significant that we were able to scientifically observe the trend using the LLE taken from pulse waves

4.3 Relationship between the Pulse Waves of Company Employees and the Fatigue Index

Depression among employees is becoming a social problem. We conducted an experiment examining the relationship between the LLE of company employees during the day and the fatigue index. From the fatigue index obtained through a series of questions, we were able to conclude that the drop in LLE was caused by "depressive state" and "anxiety." The results are shown in Table 1. Note that "anxiety" and "depressive state" show a negative correlation of -0.7 or higher when compared to the LLE during work. In other words, low LLE during work suggests a depressive state or high anxiety.

Table 1: Relationship between LLE of employees during the day and the fatigue index.

	Drop in willpower	Degree of anxiety	Depressive state	Accumulation of fatigue	Lyapunov exponent during the day
Drop in willpower		0.7235	0.7539	0.7496	-0.6385
Degree of anxiety	0.7235		0.8455	0.9358	-0.7279
Depressive state	0.7539	0.8455		0.842	-0.7279
Accumulation of fatigue	0.7496	0.9358	0.842		-0.6305
Lyapunov exponent during the day	-0.6385	-0.7279	-0.7014	-0.6305	

4.4 Relationship between LLE and Judgment and Operational Errors during Monitoring Work

In order to conduct an experiment on human error, we developed a device that creates a virtual environment for performing monitoring work on a computer. In this experiment, we examined the relationship between LLE and the error rate. The results showed the low LLE causes an increase in the error rate.

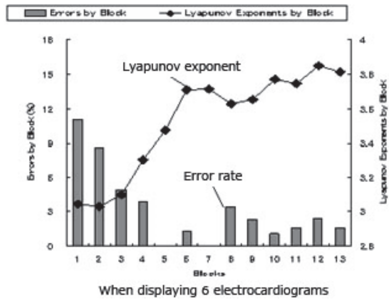


Figure 12: Relationship between LLE during monitoring work and error rate.

4.5 Changes in LLE When Giving Birth

Fig. 13 shows the results of examining the changes in LLE of seven pregnant women before and after giving birth (maternity clinic in Nara-shi). The LLE within 90 minutes of giving birth and the LLE within 90 minutes after giving birth were compared. The LLE prior to giving birth is significantly high

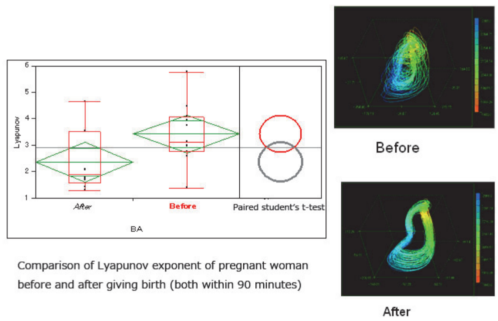


Figure 13: Comparison of LLE before and after giving birth (both within 90 minutes).

4.6 Relationship between LLE and Laughter

It is often said that laughter is the best medicine. The diagram below shows the changes in LLE when watching and not watching a comedy video for five minutes.

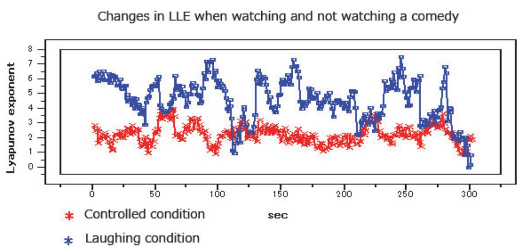


Figure 14: Changes in LLE when watching and not watching a comedy video.

From the various examples we have observed, we can summarize the LLE of a mentally healthy person as follows:

- The LLE of a mentally healthy person fluctuates constantly within a certain range. Furthermore, the LLE changes unconsciously.
- The LLE of a person suffering from depression or dementia is continuously low.
- Continuously high LLE indicates extreme tension and stress, and at risk of losing mental balance.

From the above, we can say that the mental state cannot be determined with single measurement of LLE.

5 NECESSITY OF SELF-CHECK SYSTEM FOR MENTAL HEALTH

5.1 Social Needs and Cautions concerning Measurements

There are said to be more than 30,000 suicides per year in Japan. Although depression is not the only cause, depression is often times accompanied by an alternating cycle between depressive state and manic state. Severe manic state triggered by medication is said to be a cause for suicides. We believe that knowing your own state by measuring pulse waves as a means of self-control can be effective in preventing suicides.

We know that Japan is on the way to an aging society, where one in every four persons will be 65 years old or older in 2025. Some of the issues of aging are enormous medical costs and nursing costs that ultimately affect the lifestyles of individuals. What can we do to keep working energetically even when we age, or to make sure we do not put a burden on our family with dementia? These are all issues that we must take seriously. Currently, we are examining the effects that animal-assisted therapy, music therapy and life review have on communication and motor skills. We are also conducting experiments on LLE to see which methods are effective for rejuvenating the elderly. However, methods for improving the LLE will differ for each individual. But everyone is capable of improving their LLE. We believe that it is up to the individual to discover the best method.

Judging from the above, we decided the need to develop a self-check system so that anyone can measure their LLE at any time and at any location. We developed a software program called Lyspect that measures not only the LLE from the pulse waves measured at the fingertip, but the state of sympathetic nerves, parasympathetic nerves and autonomic nerves, and the health of blood vessels. But in order to check past measurement records and to check your mental state based on the feedback of such information, there is the need for a database and the use of the Internet.

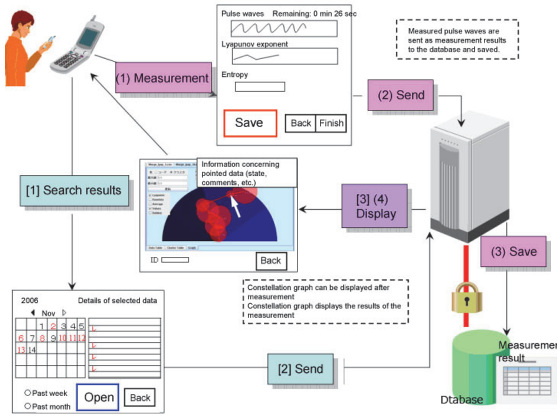


Figure 15: Image of performing finger plethysmogram using a cell phone or smartphone, performing nonlinear analysis to calculate biological information such as the LLE, and saving the data in database and loading past records.

6 INTRODUCTION OF LYSPECT

We developed a software program capable of calculating and displaying the following by measuring pulse waves. Lyspect is capable of analyzing and display the three types of values with pulse waves as input data.

Chaos analysis (calculation of LLE), vascular balance analysis, autonomic nerve balance analysis. The program is also capable of displaying LLE and HF/LF in real time by connecting a sensor.

There are two types of Lyspect: the original Lyspect that can be used for research and Lyspecting, a simple version of the original.

Fig. 16 shows the results of five measurements taken from a single person using Lyspecting, the simpler of the two.

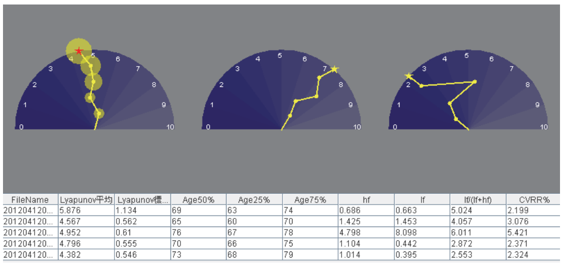


Figure 16: Results displayed by Lespecting (5 measurements).

Fig. 17 shows the results of three measurements taken from a depressive patient. The LLE is constantly low and the autonomic nerve balance indicates that the sympathetic nerves are superior.

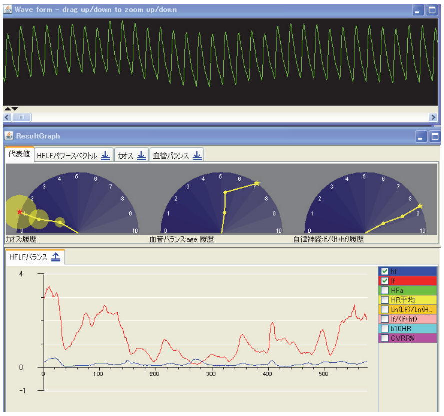


Figure 17: Results of depressive patient displayed by Lyspect.

We have also developed a software program that can be operated on android smartphones.

7 INTRODUCTION OF LYSPECT

We verified the deep relationship between human emotions and LLE calculated based on nonlinear analysis of the micro-fluctuations in pulse waves that contain chaotic properties by creating mathematical models and conducting experiments using general anesthesia. We have also developed a software program for analysis. Our challenge for the future is to develop a pulse wave sensor that any can easily use and afford. In order obtain data using a cell phone or smartphone, there is the need for the sensor to be small and light. Although we have succeeded in realizing a wireless and USB connection, we are requesting others to develop a sensor that is compact and user friendly.

There is also the need to address security issues as biological information is handled. We believe we can resolve this issue by selecting the appropriate database management method and communication method.

REFERENCES

Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE'06, 1st International Conference on Template Production*. SciTePress.

Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

Tsuda I., Tahara T., Iwanaga I., 1992. Chaotic pulsation in capillary vessels and its dependence on mental and physical conditions. Int J Bifurcation and Chaos 2: 313-324.

Sumida T., Arimitu Y., Tahara T., Iwanaga H., 2000. Mental conditions reflected by the chaos of pulsation in capillary vessels. Int J Bifurcation and Chaos 10: 2245-2255.

Sano M., Sawada Y., 1985. Measurement of the Lyapunov spectrum from a chaotic time series. Phys. Rev. Lett. 55: 1082.

Abarbanel HDI., Brown R., Sidorowich JJ., Tsimring LS., The analysis of observed chaotic data in physical systems. Rev Mod Phys 1993., 65: 1331-1392.

Tokihiko Niwa, Kenji Fujikawa, Yoshikazu,Tanaka, Mayumi Oyama, 2001. Visual Data Mining Using a Constellation Graph, ECML/PKDD-2001, Springer-Verlag. (Academic Journal, 2001.) Working Notes/29-44.

Oyama-Higa M., Miao T., Mizuno-Matsumoto Y., 2006., Analysis of dementia in aged subjects through chaos analysis of fingertip pulsewaves. 2006 IEEE Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, 2863–2867.

Takens F., In: Braaksma B. L. J., Broer H. W., Takens F., eds. 1985. Dynamical Systems and Bifurcations, Lecture Notes in Math. Vol. 1125. Springer, Heidelberg.

Miao T., Shimoyama O., Oyama-Higa M., 2006. Modelling plethysmogram dynamics based on baroreflex under higher cerebral influences. 2006 IEEE Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, 2868–2873.

Oyama-Higa M., Miao T., 2006. Discovery and application of new index for cognitive psychology. 2006 IEEE Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, 2040–2044.

Imanishi A., Oyama-Higa M., 2006. The relation between observers' psychophysiological conditions and human errors during monitoring task. 2006 IEEE Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, 2035–2039.

Relationship between Food Intake and Finger Plethysmograms

Yoko Hirohashi, Masaki Chiba, Sang-jae Lee and Atsuko Umezawa
Faculty of Health and Welfare, Nayoro City University, W4-N8-1, Nayoro, Japan
{hirohashi, machi, sj0729, umezawa}@nayoro.ac.jp

Keywords: Food Intake, Finger Plethysmograms, Non Liner Analysis, LLE, Autonomic Nerve Balance.

Abstract: The purpose of this manner of research was to investigate the influence of food intake on the human body, by means of data showing the chaotic fluctuations of fingertip pulse wave, through the use of nonlinear analysis. For the measurement of the fingertip pulse wave, 5 types of meals were utilized, using both meals where a single person ate alone with no conversation, and meals where several persons ate and conversed, using water, hot soup, and cold soup, measurements were taken at various times (just before food intake, just after food intake, and 30 minutes after food intake). At result, with meal intake, there was a temporary and significant drop in LLE levels, followed by a rise in LLE when several persons ate and conversed. Most of the participants saw little change in levels after intake of cold soup, or water. With intake of hot soup, it was observed that a little climb in LLE levels 30 minutes after eating. Many people increased levels from heat stimulation of the sympathetic nerve, then afterward a decrease, and there is speculation that there is a kind of "relaxation effect". It seems evident that partaking of warm food while in the company is a good practice.

1 INTRODUCTION

In regard to meals, one's diet has a physical and emotional effect on maintenance of life activities, maintenance and quality of health, recovery from and prevention of disease, enjoyment of life, a sense of contentment, seasons of life, etc. Also, in societal relationships, communication and enjoyment of one's family, cultural traditions, cultural aspects regarding ceremonial occasions, etc., diet has a large role and meaning, and is an essential part of daily human life.

In recent years, in dual-income families, generally speaking, in relationship to daily lifestyle in society, eating alone, always eating the same food, unbalanced diet, dependence on "fast food", etc., has become a real problem. Because of this, in kindergartens, elementary schools, and junior high schools, there has been an increased realization of the importance of "nutrition education". In a survey (Shinohara 2012), where there is a great deal of instruction regarding proper diet, the effects of "concern for good eating", "manners", "rhythm of daily life", "proper body weight", "proper bowel movement", etc., can be clearly seen. Where children are given good instruction in nutritional education, the importance of good diet has become very evident. Outside of activities to promote nutritional education, there has been an expansion of research in the following areas: studies involving the effect of food aroma (Kunieda 2011), research in improvement of dietary habits (Fujita 2012), studies regarding eating/swallowing disorders in nursing care facilities (Okada 2009).In addition, when measuring levels of Chromogranin A found in saliva, research showing the correlation between diet and relief from stress, as well as levels of Cortisol found in the bloodstream and levels of active NK cells reveal the level of enjoyment of meals, research into the influence of eating and exercise on the value of oxidation stress markers, and information regarding the value of diet on the living body can be observed.

However, regarding the relationship of diet and brain activity, various research has been conducted where brain waves are measured in connection with food aroma (Sumiya 2007, Ishiguchi 2008), but there are only a few articles that deal with prevention of dementia in elderly persons by controlling diet, and research using information regarding how diet itself affects brain activity in living persons is lacking. By analyzing the fluctuation of the fingertip pulse wave, data regarding the effect of the Largest Lyapunov Exponent (LLE) index upon brain activity, as well as information related to balance of the autonomic nerves can be obtained. By obtaining information on how eating cause fluctuation of the fingertip pulse wave, more understanding can be gained as to the influence of diet upon the human body. Knowledge regarding the meaning of human lifestyle and diet, obtained by data collected from these kinds of measurements is thought to be both objective and meaningful. Also, with Oyama and others, based on the outcome of advances in research by observing fluctuation of the fingertip pulse wave, information regarding brain activity has been newly obtained, and comparisons of fingertip pulse wave and its correlation to food intake have been observed.

The purpose of this manner of research is to investigate the influence of food intake on the human body, by means of data showing the chaotic fluctuations of fingertip pulse wave, reflecting variations of brain activity, through the use of nonlinear analysis technique.

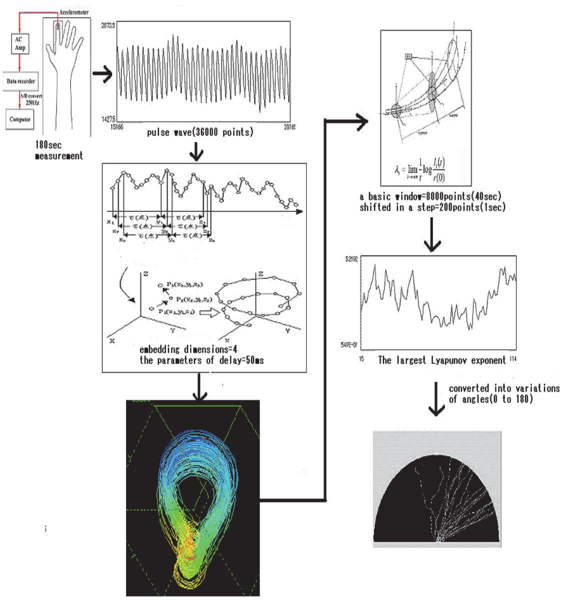


Figure 1: Flow diagram showing the procedure from the measurement of pulse waves to calculation of the Lyapunov exponent.

2 METHOD OF STUDY

2.1 About a Finger Pulse Wave and Calculation of Lyapunov Exponent

By means of nonlinear analysis of data from fluctuations in fingertip pulse wave, it is possible to obtain a numerical value of "mental revitalization", using the Largest Lyapunov Exponent index value. Using advanced research methods, effectiveness of various activities directed at mental health can be verified. Also, using data gathered from fluctuations of fingertip pulse wave, it is possible to determine autonomic nerve balance (whether the sympathetic nerve or parasympathetic nerve is dominant).

Figure 1 shows the flow diagram showing the procedure from the measurement of pulse waves to calculation of the Lyapunov exponent (Oyama 2006). To construct the attractor, we set a delay time and the number of embedding dimensions according to Tarkens theory. We used four embedding dimensions and a delay time of 50 msec. The figure on the right illustrates the method of embedding in three-dimensional phase space. Although effective information can be obtained from the shape of the four-dimensional attractor, we calculated the Lyapunov exponent, which is an index of trajectory instability and has a chaotic characteristic. (Figure 1)

By measuring the fingertip pulse waves for one minute, 43 Lyapunov exponents are obtained. We compared each condition using an average of these values (Oyama 2007).

2.2 Method of Measurement

2.2.1 Study Subject

For this research, 20 students currently enrolled between the 2nd and 4th year of studies at "N" University were used. All of the participants were female, from 20 to 22 years of age. All were in generally good health, with no known mental or physical abnormalities. Before sampling fingertip pulse wave of the participants, vital signs (body temperature, heart rate, blood pressure) were measured each time, and no one showed any abnormal condition. The number of times the experiment was conducted varied according to food intake, from 1 to 5 times. In order to make comparisons regarding intake of water, hot soup, and cold soup, the same individuals were utilized.

2.2.2 Study Location

"N" University, located in "H"

2.2.3 Study Period

January, 2013

2.2.4 Measurement Detail

With each of the participants, personal data was collected and prior explanation of the experiment was given, and care was given to insure adequate ethical treatment of those involved. In addition, a consent form was obtained from each participant, documenting their willing cooperation with the research.

On the day of the experiment, each participant's vital signs (body temperature, heart rate, blood pressure) were measured, and verification was made that there were no signs of unusual physical abnormality, prior to commencement of the experiment.

For the measurement of the fingertip pulse wave, 5 types of meals were utilized, using both meals where a single person ate alone with no conversation, and meals where several persons ate and conversed, using water, hot soup, and cold soup, measurements were taken before and after eating. At various times (just before food intake, just after food intake, and 30 minutes after food intake), 3 different readings were taken. For obtaining the measurements, a chaotic Lyspect device was utilized. A cuff was used with the device, and placed on the tip of the participant's right index finger, and each time a measurement was taken for 3 minutes. The subjects were asked to sit in an armchair, and rest their arm quietly on a table for the duration of the measurement. During measurement, the subjects were asked to not speak and maintain a quiet, unmoving position. A plan was followed where in a single experiment 5 people were observed and their readings were taken. In accordance with the measuring devices, the beginning of each experiment was carried every 5 minutes.

For the environment used in taking the measurements, the subjects were asked to not communicate with one another, and a large, quiet room was used. The room temperature was 22 degrees C. The participants were seated separately in armchairs at tables, and after taking the pre-experiment readings were allowed to eat. After taking the reading immediately following the meal, the subjects were asked to sit quietly and look at picture books. In order to prevent any influence of written words on the body, picture books with no words made by artist Anno Mitsumasa were used exclusively.

The room where the subjects ate while communicating was comparatively small, but was free of noise and the room temperature was maintained at approx. 22 degrees C. The participants sat around tables, and enjoyed pleasant conversation while eating the meal.

The meal consisted of store-bought sandwiches and onion soup (150 ml.). The sandwich size was an amount such that the female college students were sufficiently full after eating. The hot and cold soups were store-bought creamed corn soup (180 ml.), and were consumed either after heating, or after taking directly from the refrigerator.

Analysis of the experiments was conducted using Lyspect software on computer, but data regarding fluctuation of fingertip pulse wave was analyzed immediately using nonlinear analysis technique.

3 RESULTS

3.1 Meal Intake

There were 10 participants who ate their meal alone, one person at a time. During the meal, there was no communication interaction with others. 5 individuals were 2 groups of 5 people, and the experiment was conducted 2 times.

7 participants ate a meal with pleasant communication. One group consisted of 5 members, but due to absence of some members, so the remaining groups worked in pairs, conducting the experiment 2 times.

3.1.1 LLE Index Values for Those Who Ate Alone

1) 4 participants saw elevated LLE index levels immediately following the meal. Of this group, 3 saw levels initially drop, and then rise again. And 1 individual saw the LLE level climb gradually.

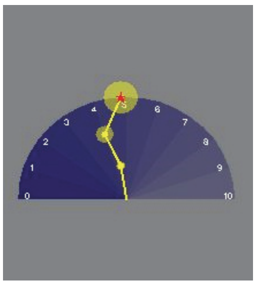


Figure 2: No.3's LLE.

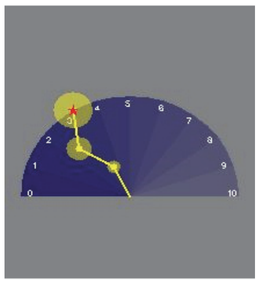


Figure 3: No.9's LLE.

2) 3 participants saw LLE index levels drop immediately following the meal. Of this group, 2 saw levels gradually drop, and 1 person saw their level greatly increase and then decrease.

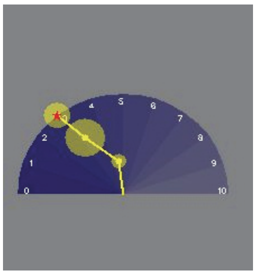


Figure 4: No.5's LLE.

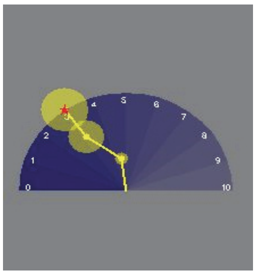


Figure 5: No.7's LLE.

3) 3 participants saw either no change or only very slight changes in their LLE index level.

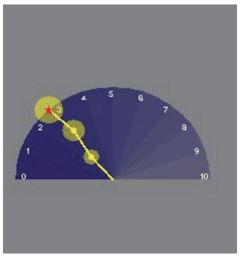


Figure 6: No.6's LLE.

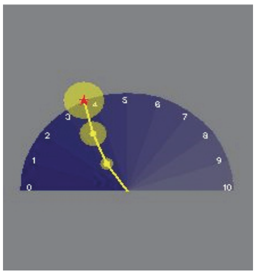


Figure 7: No.10's LLE.

3.1.2 LLE Index Values for Those Who Ate While Visiting with Others

1) 4 individuals saw LLE index levels increase after the meal. Of this group, 3 people saw their levels increase after temporarily seeing a sharp decrease.

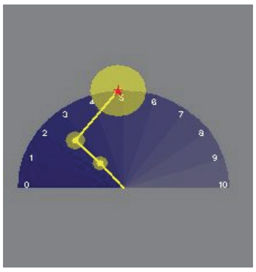


Figure 8: No.16's LLE.

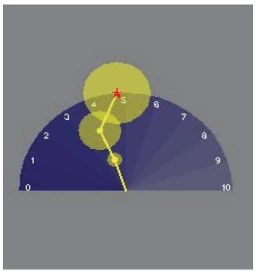


Figure 9: No.23's LLE.

2) 1 person showed almost no change in LLE index levels after the experiment. This person experienced a temporary rise and fall of the level, with the final level nearly equal to the original level.

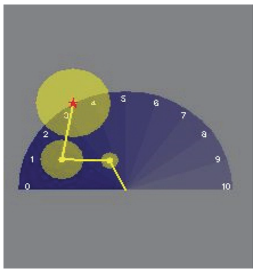


Figure 10: No.15' s LLE.

3) 2 participants saw LLE index levels drop. However, they saw a large drop and then a large increase in the levels, and while there was a great fluctuation, the final level was nearly the same as before the experiment began.

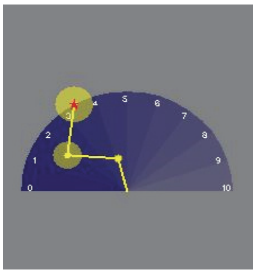


Figure 11: No.13's LLE.

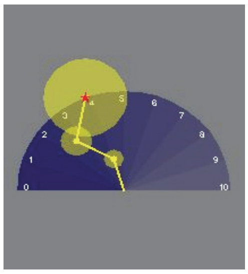


Figure 12: No.22's LLE.

4) 5 individuals saw their LLE levels rise after completing the meal. Only 1 person saw their level decrease immediately after eating, and only one person saw nearly no change to their level.

There was a marked result in that many saw very little fluctuation of their fingertip pulse wave.

3.1.3 Autonomic Nerve Balance Levels after Meal Intake

There were many individuals who saw rise in sympathetic nerve activity immediately following food intake. Among 17 participants, 13 experienced this. By chewing food and movement of the jaws, and by intake of warm food, it is thought that sympathetic nerve activity was stimulated. This is attributed to increased nervousness and bodily activity. Athletes experience the same kind of thing when they chew gum.

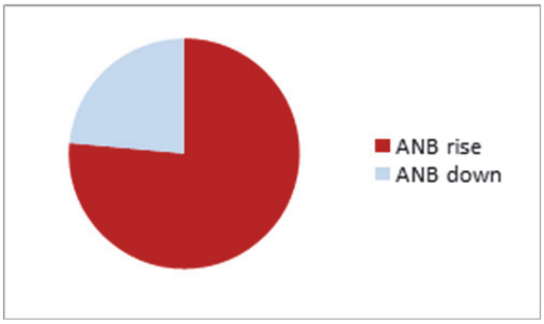


Figure 13: Autonomic nerve balance following food intake.

Among 17 participants, 11 people saw a mountain-shaped drop in levels 30 minutes after food intake. That is, immediately following food intake there was a temporary dominance of sympathetic nerve activity, and then with the start of digestion, within 30 minutes there was a shift to dominance of the parasympathetic nerve activity.

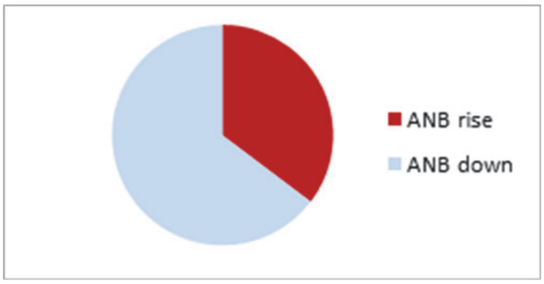


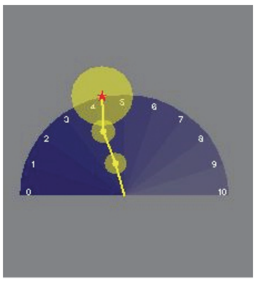
Figure 14: Autonomic nerve balance 30minutes after food intake.

3.2 Intake of Hot Soup

There were 13 participants who ate hot soup. Groups of 3-5 members were formed for the experiment.

3.2.1 LLE after Hot Soup Intake

1) 8 people showed little change (nearly flat reading on chart)



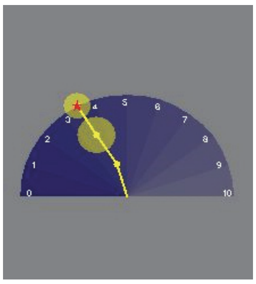
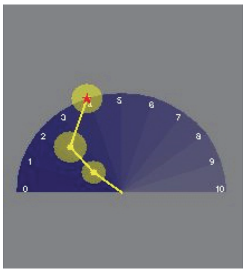


Figure 15: No.3's LLE. Figure 16: No.4's LLE.



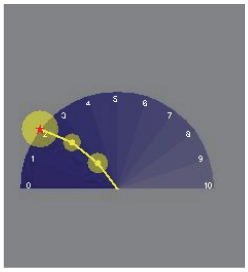
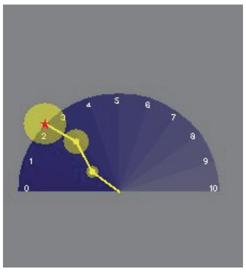


Figure 17: No's LLE. Figure 18: No's LLE.

2) 3 people showed a slight increase (no decrease after eating)



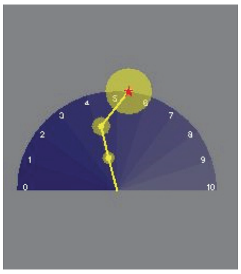


Figure 19: No.2's LLE. Figure 20: No.11's LLE.

3) 2 people showed a marked decrease, then an increase (of these, 1 person's final reading was lower than start)

3.2.2 Autonomic Nerve Balance Levels after Hot Soup Intake

1) 8 people showed sympathetic nerve increased activity immediately following meal (marked increase)

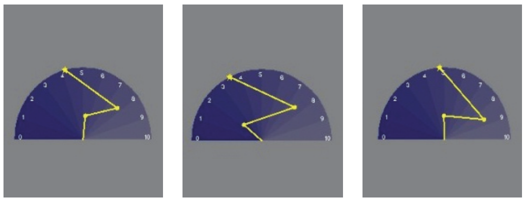


Figure 21: No.6, 11 and18's autonomic nerve balance.

2) 7 people showed a mountain-shaped increase, then decrease of autonomic nerve balance.
3) 4 people showed sympathetic nerve decreased activity immediately following meal
4) 1 person showed no change
Warm drinks (hot milk, etc.) will cause drowsiness with some people.

3.3 Intake of Cold Soup

13 individuals drank cold soup, and measured their fingertip pulse wave before and after the meal.

3.3.1 LLE after Cold Soup Intake

1) Out of 13 people, 8 people showed little change in readings before, immediately after, and 30 minutes after food intake.

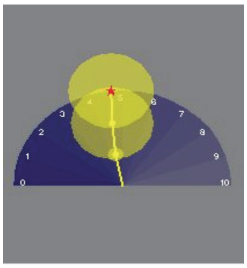


Figure 22: No.1's LLE.

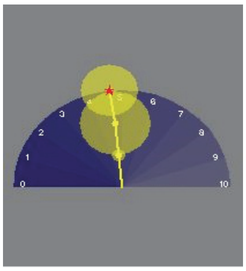


Figure 23: No.3's LLE.

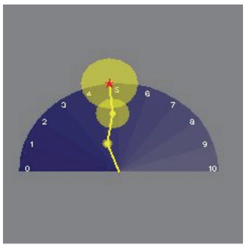


Figure 24: No.4's LLE.

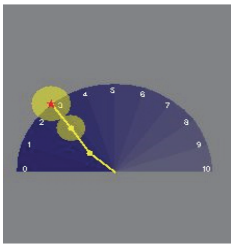


Figure 25: No.14's LLE.

2) 3 people showed an increase in LLE.
3) 1 person showed a temporary rise in LLE, followed by a decrease, and then a return to the same level as before the meal.
4) 1 person showed a temporary rise in LLE, followed by a decrease to a level lower than before the meal.

As with eating meals and drinking hot soup, there was not one case where a person saw a temporary drop in LLE followed by a large rise.

3.3.2 Autonomic Nerve Balance Levels after Cold Soup Intake

1) 8 persons showed a decrease in levels in the period from before the meal until 30 minutes afterward (when the 3rd reading was taken).
2) Among these, 6 people saw a steady decrease in levels from immediately after eating until 30 minutes later, and 2 people saw levels temporarily rise sharply, followed by a decrease to lower than original levels.
3) 2 people saw increased levels after the experiment.
4) 2 people saw little change in levels.

The body temperature may decrease.

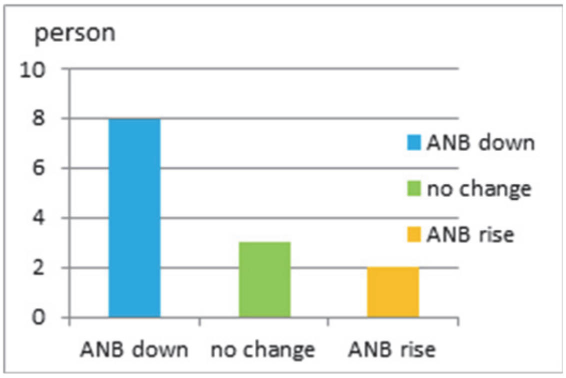


Figure 26: The change of autonomic nerve balance after cold soup intake.

3.4 Intake of Water

12 individuals took readings of their fingertip pulse wave before and after drinking water.

3.4.1 LLE after Water Intake

1) From 12 people who took readings before, immediately after, and 30 minutes after drinking water, 5 persons showed little change in LLE levels

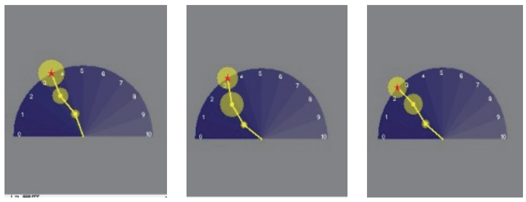


Figure 27: No.4, 14 and18's LLE.

2) 4 people showed an increase in LLE levels. Among these, 3 people saw an immediate rise, and 1 person saw a temporary drop in LLE, followed by an increase.
3) 3 people saw an immediate drop in LLE levels.

3.4.2 Autonomic Nerve Balance Levels after Water Intake

1) 5 people experienced an increase in comparison of readings before water intake (1st reading), and 30 minutes after water intake (3rd reading).
2) Among these, 3 people showed a temporary decrease in levels just after water intake, followed by an increase. 2 people saw an immediate rise in

levels. 1 person saw an initial rise, followed by a decrease.
3) 5 people saw a decrease in comparison of readings before water intake (1st reading), and 30 minutes after water intake (3rd reading).
4) Among these, 3 people saw a temporary slight rise in levels immediately following water intake, and then a drop to levels lower than before intake. 2 people saw a drop in levels, followed by an increase to a back to a level lower than at the beginning.
5) 1 person saw only a slight change in their levels.
6)- All told, the split between people who saw a rise after water intake, and those who saw a drop was 50/50.

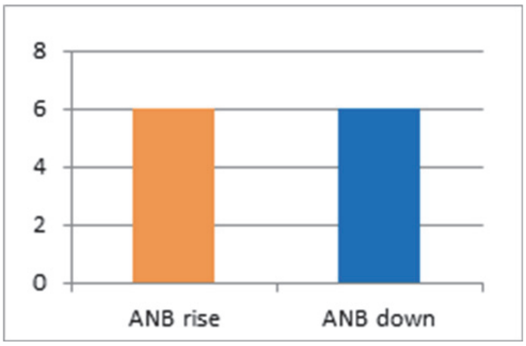


Figure 28: Change of ANB after water intake.

Because of cold soup and corn pottage, and because of long periods of time where the stomach was in a cold condition, it is considered that this is why there was a small proportion of persons whose levels rose 30 minutes after food intake. In the case of water, it is thought that this is due to absorption of the water within 15 minutes of intake.

4 CONCLUSIONS

With food intake, the frequent pattern was that immediately following a meal, there was a temporary and significant drop in LLE levels, followed by a rise in LLE. Most of the participants saw little change in levels immediately after intake of hot soup, cold soup, or water. By observing this only, it was learned that meals have a significant influence on the body. When well-chewed food enters the stomach, it appears that the brain's natural response is to trigger a "digestion" bodily activity, and due to a change in blood flow, it is thought that the LLE level, which indicates temporary brain activity, is lowered. It has been verified that LLE shows the adaptability of the outer portion of the mind, however, "immediately following food intake" centers on the act of eating, so it is understood to be a natural result that LLE adaptation in the outer portions become less. Therefore, when a major activity is being undertaken, before a test, ensuring that participants do not have a full stomach is considered the appropriate means, and is verified by bodily information. However, in the group where several people enjoyed a meal and afterward had enjoyable conversation together, LLE levels in everyone increased, and participants' sense of awareness was also much enjoyed. When considering that these participants' felt the food was delicious and were totally satisfied, it was realized that it is desirable to enjoy meals with close friends or family, and even though LLE levels immediately following eating were low, after this there was a great shift with LLE levels increasing, and an increase of mental activity.

Regarding results immediately following food intake, it was observed that many individuals saw the sympathetic nerve become dominant. However, as mentioned above, movement of the mouth and chewing activity is considered to cause a sharp increase in sympathetic nerve activity. In order to see parasympathetic nerve domination within 30 minutes of eating, activity like making the mouth move by chewing gum is thought to be effective. For this study, because the setting involved food intake, the phenomenon of decreased LLE levels immediately following food intake was seen, but there is a great interest to verify what effect simply chewing of food has on LLE levels. From the results of this survey, since it was observed that following food intake the parasympathetic nerve became dominant, it is obvious that, as expected, "rest from food" becomes necessary. Hypothetically, predicting that food intake has a healthy effect on the mind, it was possible to predict that after food intake LLE levels would rise. However, it was observed that initially LLE levels dropped before rising, and following the meal parasympathetic nerve activity and digestion activity commenced, and it was observed that resumption of work activities while the body experienced rest from food was appropriate. It is obvious that the activity of eating food is a significant source of work in humans, and an important activity in daily life

However, for participants of the group who took their meals alone, there were many cases where LLE levels either immediately lowered, or had very little fluctuation. This group included individuals who always were relaxed while eating and then became drowsy, and were aware of the effect of eating. In this way, it could be observed that there are varying levels of awareness as to the effects of eating on the body, and that for about 30 minutes after food intake, many people experience mental fatigue.

With intake of only small amounts of soup, LLE levels fluctuated very little. However, with intake of hot soup, it was observed that a significant number of participants who initially experienced little LLE fluctuation did see a climb in LLE levels 30 minutes after eating. Many people increased levels from heat stimulation of the sympathetic nerve, then afterward a decrease, and there is speculation that there is a kind of "relaxation effect" that they experienced. It seems that if a person wishes to be calm and relaxed, drinking something warm is effective. Similar to the fast absorption of plain warm water, for this experiment we used pottage soup, warm milk, kudzu starch gruel, etc., and it seems that hot beverages for comparatively long times in the stomach are beneficial.

Regarding intake of cold food, since it appears that there are many people whose LLE levels and autonomic nerve balance see little variation, it is thought that rather than cold food, hot food has a better effect on the body.

For participants who drank water and after 30 minutes saw a rise in sympathetic nerve dominance, it is believed that the result of seeing low levels after taking cold pottage soup is due to the fact that absorption of the liquids is rapid.

From these results, it seems evident that partaking of warm food while in the company of friends or family is a good practice. In Japanese society, the recent and rapid increase of young people and elderly who live alone and eat cold food appears to be a hindrance to overall health of the mind and body, and adversely affects quality of life. It seems obvious that eating of meals and good nutrition are very necessary for QOL (Quality of Life), and that this has been verified by the readings measured in the human body during this experiment.

In the future, it seems apparent that more research should be done to investigate the effects of what occurs when one eats food that is liked, what is the optimum environment and time for meals, as well as other variables.

REFERENCES

Fujita M., Kawashima k.,. (2012). The effect of Intervention for impruvement to eating habit of student nurse,for impruvement to eating habit of student nurse. Ishikawa nursing magazine, 9, 53-59

Hirohashi Y. and Oyama-Higa M. (2008). The use of a non- linear analysis of pulse waves to measure the impact of music therapy and animal therapy on psychiatric care. 2008 IEEE Conference on Systems Man, and Cybernetics proceedings, 3348-3352.

Hirohashi Y., Oyama-Higa M. and Lee S. (2011). The relations between the mental condition of the care house residents and finger plethysmograms. 2011 International Symposium on Computational Models for Life Siences proceedings, 128-135

Ishiguti A., Saito A.(2008,).The change of brain waves by the pungency of scent and subjective eveluation. Magazine of aromatherapy, 8 (1),15-19

Kunieda S.(2011).The effect by the scent to the ability of recognition. The enviroment of smell and scent,42(5), 354-360

Oyama-Higa M. and Miao T. (2006). Discovery and application of new index of cognitive psychology. 2006 IEEE Conference on Systems, Man, and Cybernetics proceedings, 2040-2044.

Oyama-Higa M., Miao T., Tsujino J., Imanishi A. (2007). Possibility of mental health self-checks using divergence of pulse waves. 2007 IEEE Conference on Systems, Man, and Cybenetics proceedings, 3952-3960.

Oyama-Higa M.,Miao T., Tanaka K. and Cheng H. (2007). Development of a self-check system for mental health using a pulse wave mouse. 2007 IEEE Conference on Systems, Man, and Cybernetics proceedings, 239-248.

Designing Spoken Dialogue Systems based on Doctor-Patient Conversation in the Diagnosis Process

Tesfa Tegegne and Theo van der Weide
Institute for Computing and Information Science, Science Faculty, Radboud University, Nijmegen, Netherlands
{*tesfa, tvdw*}*@cs.ru.nl*

Keywords: Doctor-Patient Interaction, Culture, Spoken Dialogue System.

Abstract: The paper describes the use of doctor-patient interaction in order to design a spoken dialogue system. In this paper we analyze doctor and patient information seeking and information provision behavior. In our experiments we found a difference in information seeking and information provisioning behavior of doctors and patients, although statistically not significant. We identify some important roles that can be used as a springboard to design a spoken dialogue system. Finally, we conclude that analyzing face-to-face doctor-patient interaction can serve as an effective starting point to design spoken medical dialogues.

1 INTRODUCTION

Speech is one of the most effective means of communication for humans. It plays a great role especially in man-machine interactions. Speech is natural and the vast majority of humans are already fluent in using it for interpersonal communication. In the last two decades there have been a lot of advances in the application of spoken dialogue systems in different areas: academia, military, and telecom companies. A dialogue system is one of the promising applications of speech recognition and natural language processing. Spoken language interaction with computers has become a practical possibility both in scientific as well as in commercial terms.

Spoken dialogue systems can be viewed as an advanced application of spoken language technology. Spoken dialogue systems provide an interface between the user and a computer based application that permits spoken interaction with the application relatively in a natural manner (McTear, 2002). Fraser 1997, cited by (McTear, 2002), defines spoken dialogue system as "computer systems with which humans interact on turn-by-turn basis and in which spoken natural language plays an important role in the communication". Spoken dialogue systems enable semi-literate and illiterate users to interact with a complex application in a natural way using speech. Current spoken dialogue or/and IVR (Interactive Voice Response) systems restrict users in what way they can say and how they can say it. However, users of speech-based dialogue systems often do not know exactly what information they require or how to obtain it- they require the support of the dialogue system to determine their precise requirement (a system directed dialogue). For this reason, it is essential that the spoken dialogue system is be able to engage users in the dialogue rather than simply respond to predetermined spoken commands.

The propelling factor of this study is the adoption and application of speech technology and its impact on the healthcare sector in developing countries where a significant number of the population is illiterate or semi-literate. In this paper we analyze the doctor-patient conversation in order to find out if we can emulate the methods and techniques used in the conversation to design medical spoken dialogue system that can be accessed easily by non-educated or semi-educated population with the objective of searching health information remotely.

Nowadays, mobile becomes an appropriate medium to minimize the burden of healthcare in developing countries. Several researchers, for example (Black et al., 2009; Bickmore and Giorgino, 2006; Foster, 2011), are engaged in mobile based applications such as mHealth, mLearning, mBanking, mAgriculture etc. The expansion of the mobile network and the increment of mobile phone users in Ethiopia provide a fertile ground to adopt and implement mobile based healthcare applications. Designing a medical dialogue alike doctor-patient interaction is very cumbersome, the advancement of speech recognition requires another level to fully understand human speech. It requires at least to design and develop a medical dialogue system resembling human-like conversation to reduce the error of speech recognition. Our main motivation is to understand whether face-to-face doctor-patient interaction plays a vital role in designing human-like medical spoken dialogue system for the healthcare domain in the case of Ethiopia using Amharic language. To the best of our knowledge this is the first study that not only analyzes the doctor-patient conversation but also designs and models a medical dialogue system in the case of Ethiopia.

In this paper we address the following question: Is it possible to design a spoken medical dialogue system based on doctor-patient face-to-face conversation in the diagnosis process? The paper is organized as follows. The state of the art of medical dialogue systems is discussed in section 2. The analysis of the experiment is presented in section 3. Section 4 explains the proposed spoken dialogue system. Finally, section 5 concludes the paper and gives future directions.

2 SPOKEN DIALOGUE SYSTEMS IN HEALTHCARE

The healthcare domain has gleaned the benefits of the advancements of Information Communication Technology. During the last two decades, these interfaces have been adopted as part of tele-medicine technologies (Bickmore et al., 2006; Bickmore and Giorgino, 2006), which enable the delivery of a variety of medical services to sites that are at a long distance from providers.

The ultimate goal of a dialogue system is to provide health information for stakeholders primarily using spoken dialogue. Such a system can be used for a wide range of applications including: patients self-treatment and management, disease remote monitoring, diagnosis, health education etc. In line to this Bickmore and Giorgino ((Bickmore et al., 2006)) say that automated dialogue systems are increasingly being used in healthcare to provide information, advice, counseling, disease monitoring, clinical problem identification, as well as enhancing patient-provider communication.

However, a dialogue system in the healthcare domain is not without challenges. Clinical practice ordains complicated guidelines, ontologies and procedures. This makes the dialogue system more complex and cumbersome to handle. Bickmore and Giorgino also mention some of the challenges of spoken medical dialogue system: criticality (emergency cases), confidentiality (privacy such as HIV/AIDS regimen etc.) and mixed initiatives (patient-centered vs. system-centered). They point out that incorporating medical and behavioral ontologies and deep knowledge of health communication strategies are very important for further development of medical dialogue systems.

Bickmore and Giorgino argue that face-to-face communication together with written instructions remains one of the best methods for communicating health information to patients with low literacy level. They report that face-to-face consultation is effective because providers can use verbal and nonverbal behavior, such as head nods, hand gesture, eye gaze cues and facial displays to communicate factual information to patients, as well as to communicate empathy and immediacy to elicit patient trust. According to Durling and Lumsden ((Durling and Lumsden, 2008)) a spoken dialogue takes half the time needed to achieve the same task using keyboard and mouse, regardless of the participant's ability to correct their input. By highlighting the business side of speech recognition in healthcare, Parente et al., cited by Durling and Lumsden, show that, in their opinion, the adoption of speech technology is worthwhile. The use of speech recognition has, in fact, seen the most successful adoption in the healthcare domain.

2.1 Diagnosis Systems

Diagnosis, according to Webster's dictionary,*is the act or process of deciding the nature of disease or a problem by examining symptoms*. In the medical domain, many diagnosis systems are proposed and used such as decision support systems, agent based systems, and intelligent systems (fuzzy logic, expert system, neural networks and the like). Mobile diagnostic technology is a relatively new concept in telemedicine. As suggested by the term itself, it involves two key characteristics: mobility and remote diagnosis (Celi et al., 2009). The aim of using mobile technologies for healthcare is to support the patients outside of the medical and/or home environment.

2.2 Doctor-Patient Face-to-Face Interaction

The main goal of a doctor-patient conversation is a focused gathering with a common goal pursued by its participants. Typically, a patient visits a doctor with the purpose to be relieved from feeling unwell possibly caused by an illness; the doctor's purpose of the interaction with the patient also is to relieve the patient. When both parties appear to fail to comprehend or understand each other's goal, the interaction may be dysfunctional. Studying and analyzing the doctor-patient interaction helps to convey empathy and obtain a trusted spoken dialogue system, especially when used by the semi-literate and illiterate rural people. The analysis should help to design a full-fledged medical dialogue system. As we pointed out in the previous sections, the main idea here is not to analyze the effectiveness and efficiency of the face-to-face doctor-patient interaction but rather the implications towards designing and modeling 'human-like' medical dialogue system.

A good interaction and a quality relationship between doctors and their patients is now widely recognized as a key factor in improving not only patient satisfaction but also treatment outcomes across a wide range of healthcare disciplines. The use of specific doctor communication skills has been associated with improved adherence regimens, improved psychological outcomes, more detailed medical histories and fewer malpractice suits, in addition to increase patient satisfaction (Bickmore et al., 2005). Doctor's listening behavior is a necessary ingredient for an interaction in which patients are describing and expressing themselves freely and openly (Nevile, 2006).

To test the doctor's listening behavior, the relative frequency of each category of information seeking and information giving behavior is calculated for the doctor-patient interactions, representing the number of dialogue acts in any particular category as a proportion of the total number of dialogue acts made by the speaker. A relative frequency of medical topics addressed in every interaction is calculated for all interactions. The doctor as a facilitator of doctor-patient interaction should demonstrate a high frequency of supportive and encouraging behavior in the presence or absence of patient desired behavior. Even though a patient's characteristics are very important to the quality of interaction, the doctor's facilitating behavior is essential since it allows patients either to express themselves or to repress (Nevile, 2006).

3 FINDING CULTURAL DEPENDENCIES

Information seeking behavior includes seeking information about medical topics. Direct, assertive and embedded question types are posed by both parties. On the other hand, information provisioning or information giving is providing a direct answer to a question, elaborating the question by providing supplementary information and deviating or changing the topics of the interaction without prompting from the interacting partner.

In this study we group the doctor-patient conversation into two categories: information seeking behavior and information provisioning behavior. Information seeking consists of utterances about information gathering, checking and cueing. Information provisioning contains utterances dealing with explanation, confirmation, and giving instructions. Both information seeking and information provisioning are analyzed against medical topics: illness, symptoms, diagnosis, treatment, exam/test and history (medical history) for both doctor and patient.

In the analysis we remove backchannels. Backchannels contain, greetings, and acknowledgment that carry little information value for healthcare doctors and patients. Removing backchannels should not affect the quality of information obtained from the interaction. We use a two-way ANOVA (Analysis of Variance) to see the difference between doctor and patient in the two criteria and medical topics or themes. Additionally, we check the number of questions posed by doctors and patients. Finally, we analyze the overall interaction process.

3.1 Methodology

The focus of this research is to design a spoken medical dialogue system on the basis of doctor-patient face-to-face interactions, to understand the weak and strong side of the interaction and to utilize actions taken place during the interaction. We have conducted a number of observational studies in which we recorded the interaction between patients and health professionals. Based on these observations we design the content of the conversation including question and declarative statements, the order of presentation of content, how a system responds to questions and words, sentence structure and tone used, to closely match the user expectations of what a health professional might ask, respond and sound like. In-depth interview studies show that this is perceived by patients as a successful conversation (Migneault et al., 2006).

The doctor-patient interactions are analyzed as follows. We classify the elicitation process based on semantic entities as shown in figure 1, leading to the following topics: (i) Illness, (ii) Symptoms, (iii) Diagnosis, (iv) Previous Treatment, (v) Current Treatment and (vi) Exam. Questions and Explanations are targeted towards the following variables proposed by (Nowak, 2011): (i) information, (ii) confirmation, (iii) checking, (iv) explanation, (v) cue and (vi) giving instruction.

First we address the efforts taken by both doctor and patient to identify problems and attempting to recommend. This includes the process of elicitation (information gathering), explanation, confirma-

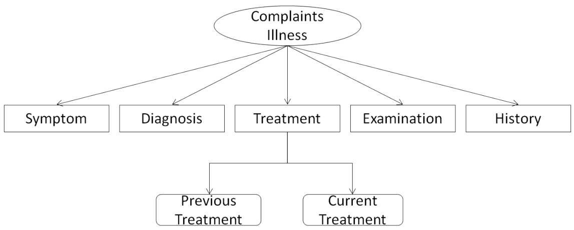


Figure 1: Semantic Entities of Doctor-patient conversation.

tion, checking, cue and giving instruction. We transcribed the audio recordings manually and tabulating into three categories: Information elicitation or information seeking and information provisioning or (information giving) in the case of the doctor and the patient in the interaction. We analyzed 29 conversations done by 4 doctors and 29 patients among 50 patients. We chose only patients coming for diagnostic purposes and ignored follow-up patients since it doesn't include the elicitation criteria (information gathering/seeking and information provisioning) and medical topics (illness, symptoms, diagnosis, treatment, exam/test and history) for diagnostic purpose.

3.2 Results

We have collected data to find out how, in the Ethiopian context, patients elicit information during a doctor-patient conversation. Therefore we have audio-taped 50 real doctor-patient interactions. The conversations took 6-7 minutes on average. The statistical analysis of the face-to-face doctor-patient interaction has resulted in the following findings. Out of 29 medical interactions, comprising of 442 turns, 171 (38%) turns classified as Information gathering, 167 (38%) utterances are uttered by doctors to elicit health information and only 4 (1%) of the utterances are uttered by the patients. Table 1 shows the breakdown of the doctor contributions to the doctor-patient interactions into (variable, topic) combinations. Table 2 shows this breakdown for the patient contributions. We may interpret this table as an estimator for probabilities as Prob (v—t, a) for variable v, topic t and actor a (either doctor or patient).

Table 1: Doctor utterances in medical information elicitation process in face-to-face interaction.

	Illn	Smpt	Diag	P.Tr	C.Tr	Exam	Hist	Total
Information	34	110	9	5	0	1	8	167
Confirmation	0	2	0	0	0	0	0	2
Checking	1	5	9	8	1	4	1	29
Explanation	6	4	9	3	1	5	0	28
Cue	2	1	2	0	1	0	1	7
Instruction	0	0	3	1	0	5	2	11
Total	43	122	32	17	3	15	12	244

Our first conclusion is that the interaction is mainly led by the doctor. The first dialogue act is a

Table 2: Patient utterances in medical information request in the face-to-face diagnosis processes.

	Illn	Smpt	Diag	P.Tr	C.Tr	Exam	Hist	Total
Information	1	2	0	0	0	1	0	4
Confirmation	6	36	1	4	2	2	1	52
Checking	0	28	2	2	0	0	2	34
Explanation	12	76	5	6	0	1	3	103
Cue	0	0	4	1	0	0	0	5
Instruction	0	0	0	0	0	0	0	10
Total	19	142	12	13	2	4	6	198

question posed by the doctor like 'How are you feeling?'. Then the patient explains complaints or feelings (s)he has. The doctor then will ask additional questions to identify causes, symptoms, and illnesses.

Table 3 and 4 show information seeking and information provision of doctors and patients. We see that information seeking behavior of doctors (46%) is higher than that of patients (10%). Whereas, patients' information provisioning behavior (35%) is much higher than doctors' (9%). Besides, symptoms 60%, illness 14%, diagnosis 10%, treatment 8%, exam test and history 4% are addressed. In these doctor-patient interactions the medical topic has been given much attention during information seeking and information provision (about 122 turns used by the doctor for information seeking and 142 turns by patients to reply the questions posed by the doctor about the symptoms). The most prominent divergence appears in the fact that doctors most frequently initiated questions targeted towards information (68%), much less frequently towards checking (12%), and even less frequently towards explanation (11%), giving instruction (5%), cue (3%) and conformation (1%). The patients merely were answering the questions posed by the doctor.

Table 3: Information seeking and information provision behavior of Patient.

	Illn	Smpt	Diag	P.Tr	C.Tr	Exam	Hist	Total
InfoSeek	1	30	2	2	0	1	2	38
InfoProv	18	112	10	11	2	3	4	160

Table 4: Information seeking and information provision behavior of doctor.

	Illn	Smpt	Diag	P.Tr	C.Tr	Exam	Hist	Total
InfoSeek	35	115	21	14	1	10	11	207
InfoProv	8	7	11	3	2	5	1	37

3.3 Discussion

As discussed in the preceding section, turn taking is a dialogue act. A turn is defined as speaking without interruption. From the doctor-patient audio recordings we found that the number of turns of doctors is 276 and 238 of the patients. This figure indicates that the doctor speaks in long turns 56% over patients 46%. In general, patients talk less than doctors and most of their interaction is in the form of giving information in response to doctor questions. Many studies indicate that doctor's dialogue acts encourage patients to discuss their opinions, express feelings, ask questions, and participate in decision making. This helps the doctor to more accurately understand the patient's goals, interests, and concerns as well allows the doctors to better align his conversation/interaction with the patient's agenda (Goold and Lipkin, 1999; Verlinde et al., 2012). On the contrary, some studies report that doctors often underestimate patients' desire for information, while overestimating their medical knowledge (Bickmore et al., 2006). Thus, allowing patients to ask questions, express concerns and state preference helps the doctor to infer matters that are important to patients in relation to their compliance.

Our findings show that the dialogue is initiated by the doctor in order to seek information about the patients health compliance and illnesses. Moreover, the dialogue is controlled by doctors to gather additional information about the illness (such as symptoms, illness history and medication). Generally, the request for information sets the initial purpose or goal that motivates the speaker's actions for the remaining section of the dialogue. The request for information further specified by one or more discourse segments. Asking questions and providing answers play a significant role in the process of the medical consultation. Mainly, the aim of doctor-centered behavior is efficiently gathering sufficient information to make a diagnosis and consider treatment options in the least amount of time necessary. This is in contrast to patient-centered interactions that can recognize patients as collaborators who can share not only their biomedical states (physical condition and well being) but also knowledge of their psychological situations (personality, culture, social relations, etc.). The result of quantitative analysis shows, there still is an equal distribution of information seeking (questions) between the dialogue participants, with almost all elicitation initiated on the part of the doctor. The data also demonstrate that both doctors and patients emphasize on asking and responding about symptoms and illnesses. For example from a total of 442 dialogue acts, 122 and 142 dialogue acts are used to elicit symptom information by doctors and patients. Generally, 59.7% of the interaction was devoted to seeking and giving information about symptoms and 14% of illnesses. The finding show a significant difference ($F= 14.02$, $P=0.0026$, $alpha= 0.05$) between doctors' and patients' information seeking behavior in f-test, but no significance difference is found in patients' and doctors' information giving behavior.

We also analyzed the data using two-way analysis of variance. We found that the information seeking score of doctors is higher than for patients, but the difference is statistically not significant. The score of information giving behavior of patients was higher than the doctors, but similarly there is no significant difference. The first impression about people often turns into long-term perceptions and reputations. So doctors in their first encounter should make good eye contact, shake patient's hand and introduce himself. In the face-to-face interaction we recorded that there were no greetings and introduction during the initial doctor-patient interaction. In each dialogue sentence or clauses the participant (doctor/patient) utterance is categorized into semantic entities (figure 1) in which the dialogue theme is emphasized. Since the conversation is between a doctor and patient for diagnosis purpose we identify the main concepts evolved in the interaction process. Compliant, symptoms, treatment, illness, exams, history, and prescription are the most common entities used in the doctor-patient conversation.

3.4 Cultural Aspects

Cultural differences may be an obstruction for effective doctor-patient interaction. The cultural perceptions of health, sickness, and medical care of patients and families may differ with that of the doctors. Speaking the same language and being born in the same location does not automatically mean sharing all the elements of a particular culture. Studies have shown that a patient's culture will affect the way they perceive their body, illness, and disease. This is also true for the doctors as their own families and communities have also helped to shape these cultural beliefs within them. Each participant in the medical interview brings with them the culture in which they were raised. At times, differing cultural beliefs can have an adverse effect on the care that one receives. Communication problems arise when the patient and doctor do not share the same culture.

Culture competencies in medical interaction provide a patient centered care by adjusting their attitudes and behaviors to the needs and desires of different patients and account for emotional, cultural, social, and psychosocial issues on disease and illness. Medical competencies relate directly with the doctor-patient interaction that are required by the doctors to conduct an effective interview and to create an acceptable plan of diagnosis and treatment. Studies indicate that issues that may cause problems in cross-cultural encounters are authority, physical contact, communication styles, gender, sexuality, and family.

Hofstede (Hofstede et al., 2010) has identified five

cultural dimensions. (1) Power Distance focuses on the perceived degree of equality, or inequality. According to Hofstede et al., [2010] ”A high power distance ranking indicates that inequalities of power and wealth have been allowed to grow with the society. In these societies equality and opportunity for everyone is stressed”. In large power distance cultures, ones social status must be clear so that others can show proper respect. In line with this, Hofstede et al. assert that the power distance exhibited in society also is reflected in the relationship of doctors and patients. They say that ”in countries with large-power distance cultures, consultations take less time, and there is less room for unexpected information exchanges”. The findings indicate that the average time spent on face-to-face consultation is 4-6 minutes. This result confirms that power distance plays a major role in doctor-patient interaction. According to Hofstede, Ethiopia is a large power distance country, so the interaction is dominated by doctors and patients rarely participated in treatment and diagnosis decision makings.This is true especially for illiterate and rural people. The power distance of literate people and doctors is better compared to the illiterate. In line with this (Verlinde et al., 2012) said that doctors asked less educated patients and low income patients more questions about their disease and medical history. Likewise, our findings indicate that doctors' information seeking behavior is more than that of patients'. Generally, in Ethiopia, patients treat doctors as superiors, consultations are shorter and controlled by doctors.
(2) Hofstede's cultural dimension indicates that Ethiopia as a low individualism country. The implication of individualism in healthcare particularly in doctor-patient interaction goes with patient autonomy, the possibility of choice, flexibility of social roles, less conformity, and psychosocial information exchange (Meeuwesen et al., 2009)
(3) Ethiopia is a masculine country (Hofstede's cultural dimension); however, regardless of other dimensions, masculinity doesn't reflect on the patient - doctor interaction in diagnosis and treatment. Some studies revealed that there is a difference between female and male doctors in creating partnerships, with patients and dealing with psychosocial issues during the conversation. Meeuwesen et al., [2009] stated that the more masculine a county, the more instrumental (disease centered) interaction will dominate, the less attention will be paid for psychosocial issues and more frequently the majority of doctors will be men or male. The analysis result shows that mainly the interaction between doctors and patients was on the theme of symptoms 60% and illnesses 14%. Eventually the theme of the conversation is disease-centered.
(4) Uncertainty Avoidance in the healthcare domain primarily deals with patients' emotionality or anxiety, or stress and doctor's task-orientation, preferences of technological solution and degree of medication. In countries with strong uncertainty avoidance (Meeuwesen et al., 2009) the more disease-centered (instrumental talking), the less affective talking and the more biomedical exchange can be expected. This a true scenario in Ethiopia cases; since doctors indulge themselves in diagnosing the illness. In the experiment, we have not found a single introduction (greetings) communication act. Hofstede further explained that ”doctors in uncertainty tolerant countries more often send patients away with comforting talk, without any prescription. In uncertainty avoiding countries doctors usually prescribe several drugs, and patients expect them to do so” (Hofstede et al., 2010).
(5) Regarding long-term orientation, as Ethiopia doesn't have data we left out in our analysis.

4 A SPOKEN DIALOGUE MODEL

Eliciting user requests in the medical spoken dialogue is the main challenge for developers and implementers of the system. Unlike a face-to-face doctor-patient interaction it is very hard to analyze the patients' attitudes and emotions. As a result the eliciting techniques should be patient centered; and the main role of the doctor is a facilitating behavior, focused and unfocused open questioning, request for clarification, summarizing and empathy. Thus, the dialogue system should act like human which can help to elicit the patients' request in order to provide accurate consultation, diagnosis and treatment.

To the best of our knowledge eliciting user medical requests using spoken medical dialogue based on some suggested principles is not assessed, and there is no any results obtained. Our objective is using the best practice of in-person doctor-patient interaction activities to be adopted in the spoken medical dialogue system to search medical information using mobile phones.

4.1 A Simplified Dialogue System

Spoken medical dialogue tends to be patient centered. Thus the system should facilitate the interaction and ask open questions in which the patients can express not only knowledge of their biomedical state (illness and complaints) but also knowledge of their psychological and social situations (personality, culture, relationships). As discussed before, the face-to-face interaction in Ethiopia is doctor dominated. However, in the dialogue system it is impossible to detect the non-verbal behavior of the patient. Thus, the elicitation should be dominated by the patient in order to seek biomedical as well as psychosocial situations of the patient. Doctor's behavior that encourages patient active participation includes asking open ended questions, ensuring and confirming patient comprehension, requesting patients' opinions, and making statements of concern, agreement and approval. Hence, spoken dialogue to resemble human-human interaction, should encourage patients to take part actively in the interaction process. Instead of being expecting responses from patients, the system must take a facilitative role in order to provide time and space for patients to speak out what their symptoms, illnesses, suggestions and to participate in decision making.

From the analysis of the in-person interaction of doctors and patients, it was found that there are some gaps that should be filled. The main gaps observed in the face-to-face interaction is doctor domination as well as we identified that the social status of patients and doctors inhibits the interaction process. Other factors that affect the face to face interaction are illiteracy and culture. In rural Ethiopia, the illiteracy rate is higher than in urban areas, so patients from rural areas visited find the interaction with the doctor is difficult; the doctor may consider that non educated rural patients do not express themselves so that a doctor prefers to ask some closed questions and open leading questions to elicit the user requirements. But even when doctors and patients born and live in the same area, they do not necessarily have the same understanding of social norms and cultures. Consequently, the non-literate rural patients are more conservative of their values and cultures; some of the illness may not be disclosed in public so to keep their culture or values they reserved from disclosing their feelings, symptoms and illnesses. For instance, a study conducted in USA revealed the gap between doctors and Ethiopian migrants in disclosing illnesses and diseases. According to this study, the migrants did not want to be told if their disease is life threatening; whereas, doctors in US disclose the nature of the illness, the risks of the illness (curable and incurable) and the magnitude of the illness (treatable or non-treatable) (Beyene, 1992).

From the face-to-face doctor-patient interaction we deduce the user request elicitation model displayed in figure 2 for spoken dialogue system for healthcare application. The model consists of four components: opening initiatives, asking information, giving information and closing. Figure 2 displays user request elicitation process for the spoken dialogue systems in a healthcare scenario based on doctor-patient face-to-face dialogue.

4.2 Design Dialogue System

Data validity, accuracy and integrity are the vital points to be considered in designing a spoken dialogue application; since automatic speech recognition(ASR) technology is not perfect. The design of spoken dialogue technology should take into account the possibility of speech recognition errors and improve the overall accuracy using dialogue actions such as re-prompts, conformations, error correction and handling etc. Secondly, it should provide equal access to novice and experienced end users of the system. Thirdly, it should also consider individual differences such as personalization and user context. Finally, before developing the dialogue system it is very important to conduct a face-to-face interview or pay live observation while a doctor is treating a patient (if possible video tape the conversations). The most commonly applied methods to design a spoken dialogue include human-human dialogues and design by simulation. Thus, our interest lies on to look into doctor-patient interactions as a means to design medical spoken dialogue.

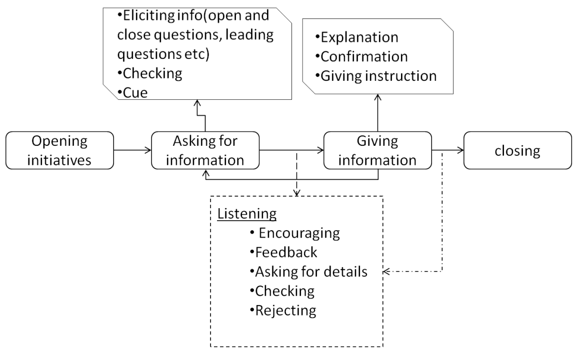


Figure 2: User request elicitation process.

4.3 Designing a Dialogue based on Doctor-Patient Interaction

Human-human dialogue provides an insight how humans accomplish a task-oriented dialogue. The doctor-patent interaction studies take place in the early stages of the speech application life cycle. They act as a starting point for spoken dialogue design and help to define requirements. The purposes of doctor-patient interaction studies are to help the designer see the task form the user point of view, develop a feeling for the style of interaction, and acquire some specific knowledge about the vocabulary and grammar used in the diagnostic process.

Doctor-patient interaction (natural dialogue) study differs significantly from the wizard-of-oz studies, that have been used extensively by others in the design of spoken dialogue systems. Researchers who use the wizard-of-oz techniques begin the process with a pre-experimental phase that involves studying natural human dialogues. Whereas the natural dialogue takes place prior to any system design or functional specifications (Yankelovich, 2008). The main purpose is to launch the design process.

Before designing the medical dialogue system, we wanted to discover how doctors and patients interact in the diagnosis process. From the analysis of the doctor-patient interaction, we found that the interaction is doctor-centered as well as we found that patients question asking behavior is hampered by the cultural influences such as: distance power, high uncertainty avoidance and the like (see section 3.5).

It is impossible to produce a medical dialogue system design based entirely on doctor-patient face-to-face interaction. Rather it can play an important role in the early stages of the development life cycle, and serve as an effective starting point for spoken medical dialogue system design.

5 CONCLUSIONS

We have analyzed the interaction in 29 audio-taped doctor-patient diagnosis dialogues in the Gamby Teaching hospital. The study is mainly conducted to investigate the information seeking and information provisioning behavior of doctors and patients. The finding shows that there is no statistical significant difference between doctor information seeking and patient information seeking behavior. Similarly, we didn't find any significant difference between patients information provisioning and doctors information provisioning behavior. From this analysis we conclude that studying face-to-face interaction between doctor and patients is an effective starting point for spoken medical dialogue system design. We also found an influence of culture on doctor-patient interaction; so cultural values should be incorporated while designing and developing a medical dialogue system. Finally, based on our results, we propose a model to assist user requirements elicitation in order to develop a medical spoken dialogue system. In the future we will implement our model to develop a medical dialogue system.

REFERENCES

Beyene, Y. (1992). Medical disclosure and refugees - telling bad news to ethiopian patients,. *The Western Journal of Medicine*, 157(3):328 332.

Bickmore, T. and Giorgino, T. (2006). Methodological review: Health dialog systems for patients and consumers. *Biomedical Informatics*, 39(5):556–571.

Bickmore, T., Giorgino, T., Green, N., and Picard, R. (2006). Special issue on dialog systems for health communication. *Biomedical Informatics*, 39(5).

Bickmore, T., Gruber, A., and Picard, R. (2005). Establishing the computer-patient working alliance in automated health behavior change interventions. *Patient Educ Couns.*, 59(1):21–30.

Black, J., Koch, F., Sonenberg, L.and Scheepers, R., Kahandoker, A., Charry, E., Walker, B., and Soe, L. (2009). Mobile solutions for front-line health workers in developing countries. In *Healthcom*, pages 89–93.

Celi, L., Sarmenta, L., Rotberg, J., Marcelo, A., and Clifford, G. (2009). Mobile care (moca) for remote diagnosis and screening. *Journal for Health Informatics in Developing Countries*, 3(1):17–21.

Durling, S. and Lumsden, J. (2008). Speech recognition use in healthcare applications. In *MoMM 2008*.

Foster, C. (2011). Icts and informal learning in developing countries.

Goold, S. and Lipkin, M. (1999). The doctor patient relationship challenges, opportunities, and strategies. *J Gen Intern Med.*, 14(Suppl1):26–33.

Hofstede, G., Hofstede, G., and Minkov, M. (2010). *Cultures and Organizations: Software of the Mind: Intercultureal Cooperation and Its Importance for Survival*. McGraw Hill.

McTear, M. (2002). Spoken dialogue technology: Enabling the conversational user interface. *ACM Computing Survey*, 34(1).

Meeuwesen, L., van den Brink-Muinen, A., and G., H. (2009). Can dimensions of national culture predict cross-national differences in medical communication? *Patient Education and Counsiling*, 75:58–66.

Migneault, J., Farzanfar, R., Wright, J., and Friedman, R. (2006). How to write health dialog for a talking computer. *Biomedical Informatics*, 39:468–481.

Nevile, M. (2006). Communication in context: A conversation analysis tool for examining recorded voice data in investigations of aviation occurrences. Technical report.

Nowak, P. (2011). Synthesis of qualitative linguistic researcha pilot review integrating and generalizing findings on doctorpatient interaction. *Patient Education and Counseling*, 82(3):429–441.

Verlinde, E., De Laender, N., De Maesschalck, S., Deveugele, M., and Willems, S. (2012). The social gradient in doctor-patient communication. *International Journal for Equity in Health*, 11(12).

Yankelovich, N. (2008). Using natural dialogs as the basis for speech interface design. *Human Factors and Voice Interactive Systems Signals and Communication Technology*, pages 255–290.

SPECIAL SESSION ON FUTURE INTERNET APPLICATIONS FOR TRAFFIC SURVEILLANCE AND MANAGEMENT (FIATS-M)

A Future of Traffic Management
Toward a Hybrid System of Roadside and Personal Mobile Devices

Tom Thomas
Center of Transport Studies, University of Twente, p.o.box 217, 7500 AE Enschede, The Netherlands
t.thomas@utwente.nl

Keywords: Travel Time, Travel Demand, Urban, Prediction, Detection Loops, GPS.

Abstract: In the last few decades, increasing traffic has led to serious problems in urban areas. Travel times for travellers have increased and the liveability in residential areas has declined due to pollution and issues concerning safety. Whereas new road and rail infrastructure in densely populated areas is costly, a better utilisation of existing infrastructure appears to be more attractive. This becomes increasingly possible due to huge developments in the world of Information and Communication Technology (ICT). However, implementing ICT solutions for the purpose of traffic management remains challenging. First, it is quite a task to set up a system in which infrastructure, travel vehicles and travellers can communicate with each other. Secondly, it is quite difficult to gather and exchange traffic and travel information in such a way that the traffic situation improves significantly. This paper deals with the latter issue, and it provides an outline for the possible architecture of a future traffic management system. It concludes that in such a system both mobile devices, like smartphones and navigation systems, and roadside devices, like loop detectors and cameras, need to be included to arrive at optimal results.

1 INTRODUCTION

Congestion has increased significantly in the last few decades. The efficient use of existing infrastructure by dynamic traffic management (DTM) is one of the strategies to reduce congestion and related problems like air pollution. An important requirement is the availability of detailed traffic information such as travel demand and travel times.

In the Netherlands and many other developed countries, highway data are collected by a high concentration of detection loops which yield information on traffic intensities and travel times. In urban areas traffic information is much scarcer and only since quite recently, traffic data gathered by roadside devices like detection loops have become available in traffic information centres (e.g., Hasberg and Serwill, 2000, Kellerman and Schmid 2000, Leitsch, 2002). For urban areas, the traffic circulation is usually estimated by a combination of intensity measurements and traffic models. However, with new measurement methods by which individual vehicles are identified (Blokpoel and Vreeswijk, 2011), accurate roadside measurements of travel times and routes will become possible.

At the same time, the use of mobile sensor data such as GPS and GSM has been increasing rapidly, which have led to separate travel time estimators (e.g., Google Traffic, 2013). Because these data are gathered by personal devices such as navigation systems and smartphones, travel information and advice can be personalized and adapted to the preferences of the individual traveller (e.g., Bie et al., 2012). In case of smartphones, travel information does not have to be limited to car trips, but may also include several other travel modes.

Roadside and mobile sensor data techniques both have their specific strengths. In this paper, a possible architecture for future DTM is put forward in which both data sources are combined at some point. In section 2, the requirements for a future DTM system are provided. Then, in section 3 and 4, the advantages, disadvantages and possibilities of mobile and roadside sensor data are described respectively. Based on the requirements for a future DTM system and the strengths and weaknesses of roadside and mobile sensor data, section 5 describes an outline for the architecture of a possible future DTM system.

2 DTM REQUIREMENTS

The requirements for a modern DTM system should be based on two pillars. First, the traffic system, usually the responsibility of a traffic manager, should run smoothly. This may mean several things, but in general the objective is to minimize the total delay on the (road) network and to minimize external costs caused for example by traffic accidents or air pollution. Secondly, the traffic users, i.e. travellers, want to travel smoothly. This may also mean various things for individual travellers, but in general travellers want to minimize their (perceived) travel time and cost while perceiving the journey as being safe and comfortable.

In other words, the traffic manager's objective is to optimize the traffic system as a whole, while travellers want to optimize their individual journeys. Both things are not necessarily resulting in the same traffic equilibrium, and in some cases they are even clearly conflicting with each other. One of the main challenges of a modern DTM is therefore to reconcile the two.

Meeting the objectives of the traffic manager and individual traveller also requires different information needs. An individual traveller needs to have traffic information about the possible travel modes and routes that are relevant to him, i.e. those that can be used to reach the preferred destination. Detailed information only needs to be available during the time of traveling. This is by no means trivial, because it still requires a prediction of the traffic situation in the near future, i.e. until the journey is expected to be completed.

However, this task is relatively easy compared to optimizing management objectives. Whereas the individual traveller is merely influenced by surrounding traffic, the traffic manager is influencing traffic itself, which effects are much harder to predict. Moreover, the effects of traffic operations like traffic lights are not necessarily instantaneous, but may show delays. For example, a traffic measure in one part of a city can have an effect in another part half an hour later. In the ideal case, the traffic situation should therefore be predictable for the whole network, for different traffic management scenarios, and during a longer period of time, for example a whole peak hour.

Fulfilling the needs of individual travellers on the other hand has its own challenges. While there is one traffic manager with one set of objectives, there are many individual travellers, all with their own perceptions, preferences and habits that play an important role in the decision making process. In this modern individual oriented society, traffic information and travel advice of one fits all is becoming less acceptable. By using individual devices like smartphones, it is also becoming technically possible to provide personalized traffic information and travel advice.

From the aforementioned considerations, one can arrive at the following requirements:

1. Traffic management requires accurate predictions about the traffic situation for the whole network, different traffic management scenarios and whole (peak) periods.
2. Individual travellers need personalized multi-modal travel advice based on their preferences and habits.
3. A traffic management measure should not lead to the *perception* of travellers that they are worse off due to the measure or are harmed unfairly by it.

The third requirement tries to reconcile differences between the interests of traffic manager and individual travellers. Of course, it is impossible to satisfy all travellers. However, it might be possible to introduce measures such that travellers do not notice they are worse off and therefore do not change their behaviour, or such that travellers do not perceive the alternative they switch to as worse or unfair.

3 MOBILE SENSOR DATA

Mobile sensors like GPS and GSM are widely used in smartphones and navigation systems nowadays. Initially, they were used for navigation, but as their numbers increase, they are now also being used for estimating travel times on road trajectories (e.g., Google Traffic, 2013). There are however more applications: they can reveal travel patterns of individuals and groups of travellers.

In most countries, including the Netherlands, the understanding of people's travel behaviour is based on cross-sectional travel surveys where in general only one day is surveyed for each respondent in representative periods (Ortuzar et al., 2010). From these data, origin destination matrices, modal split (mode choice) and route choice are estimated and used in models that model urban traffic flows.

However, this is not enough to gain a proper understanding of the dynamics in travel behaviour. More specific, cross-section data do not give any information to ascertain how choices will vary over time (i.e. policy response) if the system changes. Studies with GPS-devices show a strong variation in multi-day travel behaviour (e.g., Stopher amd Zhang, 2011). People are shown to visit new places even after several months of monitoring (Schönfelder and Axhausen, 2010). Apart from determining destination and mode choice patterns over longer periods of time, GPS data are increasingly used to study route choice (e.g., Jan et al. 2000, Zhu and Levinson 2009, Papinski and Scott 2011).

Derived from standard economics it is often assumed in transport modeling that travelers are rational decision makers and have perfect knowledge on all available choice alternatives. There is increasing recognition that these assumptions are debatable. In reality, people may have limited knowledge and constrained cognitive abilities, leading to prejudiced reasoning and seeming randomness in choice behavior (e.g. Avineri and Prashker, 2004). This has been described as bounded rationality or satisficing behavior, first introduced by Herbert Simon (Simon, 1955) , and also found its way into transportation research (Mahmassani and Chang, 1987, Jayakrishnan et al., 1994). Since then, multiple studies suggested that these irrational behaviors are neither random nor senseless; they are systematic, consistent, repetitive, and therefore predictable (Tversky and Kahnemann, 1981, Ariely, 2009).

A well-known mechanism derived from the principles of bounded rationality is the notion of indifference band (Mahmassani and Chang, 1987). According to the theory of indifference bands, drivers will only alter their choice when a change in the transportation system or their trip characteristics, for example the travel time, is larger than some individual-situation-specific threshold.

More in general, travelers appear to make their decisions based on their perception of alternatives, which is biased according to the 'choice-supportive bias'. That is, people are more likely to attach positive feeling to options they choose and attribute negative features to options they reject (Mather et al., 2003, Henkel and Mather, 2007) even if that would be irrational. In terms of travel choices this suggests that travelers have different perceptions of options they frequently use than options they hardly use (Vreeswijk et al. 2013).

These findings may play an important role in future DTM, especially in fulfilling requirements 2 and 3. Although some travelers may be worse off when the overall network performance is optimized, it may be possible to choose DTM measures for which travelers do not perceive they are worse off or do not find this a problem. This will only be possible, however, when travelers get personalized travel advice based on their preferences and habits. For this, mobile sensor data appear to be indispensable.

4 ROADSIDE DEVICES

The use of smartphone, carrying among others a GPS-sensor, will probably rise in the coming years, enabling new data acquisition opportunities (Stopher, 2009, Nitsche et al., 2012). In addition, there already are numerous smartphone applications gathering personal travel data (e.g. UbiActive (Fan et al. 2012), Trip Analyzer (Li et al., 2011), and tripzoom (Bie et al., 2012)). Finally, smartphones or navigation devices are used as probes to estimate travel times on main roads. The question thus arises whether roadside devices are still necessary in the future.

To answer this question, we need to consider requirement 1 from section 2: "Traffic management requires accurate predictions about the traffic situation for the whole network, for different traffic management scenarios and over whole (peak) periods".

This requirement implies several things at the same time. First, information is needed on the traffic situation. This is much more than travel time alone. Policy makers are not only responsible for travellers, but also for the environment that is harmed by traffic. Pollution, noise hindrance and safety are important external factors which need to be considered, especially in dense residential areas or near locations that attract vulnerable groups such as schools. This implies that certain vulnerable, busy or economically important areas, locations or corridors, may need to be monitored continuously. Because many external effects depend on traffic intensity, this important quantity should be included in the monitoring.

Secondly, predictions are required for the whole network under various (possible) management scenarios. This implies that traffic intensities and travel times should be predictable when the traffic manager decides to increase or reduce the capacity of certain roads (for example by giving more or less green time). Because travel time shows a strong non-linear dependence on network intensities (demand) and capacities (supply), it is difficult to predict travel time when intensities and capacities are unknown, especially when small changes in intensity have a large effect on travel time. This is actually the case when it matters (i.e. in urbanized areas with a lot of traffic), while at the same time intensities are known to show strong variation within days and between days (Thomas et al., 2008). Accurate travel time predictions under varying (management) scenarios are therefore only possible when network demand and supply are predictable.

From this, it can be concluded that traffic management requires continuously monitoring of travel times, traffic intensities and capacities throughout the network or at least at, in or along important locations, areas or corridors.

At the moment this is not possible with mobile sensors. Mobile sensor (GPS) samples for public use are simply much too small. This may be changing (e.g. Rieser-Schüssler et al. 2012) somewhat, but the expectation is that, in general, public GPS samples will remain limited, (partly) due to privacy restrictions and commercial interests. In other words, large amounts of mobile sensor (GPS) data may remain out of reach for traffic management. Even if mobile sensor data would increase substantially for public use, there will always be some travelers missing from the data. For example, commuters will be less inclined to use navigation. Therefore, it can be questioned whether there will ever be enough mobile sensor (GPS) data to monitor traffic intensities and capacities of the important road sections with enough accuracy.

Roadside observations can fill this gap. In urban areas, single detection loops have long been used to measure occupation levels and intensities as input for traffic light operations. Network monitoring is more difficult with these data, because delays cannot easily be estimated in saturated conditions (when queues form near traffic lights), and individual vehicles cannot be followed through the network. As a result these measurements don't provide information on OD patterns and routes.

However, this is changing due to increasing use of cameras and new induction detection techniques that enable the identification of individual vehicles (Blokpoel and Vreeswijk, 2011). Thus, with these roadside devices located at important intersections, travel times, intensities and capacities can be measured directly throughout the network. Together with prediction algorithms like neural networks (e.g., Dharia, and Adeli, 2003, Yin et al. 2002), pattern matching models (e.g., Bajwa et al., 2004), extrapolation models (e.g., Wild , 1997, Chrobok et al., 2004, Thomas et al., 2009) or clustering models (e.g., Chung, 2003, Weijermars, 2007), more accurate traffic predictions of intensities and travel times will then become possible given certain management scenarios.

5 SYNTHESIS

As we have seen in the previous sections, traffic managers and travelers use different devices, i.e., roadside and mobile devices respectively, to acquire traffic information. Although mobile devises like smartphones with GPS become increasingly important, roadside devices might remain the main source of information for traffic management, because besides travel time they are able to provide accurate information on intensities and capacities.

The traditional use of traffic information by policy makers and travelers as shown in Figure 1 therefore remains quite realistic. The Figure shows two independent symmetric systems for both traffic management and the individual traveler. Both systems have objectives, i.e., policy objectives for the traffic manager and travel objectives for the traveler. By confronting these objectives with traffic and travel information respectively, traffic operations are set to manage the traffic system, and travel decisions are made to execute a trip. The traffic information is derived from data from roadside sensors, while travel information is derived from mobile sensor data.

Of course this Figure is a simplistic illustration of reality. The division between the use of roadside and mobile sensors is in reality not this strict, and a single box could in itself represent a complicated process with feedback loops. Traffic operations, for example, represent an interplay between instruments, such as traffic lights, variable message signs or route guidance panels, and traffic managers, while travel decisions may include more than only the traveller's behaviour. In fact, nowadays most travellers are assisted by travel apps that provide the traveller with information or advice. Travel apps are therefore implicitly included in travel decisions. The use of travel information in travel apps is more subtle than the figure indicates. Mobile sensor data of other users are used to provide reliable information on relevant travel modes and routes, while historical travel choices of the user may be used to personalize the advice. However, for the broader picture, these issues do not need to be considered in detail here.

The weak part of the traditional concept, as illustrated by Figure 1, is the lack of any interaction between traffic management and traveller. This leads to drawbacks regarding all three DTM requirements.

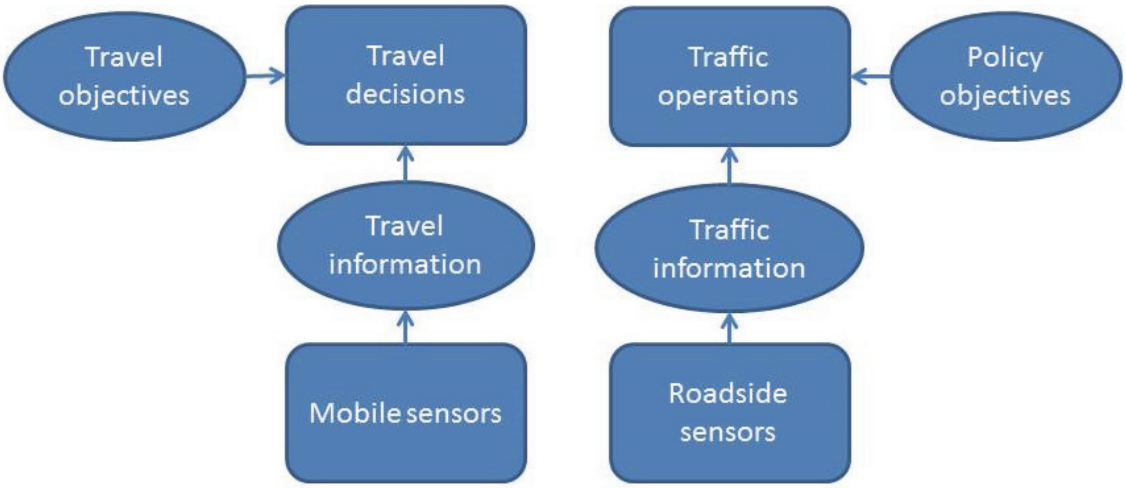


Figure 1: scheme of traditional use of mobile and roadside sensors.

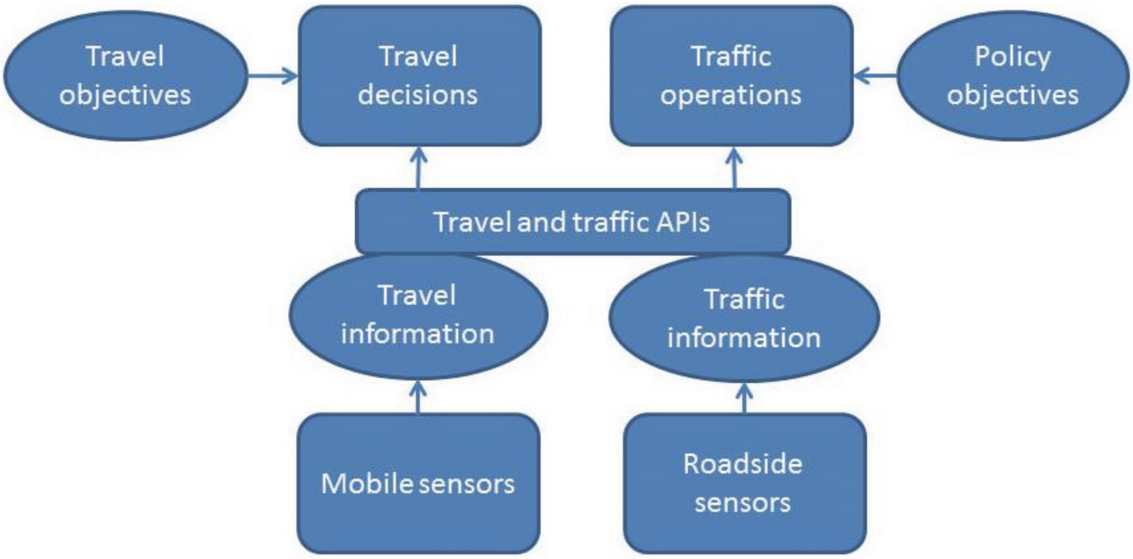


Figure 2: Scheme of integrating traffic and individual travel information on an operational level.

Surely, the usefulness of advice to the traveller would be enhanced by knowledge about (future) traffic operations (requirement 2). By the same token, the quality of traffic prediction would be increased when the intentions or likely future decisions of individual travellers are known to the traffic manager (requirement 1). Finally, without any interaction, there is no knowledge about (a change in) travellers' perceptions regarding certain traffic management measures (requirement 3).

In some projects such as SUNSET (Sunset, 2013) and I-zone (Veenstra et al., 2010), the information from travellers (mobile sensors) and traffic operators (roadside sensors) have been combined. This is shown schematically in Figure 2. Again, the architecture is more complicated in reality, and both projects comprise much more than combining different data sources. However, the general use of data in these projects is well captured by Figure 2. Information derived from roadside and mobile sensor data are combined in one large database. Third parties, mostly private companies, can retrieve this information via APIs, and can use this information in all kinds of apps they develop for travellers.

The main characteristic of such an architecture is sharing of underlying sensor data, and providing these data to the larger public. However, there are two main drawbacks of such an architecture regarding DTM. First, travel information on such

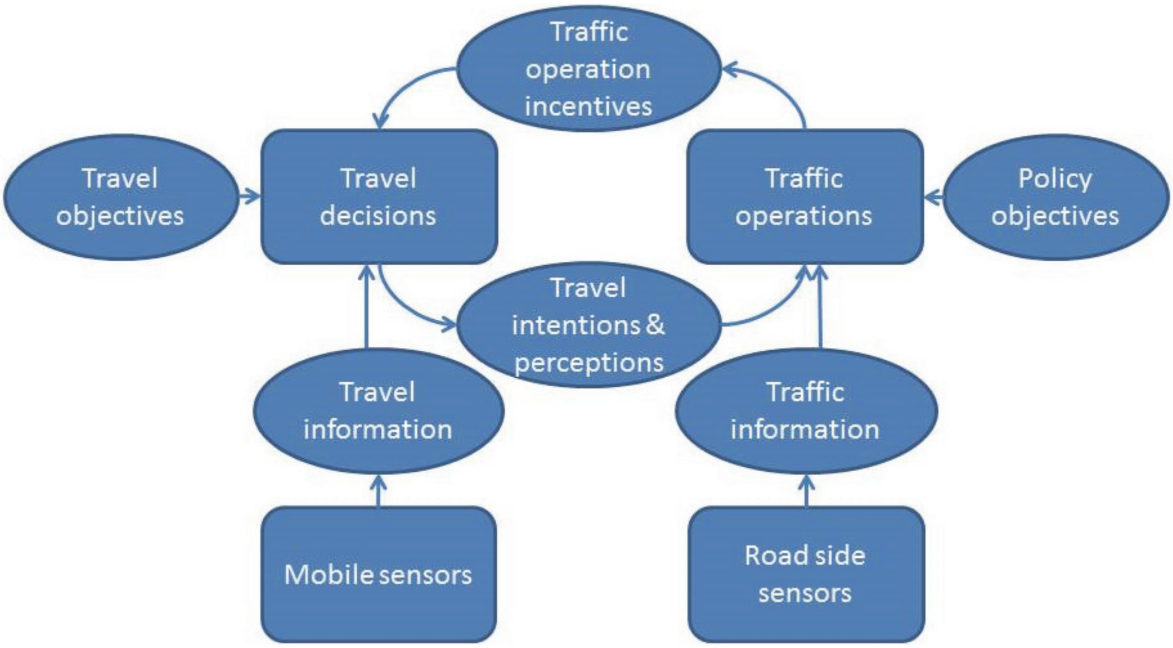


Figure 3: Scheme of integrating traffic and individual travel information on a strategic level.

an operational level is sensitive (regarding privacy) and is also regarded as quite valuable by companies that gather the data. It is therefore not very likely that these important players are willing to share their data. Secondly, these operational data do not provide the *intentions* of traffic operators and travellers. In fact, the main difference between roadside and mobile sensor data is not that they necessarily measure very different things (although not exactly the same either), but that they are used by very different users and for very different purposes. The real strength of sharing information would be on a higher, strategic, level such as shown in Figure 3.

In Figure 3, there is directly feedback between traffic operations and travel decision making. Based on traffic information from roadside devices and traffic policies, traffic operations are set to manage the traffic (such as in Figure1). However, information about travellers' reactions to and perceptions of certain management measures are provided to the traffic operators and are used to improve the management scenarios. At the same time, updated management scenarios are provided to travellers(' apps), enabling travellers to take specific traffic measures (including possible incentives for favourable behaviour) into account when planning their trip.

Instead of sharing traffic data, the main characteristic of this DTM vision is sharing of intentions, plans and measures between traffic operators and travellers. The corresponding architecture would connect well to the DTM requirements mentioned in section 2. The question would then be: what kind of information is exactly shared and how is this information shared? Should the information include detailed operational data such as green times of individual traffic lights given specific inflow intensities, or should the information be provided on a more aggregated and strategic level? To answer these questions follow up research is needed, preferably in a European project with as main aim to develop such a future DTM system for main European cities.

REFERENCES

Ariely, D., 2009. *Predictably Irrational - Revised and expanded edition*. London: HarperCollingsPublishers.

Avineri, E., Prashker, J. N., 2004. Violations of expected utility theory in route-choice stated preferences, *Transportation Research Record,* vol. 1894, pp. 222-229.

Bajwa, S. I., Chung, E., Kuwahara, M., 2004. An adaptive travel time prediction model based on pattern matching, In *11th World Congress on Intelligent Transport Systems*, Nagoya, Japan.

Bie, j., Bilsma, M., Broll, G., Cao, H., Hjalmarsson, A., Hodgson, F., Holleis, P., Van Houten Y., Jacobs, K., Koolwaaij, J., Kusumastuti, D., Luther, M., 2012. Move Better with tripzoom, *International Journal on Advances in Life Sciences*, vol. 4, no. 3&4, pp. 125 – 135

Blokpoel, R., Vreeswijk, J., 2011. Vehicle inductive profile for incident detection, In *Proceedings of the 18th World Congress on Intelligent Transport Systems*, Orlando, United States.

Chrobok, R., Kaumann, O., Wahle, J., Schreckenberg, M., 2004. Different methods of traffic forecasting based on real data, *European Journal of Operational Research*, vol. 155, pp. 558-568

Chung, E., 2003. Classification of traffic pattern,In *Proceedings of 10^th World Congress on Intelligent Transport Systems*, Madrid.

Dharia, A., Adeli, H., 2003. Neural network model for rapid forecasting of freeway link travel time, *Engineering Applications of Artificial Intelligence*, vol. 16, pp. 607-613.

Fan, Y., Chen, Q., Douma, F., Liao, C.-F., 2012. Smartphone-Based Travel Experience Sampling and Behavior Intervention among Young Adults. *Intelligent Transportation Systems Institute*, Center for Transportation Studies, University of Minnesota.

Google Maps. Maps.google.com Accessed: 2013

Hasberg, P., Serwill, D., 2000. Stadtinfoköln – a global mobility information system for the Cologne area, In *7^th World Congress on Intelligent Transport Systems*, Turin, Italy.

Henkel, L. A., Mather, M., 2007. Memory attributions for choices: How beliefs shape our memories, *Journal of Memory and Language,* vol. 57, pp. 163-176.

Jan, O., Horowitz, A.J., Peng, Z.R., 2000. Using global positioning system data to understand variations in path choice. *Transportation Research Record*, vol. 1725, pp. 37-44.

Jayakrishnan, R., Mahmassani, H. S., Hu, T. Y., 1994. An evaluation tool for advanced traffic information and management systems in urban networks, *Transportation Research Part C,* vol. 2, pp. 129-147.

Kellerman, A., Schmid, A., 2000. Mobinet: Intermodal traffic management in Munich –control centre development, In *7^th World Congress on Intelligent Transport Systems*, Turin, Italy.

Leitsch, B., 2002. A Public-privat partnership for mobility – Traffic management Center Berlin, In *9^th World Congress on Intelligent Transport Systems*, Chicago, USA.

Li, M., Dai, J., Sahu, S., Naphade, M., 2011. Trip analyzer through smartphone apps, In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ACM, 537-540.

Mahmassani, H. S., Chang, G.-L., 1987. On boundedly rational user equilibrium in transportation networks, *Transportation Science,* vol. 21, pp. 89-99.

Mather, M., Shafir, E., Johnson, M. K., 2003. Remembering chosen and assigned options, *Memory and Cognition,* vol. 31, pp. 422-433.

Nitsche, P., Widhalm, P., Breuss, S., Maurer, P., 2012. A Strategy on How to Utilize Smartphones for Automatically Reconstructing Trips in Travel Surveys. *Procedia - Social and Behavioral Sciences,* vol. 48**,** pp. 1033-1046.

Ortúzar, J. D. D., Armoogum, J., Madre, J.-L., Potier, F., 2010. Continuous Mobility Surveys: The State of Practice. *Transport Reviews,* vol. 31**,** pp. 293-312.

Papinski, D., Scott, D.M., 2009. A GIS Toolkit for route choice analysis, *Journal of Transport Geography*, vol. 19, pp. 434 – 442.

Rieser-Schüssler, N., Balmer, M., Axhausen, K.W., 2012. Route choice sets for very high-resolution data, *Working paper Transport and Spatial Planning*, EHT, Zürich, Switzerland

Schönfelder, S., Axhausen, K. W., 2010. *Urban Rhythms and Travel Behaviour. Spatial and T*emporal *Phenomena of Daily Travel*, Farnhem/Burlington, Ashgate Publishing.

Simon, H. A., 1955. A behavioural model of rational choice, *Quarterly Journal of Economics,* vol. 69, pp. 99-118.

Stopher, P. R., 2009. The travel survey toolkit: where to from here? *Transport survey methods, keeping up with a changing world***,** pp. 15-46.

Stopher, P.R., Zhang, Y., 2011. The repetitiveness of daily travel, *Transportation Research Board Annual Meeting*, Washington DC, USA.

Sunset, http://sunset-project.eu/ Accessed: 2013

Thomas, T., Weijermars, W. A. M., Van Berkum, E. C., 2008. Variations in urban traffic volumes, *European Journal of Transport and Infrastructure Research*, vol. 8. no. 3, pp. 252-264.

Thomas, T., Weijermars, W. A. M., and Van Berkum, E. C., 2010. , *IEEE transactions on intelligent transport systems,* vol. 11, no. 1, pp. 71-80.

Tversky, A., Kahnemann, D., 1981. The framing of decisions and the psychology of choice, *Science,* vol. 211, pp. 453-458.

Veenstra, S.A., Teeuw, W.B., Meeuwissen, M., 2010. Incentive Zone Enschede; Het verleiden tot gedragverandering op basis van gepersonaliseerde, multi-modale mobiliteitsinformatie en –advies, *Colloquium Vervoersplanologisch Speurwerk (CVS 2010)*, Roermond, The Netherlands.

Vreeswijk, J. D., Thomas, T., Van Berkum, E. C., Van Arem, B., 2013. Drivers' perception of route alternatives as indicator for the indifference band. In *Proceedings of 92th Annual Meeting of the Transportation Board*, Washington DC, USA.

Weijermars , W. A. M., 2007. *Analysis of urban traffic patterns using clustering*, Ph.D. Thesis, University of Twente, Enschede, The Netherlands.

Wild, D., 1997. Short-term forecasting based on a transformation and classification of traffic volume time series, *International Journal of Forecasting*, vol. 13, pp. 63-72.

Yin H., Wong, S. C., Xu, J., Wong, C. K., 2002. Urban traffic flow prediction using a fuzzy-neural approach, *Transportation Research Part C*, vol. 10, pp. 85-98.

Zhu, S., Levinson, D., 2010. Do people use the shortest path? An empirical test of Wardrop's first principle, In *Conference proceedings of the 4th International Symposium on Transportation Network Reliability*, Minneapolis, USA.

Special Session on Intelligent Systems and Business Analysis

The ERP Systems in Modern Business and Corporate Management

Boris A. Staykov, Tatiana V. Atanasova, Vladimir V. Monov and Lyubka A. Doukovska
Institute of Information and Communication Technologies, Bulgarian Academy of Sciences,
Acad. G. Bonchev str., bl. 2, 1113 Sofia, Bulgaria
bstaykov@iinf.bas.bg, {atanasova, vmonov, doukovska}@iit.bas.bg

Keywords: Enterprise Resource Planning (ERP), Management, Integration, Business Processes.

Abstract: The purpose of this paper is to make an analysis and assessment of the increasing need of using sophisticated software systems for managing business processes. In our days the most important thing is the information and the success of every business depends on how fast and effective managers deal with information. Business processes usually include manufacturing, sales order management, inventory and warehouse control, logistics, advanced forecasting and planning, financial management, customer relationship management and many others, all of them connected and depending on each other. It is practically impossible to manage all those processes effective without using software that gives a convenient, easy understandable and user friendly interface to the end user. End user of such system can be anybody from top to the bottom level of the company hierarchy. It is commonly recognized that ERP (Enterprise Resource Planning) systems are the software systems that provide the functionality to manage all that information. A major issue for these systems is that these processes are very different and specific for each business and company. Because of this there is impossible to create one universal ERP system that is good enough for everyone. Instead of this, software companies produce core software which can be expanded and tuned according to each specific company. This automatically includes a process of integration that is a key part of the ERP system.

1 INTRODUCTION

Nowadays when we speak about business modelling and software information systems, it naturally comes to ERP systems as the most significant, large, complex and sophisticated implementation of information technologies into the contemporary enterprises, business companies and corporations. But what exactly is or isn't an ERP software system? First of all we can say that it isn't software in the traditional meaning that we all know – programme that you can buy, install, run and start to use right away. ERP is a large sophisticated software solution that combines many parts (modules) that are used to manage different business processes in certain company or enterprise. The development of such a system is long and difficult process that involves many people – software engineers; programmers; business consultants; even field workers from all levels of the hierarchy of the client company that will be using the ERP system. In fact there are no two absolutely identical ERP systems. That is because each ERP solution is designed and developed for one exact client company, according to their business models, processes, needs and desires. That's why they are very expensive (the investment for ERP system can cost to the client company as high as hundreds of thousands of US dollars) and complex because they integrate many functional and cross-functional business processes. Typical ERP systems support Operations (Production), Human Resources, Finance & Accounting, Sales & Distribution, and Procurement (Magal, 2012). As mentioned before – the development of an ERP system is a long demanding task, involving many specialists and unfortunately there is no guarantee for success at the end.

Many ERP implementations have been classified as failures because they did not achieve predetermined corporate goals (Umble, 2003). The process of implementation of one ERP system has few major steps and can take between 6 months and 1 year of time! Those steps are:

- Analyzing the company business processes and creating abstract models of each one of them and the corresponding relationships between them.

- Designing database models that contain all the necessary information for those models and relationships.
- Designing the actual software. This includes user interfaces, security management system, backup and archiving systems and mechanisms etc.
- Integration of the system and training the end users how to work with the software.

Each one of these steps involves constant dialogs and communication between the ERP vendor, Client Company, different consultants in certain areas, software engineers and end users. Should any of these steps fails, there is a very high probability that the whole project will fail and lead to huge financial losses for the vendors and the clients.

But when implemented and integrated correctly and precisely, one ERP system gives to the company very powerful tools for running their business in the most effective way. That means sharp resource planning and decision making using analytical instruments; better fast and effective communication and coordination between company departments and external counteragents; efficient accounting and warehouse management; minimizing loses and abuses of any kinds.

2 EVOLUTION OF THE ERP SYSTEMS

To understand how the contemporary complex ERP systems started to exist, we will follow briefly the natural software evolution during the past 50 years, using an article on implementation procedures in ERP systems, written in 2002 by Elisabeth Umble, Ronald Haft, and Michael Umble.

During the 1960's the use of software technologies was mainly for inventory control. Companies could afford to keep lots of ''just-in-case''inventory on hand to satisfy customer demand and still stay competitive. Consequently, techniques of the day focused on the most efficient way to manage large volumes of inventory (Umble, 2003). Most of the software packages then were designed and served the purpose for more efficient inventory control and warehouse management (Ptak, 2000, Shankarnarayanan, 2000).

In the 1970's, it became increasingly clear that companies could no longer afford the luxury of maintaining large quantities of inventory. This led to the introduction of material requirements planning (MRP)systems. MRP represented a huge step forward in the materials planning process. For the first time, using a master production schedule, supported by bill of material files that identified the specific materials needed to produce each finished item, a computer could be used to calculate gross material requirements. Using accurate inventory record files, the available quantity of on-hand or scheduled-to-arrive materials could then be used to determine net material requirements. This then prompted an activity such as placing an order, cancelling an existing order, or modifying the timing of existing orders. For the first time in manufacturing, there was a formal mechanism for keeping priorities valid in a changing manufacturing environment (Umble, 2003). Later the MRP systems expanded to "closed loop MRP" (Oden, 1993), that besides inventory planning included also tools for planning the production levels, sales planning and scheduling, making business promises to customers, forecasting and different analysis tools.

In the 1980's, companies began to take advantage of the increased power and affordability of available technology and were able to couple the movement of inventory with the coincident financial activity. Manufacturing resources planning (MRP II) systems evolved to incorporate the financial accounting system and the financial management system along with the manufacturing and materials management systems. This allowed companies to have a more integrated business system that derived the material and capacity requirements associated with a desired operations plan, allowed input of detailed activities, translated all this to a financial statement, and suggested a course of action to address those items that were not in balance with the desired plan (Ptak, 2000).

By the early 1990s, continuing improvements in technology allowed MRP II to be expanded to incorporate all resource planning for the entire enterprise. Areas such as product design, information warehousing, materials planning, capacity planning, communication systems, humanresources, finance, and project management could now be included in the plan. Hence, the term, ERP was coined (Ghosh, 2012).

Since then the ERP systems are becoming larger, more sophisticated and they are being integrated in enterprises and companies of all sizes – large business corporations, medium and small business enterprises.

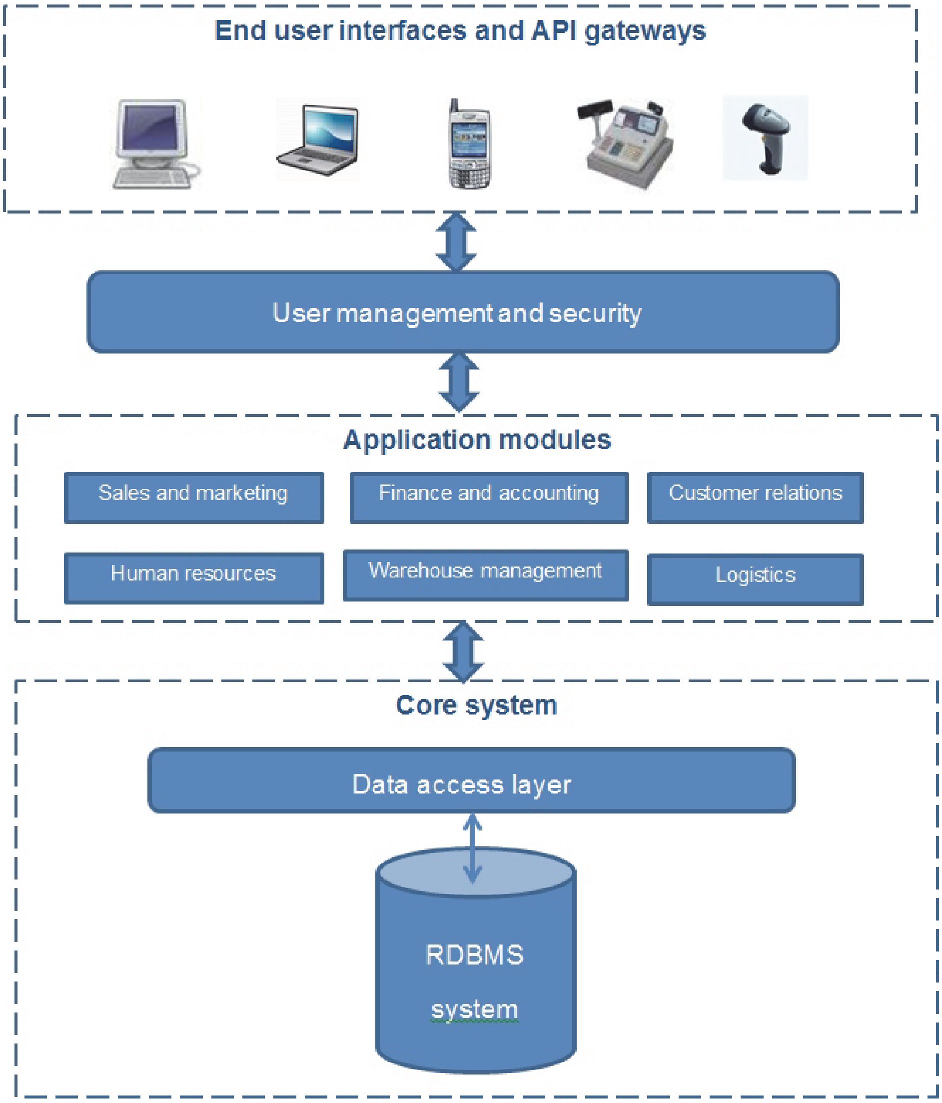


Figure 1: Common ERP system structure.

3 STRUCTURE OF THE ERP SYSTEMS

All contemporary ERP systems are client-server based applications and they allow interaction with and operation from many types of computer devices – desktop or mobile computers, industrial mobile devices, POS (point of sale) devices and even cell phones. Of course this is impossible without using the Internet, VPN and LAN networks. Also because of the big variety of devices and operating systems that interact with the ERP systems, the most convenient and maybe the only way of implementing ergonomic, user friendly and convenient user interface is to use web based technologies.

The web interface is universal,platform independent and the only thing that one device is required to have in order to interact with the ERP modules is the web browser.

Key feature of successful ERP systems are to be flexible – in order to respond to the constantly changing needs of an enterprise; to be modular – that means that different functionalities must be logically separated in the system, which allows them to be detached, modified and attached back to the system without affecting unintentionally any other functionality or part of the whole system; comprehensive – must support wide range of business processes within one or more organisations. Beyond the company – it should support online connectivity and information exchange with other business-related companies (usually using web service API interfaces).

As mentioned before – the ERP system is not single software that can be produced once and distributed as is to various companies, but is a complex modular system that is being engineered and developed according the company needs. Therefore the ERP vendors are using their own ERP development frameworks – software environment that allows them fast and efficient implementation of the software system. Most significant vendors of this type of software are SAP, Oracle, Microsoft, SAGE and Info Global (IBM) (Low, 2013).

The common structure of ERP systems is shown on Figure 1. It consists of database management system (RDMS) – where all the information is stored and managed; data access layer; application layer which consists of various modules that represent different business processes and activities, analytical tools and other instruments and tools; user management and security layer; end user interfaces.

4 THE IMPORTANT ROLE OF THE ERP SYSTEM FOR THE BUSINESS

As regarded in this article, integrating well designed and developed ERP system into enterprises and companies has a crucial role to their successful business. With today's information variety and amount, constantly increasing market needs and demands, complex services, intra and inter-companies processes and relationships, it's impossible to be successful without actively and even aggressively using the power of the contemporary information technologies. Each stage of one's enterprise activity must be precisely planned, implemented, monitored and analyzed. ERP systems combine everything in a way so that managers can have wide view of the whole picture in front of them, so they can make effective and in time decisions. Efficient warehouse management is important for materials ordering and production planning. This is directly connected to customer management, logistics, effective distribution and service support. Efficient accounting and financial planning depends on how fast and precise the information about all those business processes is maintained, presented and analyzed.

The most important benefits of using ERP systems, based on Kay Roman detailed research article (Roman, 2009) are as follows:

1. Enhanced Technology – the old legacy software systems can't meet the current technology needs. ERP systems allow speeding up all operations;
2. Efficiency in Processes – The new system will eliminate labour-intensive manual processes and current duplication of efforts; will streamline critical business processes for many departments; will make data collection better and more efficient;
3. Integrated and consistent information - A major benefit of a new system will be replacing multiple, disconnected databases with a single, integrated system;
4. Easier Reporting - An advantage of a new system will be improved and more customizable reporting;
5. User Friendliness of the System - A new system will provide easier access to information and overall improvements in its use for the functional staff;
6. Access to Data - Increased self-service for data needs, real time data, Better access to information and decision making;
7. Ability to Provide Better Customer Service - Speedier and more applicant friendly service;
8. Increased Functionality - Better functionality for users;
9. Better Communications- Promotes more collaboration with business partners;
10. Increased Security of Data - Data restrictions can be enhanced.

5 CONCLUSIONS

ERP systems have a key role for successful business in modern enterprises and business companies. They are natural evolution of the general software systems and the use of constantly expanding nowadays information technologies. ERP are the most complex and sophisticated software systems, developed for certain company according to its needs and specific business processes. Therefore their integration is a long difficult process, involving many specialists and consultants. If not planned and designed precisely and carefully, such a project can be easily turned into a failure and huge financial loses. When done correctly, once integrated this system brings big benefits to the companies, allowing them better planning, managing forecasting and expanding their successful business.

ACKNOWLEDGEMENTS

The research work reported in the paper is partly supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions).

REFERENCES

Magal Simha R., Jeffrey Word, Integrated Business Processes with ERP Systems, 2012.

Umble Elisabeth J., Ronald R. Haft, Michael M. Umble, Enterprise resource planning: Implementation procedures and critical success factors, European Journal of Operational Research 146, 241–257, 2003.

Ghosh Rittik, A Comprehensive Study on ERP Failures Stressing on Reluctance to Change as a Cause of Failure, Journal of Marketing and Management, 3 (1), 123-134, 2012.

Ptak C., E. Schragenheim, ERP: Tools, Techniques, and Applications for Integrating the Supply Chain, St. Lucie Press, Boca Raton, FL, 2000.

Shankarnarayanan S., ERP systems – using IT to gain a competitive advantage, 2000.

Oden H., G. Langenwalter, R. Lucier, Handbook of Material and Capacity Requirements Planning, McGraw-Hill, New York, 1993.

Low Terry, Top ERP systems in 2013. http://www.dvalde.com/top-erp-systems-in-2013.html

Roman Kay, Benefits of Implementing an ERP, Collegiate Project Services 2009. http://www.collegiateproject.com

Innovation Cycles Control through Markov Decision Processes

Vassil S. Sgurev, Stanislav T. Drangajov, Lyubka A. Doukovska and Vassil G. Nikov
Institute of Information and Communication Technologies, Bulgarian Academy of Sciences,
Acad. G. Bonchev Str., Bl. 2, 1113 Sofia, Bulgaria
sgurev@bas.bg, sdrangajov@gmail.com, doukovska@iit.bas.bg, vasilnikov@abv.bg

Keywords: Innovation Introduction, Markov Decision Processes.

Abstract: Innovations are introduced in several cycles, or steps which are of stochastic character. Successful completion of each cycle results in the beginning of the next one. Initial stages are connected with expenses of risk (venture) capital and the investments are returned in the final stages, usually with quite big profit. A helpful approach for control of the innovation process is the use of Markov decision processes which have proved to be an efficient tool for control of multi state stochastic processes. Those stages may be summarizes as: 1 – prestart stage; 2 – start stage; 3 – initial expansion stage; 4 – quick expansion stage; 5 – stage of reaching liquidity of venture investments; 6 – stage of project failure and its cancelling. The transition from state to state may be controlled through control techniques of Markov Decision Processes so that maximum profit is achieved in shortest time. The stages are conditional and some of them may be united, e.g. 1 and 2, or 3 and 4.

1 INTRODUCTION

It is known that the innovations' introduction through the respective innovation cycles as a rule is accompanied with considerable uncertainty and it is of definitely expressed stochastic character. As the successful completion of each innovation project very often results in considerable profit this stimulates the investment of considerable venture (risk) means. A very important task arises for preliminary careful considering and calculating the stochastic character of the on going processes.

A multi step discrete Markov decision process with mixed policies is proposed in the present work, for the innovation risks interpretation. The innovation process is accomplished, and probably finished, as a rule, in a cycle of the following 6 stages: 1 – prestart and start stage; 2 – initial expansion stage; 3 – quick expansion stage; 4 – preparatory stage; 5 – stage of reaching liquidity of the venture investment; 6 – stage of project failure and its liquidation (Grossi, 1990, Cormican, 2004, Bernsteina, 2006). Besides, the process at each stage may be in different states where the decision maker may undertake different actions which result in the transition to a new state with respective profits and losses. The first three stages are connected with initial investments and respective losses. The objective is they to be minimized. The last three stages may generate profit and ensure full return of the investments and considerable gains, but it may also result in considerable loss if the innovation product is a failure. It is to be clearly noticed that the innovation introduction is a risky enterprise and not each attempt is successful and winning.

It should be explicitly noticed that the innovation process may only pass from a given stage to the next one and can never return to a previous stage. No other stages except the last ones – success or failure, are absorbing - i.e. the innovation process may not stay for ever in any of the initial stages or it fails. The process may stay in a given stage for some time. It is a responsibility of the decision maker to undertake such control actions that the process leaves as soon as possible the first three stages, which generate expenses, with min losses and reaches the final stage, which generate profit.

It is to be also noted, that depending on the decision makers actions a stage may be omitted, e.g. to pass directly from stage r to stage $r+2$. I.e. stages so described are to some degree conditional but nonetheless the process may develop in only forward direction.

2 MARKOV DECISION MODEL FOR THE INNOVATION PROCESS

We consider an innovation process, which might be at any of the six stages of implementation of a new product. Of course this is for purposes of methodology. In fact one should begin from the first stage and reach the last one.

We introduce the following denotation: $N_j = \Gamma_j^{-1}$ where the right hand part of the upper equation is a reverse mapping of node *j* of the graph from Figure 1. P_{ij}^k denotes the transition probability of the innovation process to pass from state $i \in N$ to state $j \in N$ when using control $k \in K_i$, where K_i is the set of possible policies from state *i*. As leaping across or going back to stages of the innovation process is impossible, then:

$$P_{ij}^k = \begin{cases} \geq 0, i \in N_j; j \in N; k \in K_i; \\ = 0, otherwise; \end{cases}$$

$$0 \leq P_{ij}^k \leq 1; \sum_{j \in N} P_{ij}^k = 1; i \in N; k \in K_i.$$

By x_i^k will be denoted the probability the innovation process to fall in state i, at using control $k \in K_i$ from this state.

An important feature of the innovation process is that at transition from one stage to the next one in the first three stages resources are spent, and the transition from stage to the other in the last three stages increasing profit is gained, i.e.:

$$r_i^k = \begin{cases} \leq 0; i \in \{1,2,3\}; k \in K_i; \\ \geq 0; i \in \{4,5,6\}; k \in K_i. \end{cases} \quad (1)$$

Then the maximum restoration of the venture funds initially invested will be obtained at optimal choice of control actions from each possible state of the process, i.e.:

$$\{k^* \in K_i \,/\, i \in N_r\}.$$

This optimal control selection from the separate states corresponds to maximization of the objective function:

$$\sum_{j \in N} \sum_{k \in K_j} r_j^k x_j^k \to \max \quad (2)$$

Different methods of linear and dynamic programming (Mine, 1975) may be used for finding the optimal solution of the objective function above with the existing linear probability constraints.

The specific structure of the proposed here multi step discrete Markov decision process corresponds to a sufficient degree to the processes of realization of innovations and provides possibilities for efficient control of venture financing of innovations at their realization.

3 NUMERICAL EXAMPLES

Next Figure 1 illustrates a Markov Decision Process for control of the development of an innovation through the 6 stages. The set of arcs *U* show the possible transition from one stage (state) of the innovation process to another one. The denotations on the arcs of the decision graph should be decoded as follows:

$P_{i,j}^{k_l}$ - the probability for transition from stage *i* to stage *j* using control action k_l in state *i*.

In the final two stages, 5 and 6, which are ergodic there is one only possible action. At the other stages we accept for illustration that there are two possible control actions with the respective transition probabilities.

Formally this means that the possible actions $\{k_j\}$ in each state $j \in N$ are defined in the following way:

$$k_j = \begin{cases} \{1\}, j \in \{5,6\}; \\ \{1,2\}, j \in N \setminus \{5,6\} \end{cases}$$

The initial probability the process to be in state $i \in N$ is equal to:

$$a_i = \begin{cases} 1, i = 1; \\ 0, otherwise. \end{cases}$$

The problem for finding optimal policies for the Markov Decision Process shown in Figure 1 may be reduced to the following linear programming problem with objective function (2) and constraints:

$$\sum_{k \in K_j} x_i^k = 1, \text{ if } i = 1 \quad (3)$$

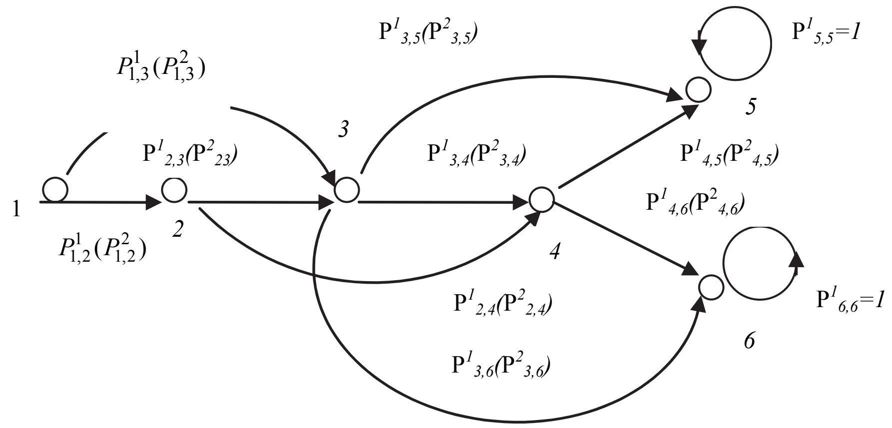


Figure 1: Exemplary transition graph for an innovation process.

$$\sum_{k \in K_j} x_j^k - \sum_{j \in N_j} \sum_{k \in K_i} P_{ij}^k x_i^k = 0 \text{, if } j \in N \quad (4)$$

$$x_j^k \geq 0 \text{, if } k \in K_j; j \in N \quad (5)$$

If we take as a base the graph in Figure 1, then the equations (2) and (3) to (5) will acquire the following form:

$$r_1^1 x_1^1 + r_1^2 x_1^2 + r_2^1 x_2^1 + r_2^2 x_2^2 + r_3^1 x_3^1 + r_3^2 x_3^2 + r_4^1 x_4^1 + r_4^2 x_4^2 + r_5^1 x_5^1 + r_6^1 x_6^1 \rightarrow \max \quad (6)$$

under the constraints:

$$x_1^1 + x_1^2 = 1 \quad (7)$$

$$x_2^1 + x_2^2 - P_{1,2}^1 x_1^1 - P_{1,2}^1 x_1^2 = 0 \quad (8)$$

$$x_3^1 + x_3^2 - P_{1,2}^1 x_1^1 - P_{1,3}^1 x_1^2 - P_{2,3}^1 x_2^1 - P_{2,3}^2 x_2^2 = 0 \quad (9)$$

$$x_4^1 + x_4^2 + P_{2,4}^1 x_2^1 - P_{2,4}^2 x_2^2 - P_{3,4}^1 x_3^1 - P_{3,4}^2 x_3^2 = 0 \quad (10)$$

$$x_5^1 - P_{3,5}^1 x_3^1 - P_{3,5}^2 x_3^2 - P_{4,5}^1 x_4^1 - P_{4,5}^2 x_4^2 = 0 \quad (11)$$

$$x_6^1 - P_{3,6}^1 x_3^1 - P_{3,6}^2 x_3^2 - P_{4,6}^1 x_4^1 - P_{4,6}^2 x_4^2 = 0 \quad (12)$$

$$x_5^1 + x_6^1 = 1 \quad (13)$$

$$x_j^k \geq 0; k \in K_j; j \in N \quad (14)$$

Let the profits (expenses) $\{r_i^k\}$ have the following values:

$$r_1^1 = -10; r_1^2 = -11; r_2^1 = -5; r_2^2 = -6;$$
$$r_3^1 = 7; r_3^2 = 8; r_4^1 = 10; r_4^2 = 12;$$
$$r_5^1 = 0; r_6^1 = 0.$$

The transition probability values are defined in the following table:

Table 1: Transition probabilities.

STATE 1				STATE 2			
Policy 1		Policy 2		Policy 1		Policy 2	
$P_{1,2}^1$	0,8	$P_{1,2}^2$	0,9	$P_{2,3}^1$	0,7	$P_{2,3}^2$	0,8
$P_{1,3}^1$	0,2	$P_{1,3}^2$	0,1	$P_{2,4}^1$	0,3	$P_{2,4}^1$	0,2
STATE 3				**STATE 4**			
Policy 1		Policy 2		Policy 1		Policy 2	
$P_{3,4}^1$	0,6	$P_{3,4}^2$	0,5	$P_{4,5}^1$	0,8	$P_{4,5}^2$	0,7
$P_{3,5}^1$	0,3	$P_{3,5}^2$	0,3	$P_{4,6}^1$	0,2	$P_{4,6}^2$	0,3
$P_{3,6}^1$	0,1	$P_{3,6}^2$	0,2				
STATE 5				**STATE 6**			
Policy 1		-		Policy 1		-	
$P_{5,5}^1$	1	-		$P_{6,6}^1$	1	-	

The respective transition probabilities are shown above the arcs of the graph shown in Figure 1, when using different possible policies. If only policy 1 or respectively – only policy 2 is used, then the transition probabilities tables will have the following form:

Table 2: Transition probabilities for policy 1.

	1	2	3	4	5	6
1.	0	0,8	0,2	0	0	0
2.	0	0	0,7	0,3	0	0
3.	0	0	0	0,6	0,3	0,1
4.	0	0	0	0	0,8	0,2
5.	0	0	0	0	1	0
6.	0	0	0	0	0	1

Table 3: Transition probabilities for policy 2.

	1	2	3	4	5	6
1.	0	0,9	0,1	0	0	0
2.	0	0	0,8	0,2	0	0
3.	0	0	0	0,5	0,3	0,2
4.	0	0	0	0	0,7	0,3
5.	0	0	0	0	1	0
6.	0	0	0	0	0	1

Table 2 reflects transition probabilities for policy 1 and Table 3 – for policy 2 respectively.

At least two classes may be distinguished in this matrix – one quasi block diagonal ergodic, and one absorbing, corresponding to states 5 and 6. When the process being controlled falls in one of the latter states it remains there for ever.

At defining the optimal control through relations (5) to (13) in the rows of both matrices #3 and #4 will be used, in general with different probabilities, i.e. both pure and mixed policies will be used, as seen in the solving of the particular problem.

The linear programming problem (6) to (14) includes 10 variables $\{x_j^k\}$ and 8 constraints. Its solution results in the following optimal values of the variables:

Table 4: Linear programming problem solution.

Variables	Optimal values
x_1^1	1
x_1^2	0
x_2^1	0,8
x_2^2	0
x_3^1	0,76
x_3^2	0
x_4^1	0
x_4^2	0,696
x_5^1	0,7152
x_6^1	0,248

It is seen from the table above, that in the example considered the optimal solution leads to pure optimal policies of both types – 1 or 2. Next table shows the optimal pure policy and the respective optimal strategy.

Table 5: Optimal pure policy and respective strategy.

State $i \in N$	1	2	3	4	5	6
Optimal policy $k^* \in K_i$	1	1	1	2	1	1
Optimal strategy $\{k^* \in K_i \,/\, i \in N\}$	{1, 1, 1, 2, 1, 1}					

The following matrix of the achieved optimal transition probabilities of the Markov process may be drawn up on the base of the optimal policies.

The new (optimal) transition probabilities matrix thus constructed also consists of a quasi diagonal ergodic class and an absorbing class of two states. In it one row (the fourth) of Table 4 is used, corresponding to policies 2. The remaining rows are from Table 3, corresponding to policies 1. In this sense it is mixed by using both policies – 1 and 2.

Table 6: Optimal transition probabilities matrix.

	1	2	3	4	5	6
1.	0	0,8	0,2	0	0	0
2.	0	0	0,7	0,3	0	0
3.	0	0	0	0,6	0,3	0,1
4.	0	0	0	0	0,7	0,3
5.	0	0	0	0	1	0
6.	0	0	0	0	0	1

The Markov process thus constructed will flow step by step according to the transition probabilities from Table 6. In the next Figure 2 its stochastic parameters are shown for the purpose of clearness – on the arcs the respective probabilities $\{P_{ij}\}$ are shown for falling from the initial state 1 into state $j \in N$ at passing through the previous state $i \in N$, and in squares next to the vertices the final probabilities $\{\pi_{i,j} \,/\, j \in N\}$ are shown for the process to fall from the initial state 1 into the corresponding state $j \in N$.

The final probabilities are also shown in the following table:

Table 7: Final probabilities.

Final prob.	$\pi_{1,1}$	$\pi_{1,2}$	$\pi_{1,3}$	$\pi_{1,4}$	$\pi_{1,5}$	$\pi_{1,6}$
Values	1	0,8	0,76	0,69	0,72	0,28

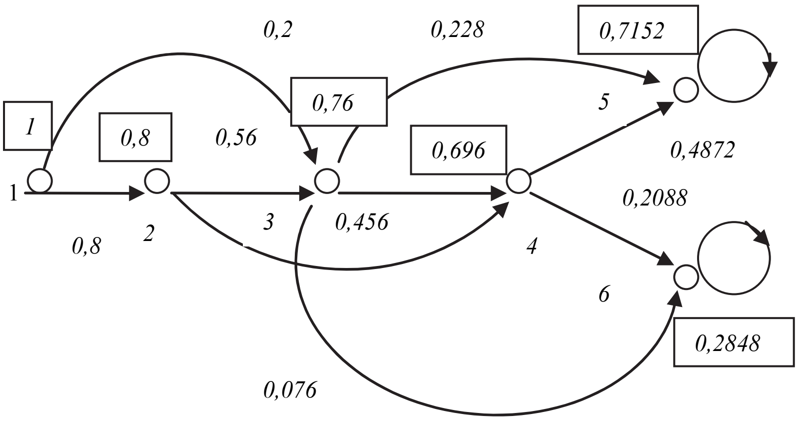


Figure 2: Markov process stochastic parameters from state to state and in the states.

On the base of the optimal values of the variables $\{x_j^k\}$ of Table 4 through (6) the maximum value of the objective function is computed to be - 0,328.

The results obtained provide the possibility some conclusions to be made:

I. When teaching one of the two final states – 5 or 6 the investment made is not paid off in full as 0,328 units remain to be paid off. If the process has fallen in state 5, then the project is successful and in may go on further to pay off the investments made and to produce profit. In case that the process fell in state 6, the project is a failure and it is almost sure it will be cancelled. The amount of 0,328 units should be registered as a loss in this case.

II. Even at optimal decisions for leading the stochastic innovation process, the end the end of the project cannot be certainly predicted – a considerable probability (in the case considered almost 0,3) exists it to end as a failure. This reflects the real conditions in similar class of processes, which are always of explicitly expressed stochastic character.

III. The method proposed for innovation processes control on the base of Markov decision processes has another important advantage - optimal policies and strategies may be recomputed on the base of new and more refined data after each step completed step of the process and the state it falls into. This may result in better final result by improving the strategy initially computed.

IV. It is possible to use more precise classes of Markov decision, e.g. by using profit discount at each step at each step, with constrained capacity or through Markov flows or Markov games (Sgurev, 1993).

4 CONCLUSIONS

In conclusion the following general inferences may be drawn:

1. The innovation processes are highly stochastic and uncertain, which results to highly imprecise prognostication of their completion. And this is connected with a big risk at the venture financing of such processes.

2. The method proposed in the present work for using multistep Markov decision processes for description of the innovation processes provides a possibility their stochastic character to be recognized to a considerable degree and an effective procedure to be proposed for their behavior control.

ACKNOWLEDGEMENTS

The research work reported in the paper is partly supported by the project AComIn “Advanced Computing for Innovation”, grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions) and partially supported by the European Social Fund and Republic of Bulgaria, Operational Programme “Development of Human Resources” 2007-2013, Grant № BG051PO001-3.3.06-0048.

REFERENCES

Grossi G., Promoting Innovations in a Big Business, Long Range Planning, vol. 23, #1, 1990.

Mine H., S. Osaki, Markovian Decision Processes, Amer. Elsevier Publ. Comp., Inc, N.Y., 1975.

Cormican K., D. O'Sullivan, Auditing best practice for effective product innovation management, Technovation, Volume 24, Issue 10, pp. 761-851, Elsevier, 2004.

Bernsteina B., P. J. Singhb, An integrated innovation process model based on practices of Australian biotechnology firms, Volume 26, Issues 5–6, pp. 561–572, Elsevier, 2006.

Sgurev V., Markov Flows, Publishing House of the Bulgarian Academy of Sciences, Sofia, 1993 (in Russian).

Generalized Net Model of the Methodology for Analysis of the Creditworthiness and Evaluation of Credit Risk in SMEs Financing

George L. Shahpazov[1], Lyubka A. Doukovska[1] and Krassimir T. Atanassov[2]
[1]*Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Acad. G. Bonchev str., bl. 2, 1113 Sofia, Bulgaria*
[2]*Institute of Biophysics and Biomedical Engineering, Acad. G. Bonchev str., bl. 105, 1113 Sofia, Bulgaria*
atlhemus@abv.bg, doukovska@iit.bas.bg, krat@bas.bg

Keywords: Generalized Net Model, Small and Medium-Sized Enterprises (SMEs), Credit Risk, Creditworthiness.

Abstract: The launch of the new programming period of the EU in 2014 will lead to many changes in the way the EU budget is funded. The European Commission is considering various ways to generate their own income to make it more independent of the Member States. Unfortunately the consequences of the reforms might have negative influence to European and in particular to the Bulgarian economy and especially for SMEs. The effective results in these conditions are to ensure financial resources for SMEs beneficiaries - increasing the amount of advance payments, creating additional financial instruments. Prior to SMEs financing, a methodology for analysis of creditworthiness and credit risk assessment procedures are applied. The aim of the methodology is to contribute and establish an unified and systematic approach to analyzing and assessing credit risk, which is to lead to a more thorough and objective assessment of the credit and minimize the risk undertaken by the financial institution. The system of credit risk assessment is a collaboration of estimates of the specified indicators. The final conclusion of the process should result into a motivated standpoint, based on which, a decision on further conduct of the Bank towards the loan request will be made (guarantees and other commitments, bearing credit risk), along with periodic risk assessment procedure on already granted loans. In this paper is provided an analysis using the Generalized Nets. They are used as a tool for modelling of different processes in industries and medicine. In the present paper, an application of these nets apparatus for assistive technology and the advantages of using such model, for SMEs financial support mechanism are discussed.

1 INTRODUCTION

Considering the harder economic conditions, to which SME's are exposed, the attitude to external financing changes. The research of the sector show that 10 years ago about 7% of enterprises utilized investment loans, 17% had access to working capital funds, and 67% didn't have any access to financing. The aggressive development of banking system along with EU structured funds, significantly increased the accession of SME's to venture funding. From year 2010 onwards, about 55% of companies are able to reach financing of any type.

In 2010 most popular sources of financing between SME's was own resources (about 42%), illegitimate financing from frends and relatives (close to 17%), and at last EU funds and Bank financing (near 30%). A year earlier above 50% of companies are financed with own equity. Limitations and obstacles in financing occur mainly due to the reduced investment intentions of SME's within the last few years. Main reasons for it are lack of economic stability within the country and EU, along with gradual increase of intercompany leverage. The figures show that, intercompany debp over the past 3 years has gone up over 100%.At present time about 83% of all SME's have uncollected receivables (Bulgarian Industrial Association).

One third of all investments made by SME's are into new equipment and machinery (about 35%), re qualification, training and advertisement is the second investment direction (29%), development of present and design of additional newer products (22%), introduction of systems for intercompany management processes (9%).

Alternative ways of raising funds by SME's are via leasing schemes, where at present about 32% of

SME's are able to reach their investment goals, whereas couple of years earlier the figure is close to 45%.

Due to worsen economic environment and interbanking debt, weaker turnover and profit results, most SME's are unable to rely on own resources. This is valid to such an extent that the financing with own funds has decreased 10 times and in spite of the difficulties, concerning the receipt of a bank loan, it has turned into the most preferred source of funds.

The most popular source of financing among commercial banks and leasing companies is public procurement. Statistics show that about 15% of SME's take advantage of public procurement. Raising funds via government programs was used by 2.9% of the companies, and access to financing via programs of non-government organizations has a share of 2%. Financing via EU structured funds had an insignificant portion (1.6%) up until few years ago. Nowadays the percentage has increased considerably and 45% of SME's is making efforts to receive the embedded financing and grant schemes, (Bulgarian Small and Medium Enterprise Promotion Agency).

Regardless of the above mentioned statistics there hasn't been any considerable changes in regards to the specific difficulties, with which SME are confronted upon the receipt of a bank loan. Most of which they encounter are:

- Considerableinterest rates and requirements for sufficient loan collateral. Often companies do not dispose with the necessary real estates, and the interest rates are close to the profitability of their assets.
- Lacking or insufficient credit history (valid to an even greater extent for the new companies). The reason for this often is the concealing of tax, despite the decrease in the tax and social security burden in the last years.
- The relatively low economic and legal general knowledge of the owners of SMEs.
- Incapacity for the preparation of a long-term plan for the development of business. This is the result of the unstable economic environment, as well as of the incapacity of SMEs to prepare reliable long-term financial forecasts.
- High fees, "hidden" interest and the heavy paperwork, associated with loan granting/project financing.
- Requirements for minimum equity and minimum turnover.

2 SHORT REMARKS ON GENERALIZED NETS

Generalized Nets (GN) (Atanassov, 1991, Atanassov, 2007) are extensions of Petri nets and other modifications of them. They are tools intended for the detailed modelling of parallel processes.

A GN is a collection of *transitions* and *places* ordered according to some rules (see Figure 1). The places are marked by circles. The set of places to the left of the vertical line (the transition) are called *input places*, and those to the right are called *output places*. For each transition, there is an index matrix with elements called *predicates*. Some GN-places contain *tokens* – dynamic elements entering the net with initial characteristics and getting new ones while moving within the net. Tokens proceed from an input to an output place of the transition if the predicate corresponding to this pair of places in the index matrix is evaluated as *"true"*. Every token has its own identifier and collects its own history that could influence the development of the whole process modelled by the GNs.

Two time-moments are specified for the GNs: for the beginning and the end of functioning, respectively.

A GN can have only a part of its components. In this case, it is called *reduced GN*. Here, we shall give the formal definition of a reduced GN without temporal components, place and arc capacities, and token, place and transition priorities.

Formally, every transition in the used below reduced GN is described by a three-tuple:

$$Z = \langle L', L'', r\rangle \quad (1)$$

where:

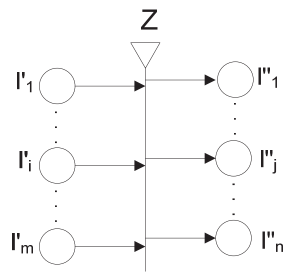


Figure 1: A GN transition.

(a) L' and L'' are finite, non-empty sets of places (the transition's input and output places, respectively), for the transition these are:

$L' = \{l'_1, l'_2, ..., l'_m\}$ and $L'' = \{l''_1, l''_2, ..., l''_n\}$;

(b) r is the transition's *condition* determining which tokens will pass (or *transfer)* from the transition's inputs to its outputs; it has the form of an Index Matrix (IM):

$$r = \begin{array}{c|ccccc} & l''_1 & \dots & l''_j & \dots & l''_n \\ \hline l'_1 & & & & & \\ \dots & & & r_{i,j} & & \\ l'_i & & (r_{i,j} & - & predicate) & \\ \dots & & (1 \le i \le m, & & 1 \le j \le n) & \\ l'_m & & & & & \end{array}$$

where $r_{i,j}$ is the predicate that corresponds to the i-th input and j-th output place.

When its truth value is *"true"*, a token from the i-th input place transfers to the j-th output place; otherwise, this is not possible.

The ordered four-tuple:

$$E = \langle A, K, X, \Phi \rangle \quad (2)$$

is called a *reduced Generalized Net* if:

(a) A is the set of transitions;
(b) K is the set of the GN's tokens;
(c) X is the set of all initial characteristics which the tokens can obtain on entering the net;
(d) Φ is the characteristic function that assigns new characteristics to every token when it makes the transfer from an input to an output place of a given transition.

Many operations (e.g., union, intersection and others), relations (e.g., inclusion, coincidence and others) and operators are defined over the GNs. Operators change the GN-forms, the strategies of token transfer and other. There are six types: global, local, hierarchical, reducing, extending and dynamic operators.

3 GENERALIZED NET MODEL

A GN model is described in In this paper will be used GN shown on Figure 2. Five types of tokens move in this GN.

The tokens from the first type are α_1 and α_2, and they represent bank-administrators. The tokens have the initial and current characteristics: *"Credit specialist at branch level"* in place l_8 and *"Experts at Headquarters level"* in place l_{15}.

The tokens from the second type are the φ-tokens that permanently enter place l_1 with initial characteristic *"Potential SME Borrower"*.

The tokens from the third type are χ_1, χ_2 and χ_3, representing Bank management. They have the initial and current characteristics: *"Credit Council"* in place l_{18}, *"Management Board"* in place l_{21} and *"Supervisory Board"* in place l_{24}.

In some time-moments, some token φ will split to the original token φ and a token π, while some α-token and the χ_3-token will split to the original α- or χ_3-token and a β-token. These new types of tokens will be discussed below.

$$Z_1 = \langle \{l_1, l_4, l_{13}\}, \{l_2, l_3, l_4\}, r_1 \rangle \quad (3)$$

$$r_1 = \begin{array}{c|ccc} & l_2 & l_3 & l_4 \\ \hline l_1 & false & false & true \\ l_4 & W_{4,2} & W_{4,3} & true \\ l_{13} & false & false & true \end{array},$$

where:

$W_{4,2}$ = *"There is a SME client that has prepared a project"*,

$W_{4,3}$ = *"There is an answer from the SME client to a question from the credit specialist at branch level"*.

Token φ enters place l_4 without a new characteristic.

Token β_4 enters place l_4 and unites with token φ, staying there.

If $W_{4,2}$ = *true*, then token φ splits to the original token φ and token π. The second one enters place l_2 and there it obtains the characteristic *"Loan application, based upon a prepared project"*. If $W_{4,3}$ = *true*, then token φ splits to the original token φ and token β_1. The second one enters place l_3 and there it obtains the characteristic *"Requested additional information in regards to submitted project"*. This token is generated in a result of token β_4 that enters place l_4.

$$Z_2 = \langle \{l_2, l_5, l_8, l_{14}\}, \{l_5, l_6, l_7, l_8\}, r_2 \rangle \quad (4)$$

$$r_2 = \begin{array}{c|cccc} & l_5 & l_6 & l_7 & l_8 \\ \hline l_2 & true & false & false & false \\ l_5 & W_{5,5} & W_{5,6} & W_{5,7} & false \\ l_8 & false & false & W_{8,7} & true \\ l_{14} & false & false & false & true \end{array},$$

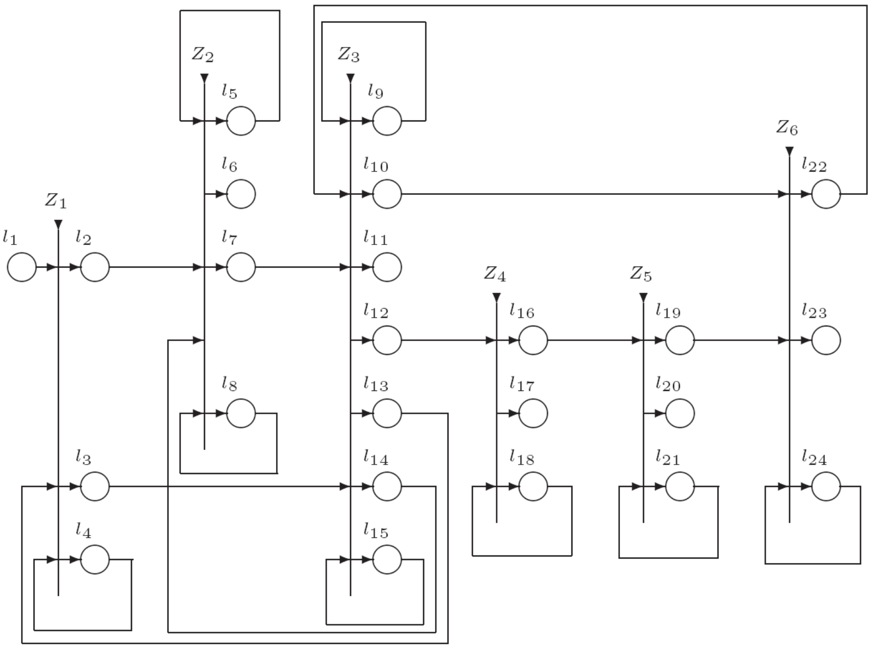


Figure 2: Generalized net model.

where:
$W_{5,5}$ = *„By the moment, there is not a solution for the project*",
$W_{5,6}$ = *"Project rejected at first level (at branch level)"*,
$W_{5,7}$ = *"Project accepted at branch level, sent to Headquarters for further detailed research"*,
$W_{8,7}$ = *"There is an answer of a question initiated by Headquarters experts in regards to the submitted project"*.

Token π enters place l_5 without any new characteristic. Token β_5 enters place l_8 and unites with token α_1.

When $W_{5,5}$ = *true*, token π continues to stay in place l_5 without a new characteristic. When $W_{5,6}$ = *true*, token π enters place l_6 with a characteristic *"Project rejected (due to specific motives)"*. When $W_{5,7}$ = *true*, token π enters place l_7 with a characteristic *"Project accepted (due to specific motives)"*. If $W_{8,7}$ = *true*, then token α_1 splits to the original token α_1 and token β_2. The second one enters place l_7 and there, it obtains the characteristic *"Answer from branch level"*.

$$Z_3 = \langle\{l_3, l_7, l_9, l_{15} l_{22}\}, \{l_9, l_{10}, l_{11}, l_{12}, l_{13}, l_{14}, l_{15}\}, r_3\rangle \quad (5)$$

r_3 =

	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{14}	l_{15}
l_3	*false*	*false*	*false*	*false*	*false*	*false*	*true*
l_7	$W_{7,9}$	*false*	*false*	*false*	*false*	*false*	$W_{7,15}$
l_9	$W_{9,9}$	*false*	$W_{9,11}$	$W_{9,12}$	*false*	*false*	*false*
l_{15}	*false*	$W_{15,10}$	*false*	*false*	$W_{15,13}$	$W_{15,14}$	*true*
l_{22}	*false*	*false*	*false*	*false*	*false*	*false*	*true*

,

where:
$W_{7,9}$ = *"The current token is from π-type"*,
$W_{7,15}$ = *"The current token is from β_5-type"*,
$W_{9,9}$ = *„By the moment, there is not a solution for the project*",
$W_{9,11}$ = *"Rejected at Headquarters level"*,
$W_{9,12}$ = *"Accepted and prepared for loan granting"*,
$W_{15,10}$ = *"An inquiry is initiated and addressed to the Supervisory Board"*,
$W_{15,13}$ = *"An inquiry is initiated and addressed to the SME Client-borrower"*,
$W_{15,14}$ = *"An inquiry is initiated and addressed to branch level"*.

Token β_1 enters place l_{15} and unites with token α_2.

When $W_{7,9}$ = *true*, token π enters place l_9 without a new characteristic.

When $W_{9,15}$ = *true*, token β_2 enters place l_{15} and unites with token α_2, that obtains the above mentioned current characteristic.

When $W_{9,9}$ = *true*, token π contains to stay in place l_9 without a new characteristic.

When $W_{9,11} = true$, token π enters place l_{11} with a characteristic *"Project rejected at Headquarters level (due to specific motives)".*

When $W_{9,12} = true$, token π enters place l_{12} with a characteristic *"Project accepted at Headquarters level (due to specific motives)".*

When $W_{15,10} = true$, token α_2 splits to the original token α_2 and token β_3. The second one enters place l_{10} and there, it obtains the characteristic *"An inquiry is addressed to the Supervisory Board for specific project" or "An answer of Head quarters level to the Supervisory Board"*

When $W_{15,13} = true$, token α_2 splits to the original token α_2 and token β_4. The second one enters place l_{13} and there, it obtains the characteristic *"An inquiry is addressed to the SME Client-borrower in regards to a specific detail of the project".*

When $W_{15,14} = true$, token α_2 splits to the original token α_2 and token β_5. The second one enters place l_{14} and there, it obtains the characteristic *"An inquiry is addressed to Branch level in regards to specific details of the project".*

$$Z_4 = \langle \{l_{12}, l_{18}\}, \{l_{16}, l_{17}, l_{18}\}, r_4 \rangle \quad (6)$$

$$r_4 = \begin{array}{c|ccc} & l_{16} & l_{17} & l_{18} \\ \hline l_{12} & W_{12,16} & W_{12,17} & true \\ l_{18} & false & false & W_{18,18} \end{array},$$

where:

$W_{12,16}$ = *"There is a positive decision by Credit council in regards to specific project",*

$W_{12,17}$ = *"There is a negative decision by Credit council in regards to specific project"*,

$W_{18,18}$ = *"There is a token in place l_{12}".*

When $W_{12,16} = true$, token π enters place l_{16} with a characteristic "*The project is voted and accepted for financing by the Credit council under the original or new updated parameters*".

When $W_{12,17} = true$, token π enters place l_{17} without any characteristic.

$$Z_5 = \langle \{l_{16}, l_{21}\}, \{l_{19}, l_{20}, l_{21}\}, r_5 \rangle \quad (7)$$

$$r_5 = \begin{array}{c|ccc} & l_{19} & l_{20} & l_{21} \\ \hline l_{16} & W_{16,19} & W_{16,20} & true \\ l_{21} & false & false & W_{21,21} \end{array},$$

where:

$W_{16,19}$ = *"The Project receives affirmative decision when voted by Management Board",*

$W_{16,20}$ = *"The Project receives negative decision when voted by Management Board",*

$W_{21,21}$ = *"There is a token in place l_{16}".*

When $W_{16,19} = true$, token π enters place l_{19} with a characteristic *"The project is voted and accepted for financing by the Management Board under the original or new updated parameters".*

When $W_{16,20} = true$, token π enters place l_{20} without any characteristic.

$$Z_6 = \langle \{l_{10}, l_{19}, l_{24}\}, \{l_{22}, l_{23}, l_{24}\}, r_6 \rangle \quad (8)$$

$$r_6 = \begin{array}{c|ccc} & l_{22} & l_{23} & l_{24} \\ \hline l_{10} & false & false & true \\ l_{19} & false & W_{19,23} & false \\ l_{24} & W_{24,22} & false & W_{24,24} \end{array},$$

where:

$W_{19,23}$ = *"Final decision by Supervisory Board",*

$W_{24,22}$ = *"There is an answer of the Supervisory Board to the Management Board level or there is an answer of the Supervisory Board to a question from the Credit council "*,

$W_{24,24}$ = *"There is a token in place l_{19}".*

Token β_3 enters place l_{24} and unites with token χ_3.

When $W_{19,23} = true$, token π enters place l_{23} with a characteristic "*Final decision (positive or negative) of the Supervisory Board about the project".*

When $W_{24,22} = true$, token χ_3 splits to two tokens – the original token χ_3 and token β_6 that obtains the characteristic *"Answer of the Supervisory Board".*

4 CONCLUSIONS

The so constructed GN model describes the most important steps of the process of evaluation of a business project proposal intended for financing**.** In a next research, the authors plan to elaborate the model in the aspect related to the process of decision making within the frames of the bank administration.

First, the model can be used for real-time control of the processes, flowing in a particular bank. If this is the case, the databases of the model will correspond to the real databases of that bank, and the process of adding new characteristics of the respective GN-tokens will correspond to the process of inputting new information in the bank's databases. The tokens, representing the bank's clients, will have as initial characteristics their specific parameters and with their real project proposals intended for financing**.** The movement of these real projects will be observed and information for the current status of each of them can be obtained from the model. Practically, the GN-model will synchronize the real processes, related to the above described procedure.

Second, it can be a tool for prognostics of different situations, related to the modeled processes, for example in a given moment of time, a large number of projects may be submitted, and these have to be evaluated in parallel or compete for a limited amount of funding.

Third, on the basis of the model, some changes of the process of evaluation can be simulated and the results can be used for searching the optimal scheduling of the separate steps of this process.

ACKNOWLEDGEMENTS

The research work reported in the paper is partly supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions) and partially supported by the European Social Fund and Republic of Bulgaria, Operational Programme "Development of Human Resources" 2007-2013, Grant № BG051PO001-3.3.06-0048.

REFERENCES

Bulgarian Small and Medium Enterprise Promotion Agency – www.sme.government.bg

Shahpazov G., L. Doukovska - Structuring of Growth Funds with the Purpose of SME's Evolution under the JEREMIE Initiative, Proc. of the Second International Symposium on Business Modeling and Software Design - BMSD'12, 4-6 July 2012, Geneva, Switzerland, ISBN 978-989-8565-26-6, pp. 159-164, 2012.

European Commission, "SEC (2011) 876 final/2", Brussels, 27.10.2011.

The Global Competitiveness Report 2011, www.ced.bg.

Atanassov K. - Generalized Nets. World Scientific, Singapore, New Jersey, London, 1991.

Atanassov K. - On Generalized Nets Theory, "Prof. M. Drinov" Academic Publishing House, Sofia, Bulgaria, 2007.

Forecasting Price Movement of SOFIX Index on the Bulgarian Stock Exchange – Sofia using an Artificial Neural Network Model

Veselin L. Shahpazov, Vladimir B. Velev and Lyubka A. Doukovska
Institute of Information and Communication Technologies, Bulgarian Academy of Sciences,
Acad. G. Bonchev str., bl. 2, 1113 Sofia, Bulgaria
veselin_georgiev@abv.bg, vladimir.velev@allianz.bg, doukovska@iit.bas.bg

Keywords: Forecasting, SOFIX Index, Artificial Neural Network, Business, Predicting Stock Prices, Supervised Learning.

Abstract: The Bulgarian capital market is characterized by its relatively short history and its low liquidity, SOFIX is the first index of BSE-Sofia based on the market capitalization of the included issues of common shares, adjusted with the free-float of each of them. The authors intend to use a model implying an Artificial Neural Network to predict the future price of the index. A neural network has the ability to extract useful information from large sets of data, which often is required for a satisfying description of a financial time series. Capital markets are known for their complexity and unpredictability and are best described as chaotic systems. Artificial Neural Networks can be used to find relationship in large sets of data which have some unknown relationship between input and output. Once that relationship is found, the neural network can be used to compute the output for similar (but usually different) input.

1 INTRODUCTION

The Bulgarian capital market is characterized by its relatively short history and its low liquidity, especially in recent years. Yet the companies listed represent different economic segments from the heavy production industry to pharmaceuticals and local banking. SOFIX is the best known and the first index of BSE-Sofia, which calculation started on October 20, 2000. SOFIX is based on the market capitalization of the included issues of common shares, adjusted with the free-float of each of them. SOFIX is the most successful index calculated by BSE-Sofia and is the first one on which structured products are based on. The index covers the 15 issues of shares complying with the general requirements for selection of constituent issues that have the greatest market value of the free-float.

Artificial Neural Networks are flexible computing frameworks and universal approximators that can be applied to a wide range of time series forecasting problems with a high degree of accuracy, (Atsalakis, 2009). They are an artificial intelligence method for modeling complex target functions. For certain types of problems, such as learning to interpret complex realworld sensor data, Artificial Neural Networks are among the most effective learning methods currently know. During the last decade they have been widely applied to the domain of financial time series prediction and their importance in this field is

growing. The ability of neural networks to closely approximate unknown functions to any degree of desired accuracy has generated considerable demand for Neural Network research in Business. The attractiveness of neural network research stems from researchers need to approximate models within the business environment without having a priori knowledge about the true underlying function, (Sexton, 1998). However, despite all advantages cited for artificial neural networks, their performance for some real time series is not satisfactory.

Predicting stock prices with traditional time series analysis has proven to be difficult. An artificial neural network may be more suitable for the task. Primarily because no assumption about a suitable mathematical model has to be made prior to forecasting. Furthermore, a neural network has the ability to extract useful information from large sets of data, which often is required for a satisfying description of a financial time series, (Nygren, 2004). In recent years, neural networks have received an increasing amount of attention as a very popular forecasting and data mining tool. Their origin stems from the attempt to model the human thought process as an algorithm which can be efficiently run on a computer. Software is developed to mimic this thought process. A neural network can be used to find relationship in large sets of data which have some unknown relationship between input and output. Once that relationship is found, the neural network can be used to compute the output for similar (but usually different) input, (Choong, 2009). One of the advantages include automatic learning of dependencies only from measured data without any need to add further information (such as type of dependency like with the regression). The neural network is trained from the historical data with the hope that it will discover hidden dependencies and that it will be able to use them for predicting into future.

Many authors propose the use of the Artificial Neural Networks (ANN), a machine learning method, to improve trading performance, (Lawrence, 1997, Tilakaratne, 2008, Alhaj, 2011, Khan, 2010, Montana, 1989, Skabar 2002, Tan, 2006, Maciel, 2008).

2 PROBLEM FORMULATION

The neural network that the authors intend to use for predicting the price of Sofix index will be trained with supervised learning. The network that will be used will be a feed forward, multi-layer perceptron network with very fast learning and an advanced mechanism to prevent overfitting. The Multi-layer perceptron (MLP) networks trained using backpropagation (BP) algorithm are the most popular choice in neural network applications in finance, (Atsalakis, 2009 and Atanasova, 2006). The MLP networks are feed forward neural networks with one or more hidden layers which is capable to approximate any continuous function up to certain accuracy just with one hidden layer (Cybenko, 1989).

The MLP consists of three types of layers. The first layer is the input layer and corresponds to the problem input variables with one node for each input variable. The second layer is the hidden layer used to capture non-linear relationships among variables. The third layer is the output layer used to provide predicted values.

The data used for the study is from the Bulgarian stock exchange official website, and consists of the following: last price, open, high, low and volume traded, as well as five of the most commonly used technical indicators (according to study conducted by bigtrends.com): the 30 day moving average, 60 day moving average, 200 day moving average, the 14 day relative strength index and the 30 day relative strength index. The training data covers a period of two years and two months or from 04.01.2011 until 08.03.2013. We find the period appropriate because it consists of an uptrend and a downtrend in the first couple of months than the price of SOFIX consolidates and in the last months of the observed period enters into an uptrend. This we believe will permit us to train the network in a better way and produce results with a minimal error.

The training data is split into three parts, with the major part of 50% of the data is treated as actual training data, and the rest are treated as a testing data (25%) and validation data (25%).

The software used during this study is STATISTICA 7.0. The preferred Neural Network structure was a three layer perceptron.

The software operates using back-propagation and conjugate gradient descent algorithms for training the network. The Artificial Neural Networks implements the on-line version of back propagation; i.e. it calculates the local gradient of each weight with respect to each case during training. Weights are updated once per training case.

The update formula is:

$$\Delta\omega_{ij}(t) = \eta\delta_j o_i + \alpha\Delta\omega_{ij}(t-1) \quad (1)$$

where:

η - the learning rate;

δ- the local error gradient;

α- the momentum coefficient;

o_i - the output of the i'th unit.

Thresholds are treated as weights with o_i = -1. The local error gradient calculation depends on whether the unit into which the weights feed is in the output layer or the hidden layers. Local gradients in output layers are the product of the derivatives of the network's error function and the units' activation functions. Local gradients in hidden layers are the weighted sum of the unit's outgoing weights and the local gradients of the units to which these weights connect.

The Conjugate gradient descent (Bishop, 1995; Shepherd, 1997) is an advanced method of training multilayer perceptron's. It usually performs significantly better than back propagation, and can be used wherever back propagation can be. It is the recommended technique for any network with a large number of weights (more than a few hundred) and/or multiple output units. Conjugate gradient descent is a batch update algorithm: whereas back propagation adjusts the network weights after each case, conjugate gradient descent works out the average gradient of the error surface across all cases before updating the weights once at the end of the epoch.

The most widely used activation function for the output layer are the sigmoid and hyperbolic functions. In this paper, the sigmoid transfer function is employed and is given by:

$$E(t) = \frac{1}{1+e^{-t}} \quad (2)$$

The criteria which will evaluate the Neural Networks performance will be the error of the network on the subsets used during training (Root Mean Square-RMS).

$$RMS = \sqrt{\frac{\sum_{i=1}^{n}(\hat{\delta}_i - \delta_i)^2}{n}} \quad (3)$$

This is less interpretable than the performance measure, but is the figure actually optimized by the training algorithm (at least, for the training subset).

This is the RMS of the network errors on the individual cases, where the individual errors are generated by the network error function, which is either a function of the observed and expected output neuron activation levels (usually sum-squared or a cross-entropy measure); or Sum-squared. The error is the sum of the squared differences between the target and actual output values on each output unit. This is the standard error function used in regression problems.

The weights and biases of the network are automatically initialized to small random numbers by the software.

3 STUDY RESULTS

In order to achieve better results with training the Neural Network the authors decided to transform the input data from values to change. Since the period of time that the input data represents is limited to nearly two years the index fluctuates between a minimum value of 286.03 and a maximum value of 455.75.

The initial results showed that the networks that utilized all eight input parameters (last price, open, high, low and volume traded 30 day moving average, 60 day moving average, 200 day moving average, the 14 day relative strength index and the 30 day relative strength index) performed consistently much worse than the ones that isolated some of the input parameters. The networks that based their prognoses solely on the last price showed the best results in terms of test error.

The tests were conducted with different network architecture but the best result was obtained with a three layer perceptron, consisting of 1 input (43 time lagged steps) 7 nodes in the hidden layer and 1 output, training consisted of 100 epochs using back-propagation and 23 epochs using the conjugate gradient descent algorithm, the test error amounted to 0.119279. It is important to outline that the software stops the learning process when the minimum error is reached, that way it prevents the network from over-fitting.

After taking into consideration the specifics of the Bulgarian stock market (the extremely low traded volume, a problem that exposes the index to manipulation which will lead to distortion of the network results) the authors have found appropriate to conduct moving average smoothing to the input data commencing with 3 days smoothing (calculating the average value of the last three days).

As seen from the results shown in Table 1 this data manipulation managed to bring down the test error considerably. The table shows the effects of increasing the period that has been smoothed, the

Table 1: Results, produced with 3 to 9 days smoothing of last price data and respective network structures learning samples and test error (RMS).

	Network structure Input-hidden-output	Learning samples BP/CGD	Test error RMS
3 days smoothing	1(12)-5-1	BP-100; CGD-115	0.076298
4 days smoothing	1(12)-11-1	BP-100; CGD-58	0.089827
5 days smoothing	1(25)-15-1	BP-100; CGD-48	0.067025
6 days smoothing	1(10)-5-1	BP-100; CGD-76	0.075116
7 days smoothing	1(20)-13-1	BP-100; CGD-34	0.057099
8 days smoothing	1(15)-14-1	BP-100;CGD-68	0.057903
9 days smoothing	1(15)-5-1	BP-100; CGD-67	0.064887

different network structures, learning samples and most importantly the resulted test error. The error reaches its minimum with 7 day smoothing and a structure of the network with 1 input (20 time lagged steps) 13 nodes in the hidden layer and 1 output, training consisted of 100 epochs using back-propagation and 34 epochs using the conjugate gradient descent algorithm, the test error amounted to 0. 057903.

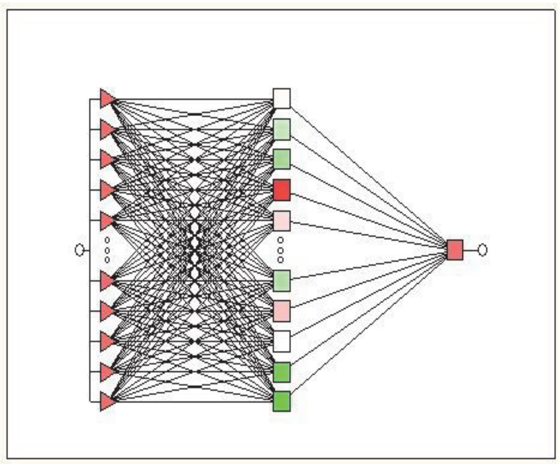


Figure 1: Structure of Neural Network with 1 input (20 time lagged steps) 13 nodes in the hidden layer and 1 output.

Figures 1, 2, 3 and 4 show structure, predicted values versus observed plot, graphic of the predictions over the analysed period and the residual error diagram respectively.

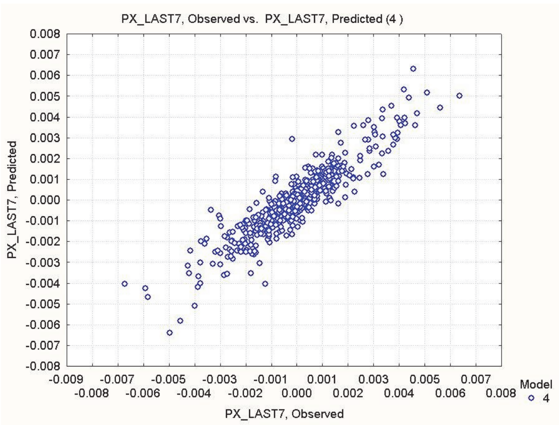


Figure 2: Predicted values versus observed values plot.

As we can see from Figure 4, the error tends to increase in in the beginning and the end of the analysed period of time. This can be explained from an economic perspective with the fact that during both periods the Bulgarian stock market is in a trend situation usually characterized with higher volatility in the prices of stocks, especially in downtrends. This leads the authors to believe that predictive abilities of a neural network are better suited for markets that find themselves in a period of consolidation.

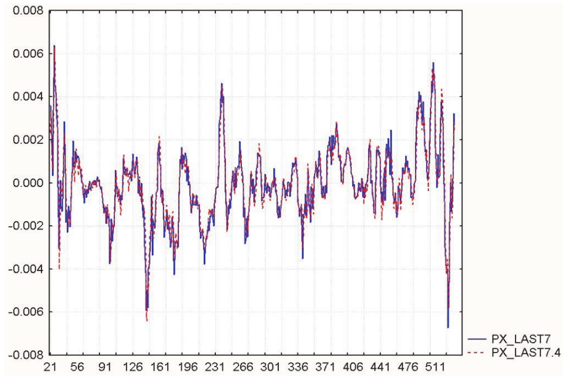


Figure 3: Predicted change (dotted line) versus observed change.

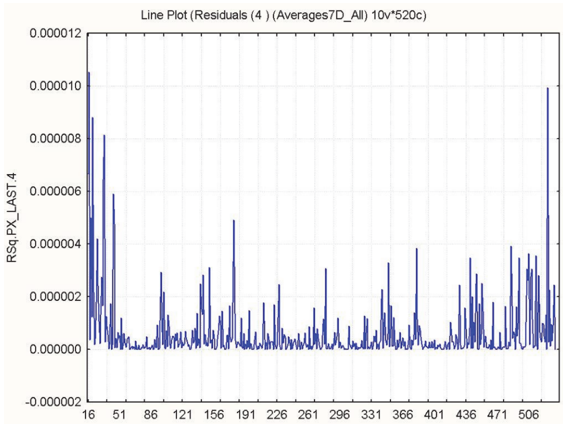


Figure 4: The residual error.

The Networks that were fed with the same pre-processed data but trained with all 10 input parameters showed worse results. In the case with the best results (7 days smoothing) in terms of test error, the value of RMS 0.068935, the network structure produced by the software was 9 input (15 time lagged steps) 21 nodes in the hidden layer and 1 output, training consisted of 100 epochs using back-propagation and 81 epochs using the conjugate gradient descent algorithm.

4 CONCLUSIONS

In this paper, the problem of predicting the price of Bulgarian Stock Exchange's Sofix index using neural networks is considered. The analyzed period is of two years and two months or from 04.01.2011 until 08.03.2013. Data used for the case consists of the daily values of last price, open, high, low and volume traded 30 day moving average, 60 day moving average, 200 day moving average, the 14 day relative strength index and the 30 day relative strength index.

The criteria which was used to evaluate the Neural Networks performance was the error of the network on the subsets used during training (Root Mean Square)

The input data was preprocessed and transformed from values into daily changes. Initial readings showed that better results would be achieved if the input is one compared to using all or fragments of the initial data set.

Smoothing ranging from 3 to 9 days was performed in order to eliminate the effects of the low liquidity and higher volatility in the market.

Results showed that this data manipulation managed to bring down the test error considerably. The produced neural network was structured by 1 input (20 time lagged steps) 13 nodes in the hidden layer and 1 output, training consisted of 100 epochs using back-propagation and 34 epochs using the conjugate gradient descent algorithm, the test error amounted to 0.057903. In comparison the best performing network using all or partial input data managed an error of 0.068935.

The obtained result was found good but the authors see further room for improvement of the predicting capabilities of the model. The error margin is still considered big and attempts to bring it further down will be made, especially improving the predictive capabilities for trends. The low liquidity and high volatility environment of the Bulgarian stock market is a challenge that could be addressed more efficiently with similar neural networks that have different structure and learning algorithms.

Future work will involve different input data and data pre-processing, possibly other types of neural networks and algorithms.

ACKNOWLEDGEMENTS

The research work reported in the paper is partly supported by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions) and partially supported by the European Social Fund and Republic of Bulgaria, Operational Programme "Development of Human Resources" 2007-2013, Grant № BG051PO001-3.3.06-0048.

REFERENCES

Lawrence R., Using Neural Networks to Forecast Stock Market Prices, Department of Computer Science University of Manitoba, December 12, 1997.

Tilakaratne C. D., S. A. Morris, M. A. Mammadov, C. P. Hurst, Predicting Stock Market Index Trading Signals Using Neural Networks, Centre for Informatics and Applied Optimization School of Information Technology and Mathematical Sciences University of Ballarat, Ballarat, Victoria, Australia. 2008.

Alhaj Ali S. M., A. A. Abu Hammadb, M. S. Samhouria, A. Al-Ghandoora, Modelling Stock Market Exchange Prices Using Artificial Neural Network: A Study of Amman Stock Exchange, Industrial Engineering, Department, Faculty of Engineering, Hashemite University, Jordan. JJMIE Volume 5, Number 5, 2011.

Nygren K., Stock Prediction – A Neural Network Approach, Master Thesis Royal Institute of Technology KTH, Supervisor: Prof. Kenneth Holmstrom, 2004.

Khan Z. H., T. S. Alin, M. A. Hussain, Price prediction of share market using Artificial Neural Network, Department of CSE, SUST Sylhet, Bangladesh. International Conference on Computer Information Systems and Industrial Management Applications (CISIM), 2010.

Montana D. J., L. Davis, Training Feed-forward Neural Networks Using Genetic Algorithms, BBN Systems and Technologies Corp., IJCAI'89 Proceedings of the 11th international joint conference on Artificial intelligence - Volume 1 Pages 762-767, 1989.

Skabar A., I. Cloete, Neural Networks, Financial Trading and the Efficient Markets Hypothesis, School of Information Technology International University in Germany, Twenty-Fifth Australasian Computer Science Conference (ACSC2002), 2002.

Tan Clarence N. W., An Artificial Neural Networks Primer with Financial Applications Examples in Financial Distress Predictions and Foreign Exchange Hybrid Trading System, PhD Thesis's School of Information Technology, Bond University, Gold Coast, Australia, 2006.

Maciel L. S., R. Balliani, Design a Neural Network for time series financial forecasting: accuracy and robustness analysis, Instituto de Economia (IE), Universidade Estadual de Campinas (UNICAMP) São Paulo – Brasil, 2008.

Sexton R. S., B. Alidaee, R. E. Dorsey, J. D. Johnson, Global Optimization for Artificial Neural Networks: A Tabu Search Application, European Journal of

Operational Research, Volume 106, Issues 2–3, Pages 570–584 April 1998.
Atsalakis G. S., Valavanis K. P., Surveying Stock Market Forecasting Techniques – Part II: Soft Computing Methods, Expert Systems with Applications, Journal Expert Systems with Applications: An International Journal Volume 36 Issue 3, Pages 5932-5941 April, 2009.
Cybenko G., Approximations by Superpositions of a Sigmoidal Function, Mathematics of Control, Signals, and Systems, Pages 303-314, 1989.
Atanasova T. Simeonova V., Prediction of stock market index using neural networks and genetic algorithms, University of Economics – Varna, Bulgaria, 2006.
Choong J., Build Neural Network with MS Excel, Published by XLPert Enterprise by XLPert Enterprise, 2009.

AUTHOR INDEX

Proceedings of BMSD 2013
Third International Symposium on Business Modeling and Software Design
ISBN: 978-989-8565-56-3
http://www.is-bmsd.org