LNBIP 319

Boris Shishkov (Ed.)

Business Modeling and Software Design

8th International Symposium, BMSD 2018
Vienna, Austria, July 2–4, 2018
Proceedings

Springer

Lecture Notes in Business Information Processing 319

More information about this series at http://www.springer.com/series/7911

Boris Shishkov (Ed.)

Business Modeling and Software Design

8th International Symposium, BMSD 2018 Vienna, Austria, July 2–4, 2018 Proceedings

Springer

Editor
Boris Shishkov
Bulgarian Academy of Sciences,
Institute of Mathematics and Informatics (IMI)/
Interdisciplinary Institute for Collaboration
and Research on Enterprise Systems
and Technology (IICREST)
Sofia
Bulgaria

ISSN 1865-1348 ISSN 1865-1356 (electronic)
Lecture Notes in Business Information Processing
ISBN 978-3-319-94213-1 ISBN 978-3-319-94214-8 (eBook)
https://doi.org/10.1007/978-3-319-94214-8

Library of Congress Control Number: 2018947352

Printed on acid-free paper

This Springer imprint is published by the registered company Springer International Publishing AG
part of Springer Nature
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

How did enterprises look in the 1970s (when essential technology-driven business transformations started)? What were the rudimentary business process automations at that time and how is this different from the current business process automations that go beyond conventional data manipulation and record-keeping activities? How did enterprises exchange information then, not counting on the global telecommunications and the digital multimedia and what are the differences now when a cellphone alone seems to be capable of supporting video communication, answering complex questions, and providing satellite navigation? Were associations between different enterprises possible (without Web services and cloud infrastructures) to combine manufacturing, assembly, wholesale distribution, and retail sales in what is currently called business process externalization? Were software technologists then able to develop truly adaptable information systems, not counting on sensor technology? We argue that answering these questions would bring us to the conclusion that over the past several decades enterprises have been shifting to experience a growing dependency on ICT – information and communication technology. For this reason, it is not surprising that software engineering is becoming increasingly relevant with regard to enterprise developments. Hence, even though enterprise engineering and software engineering have developed separately as disciplines, it is currently important to bring together enterprise modeling and software specification; we argue that this would allow enterprises to adequately utilize current technology. This is especially valid for the latest technological "booms": (a) blockchain technology; (b) Internet of Things. The former (a) is about rethinking the way inter-organizational business processes are managed, assuming a high degree of security: In fact, once a transaction is certified and saved within one of the chain blocks, it can no longer be modified or tampered with (each block consists of a pointer that connects it to the previous block, a timestamp that certifies the time at which the event actually took place, and the transaction data); hence there is no central party serving as a single point of trust and failure. Therefore, the actual blockchain technology developments concern both enterprise (business-processes-related) issues and technical (software-engineering-related) issues. The latter (b) is about bringing together human actions, technical applications, networked devices, sensors, actuators, and so on, for the sake of providing a real-time (sensors-driven) "tuning" of what people and devices are doing. Therefore, further Internet of Things developments concern (among other things) the demand for better aligning the pieces of software running on numerous devices to the overall business processes that are related not only to those devices but also to humans, institutions, regulations, and so on. Thus, bringing together business (enterprise) modeling and technical (software) design is crucial.

BMSD (http://www.is-bmsd.org) is an annual international symposium that brings together researchers and practitioners who are inspired to consider that challenge.

Since 2011, we have enjoyed seven successful BMSD editions. The first BMSD edition (2011) took place in Sofia, Bulgaria, and the theme was: "Business Models and Advanced Software Systems." The second BMSD edition (2012) took place in Geneva, Switzerland, with the theme: "From Business Modeling to Service-Oriented Solutions." The third BMSD edition (2013) took place in Noordwijkerhout, The Netherlands, and the theme was: "Enterprise Engineering and Software Generation." The fourth BMSD edition (2014) took place in Luxembourg, Grand Duchy of Luxembourg, and the theme was: "Generic Business Modeling Patterns and Software Re-Use." The fifth BMSD edition (2015) took place in Milan, Italy, with the theme: "Towards Adaptable Information Systems." The sixth BMSD edition (2016) took place in Rhodes, Greece, and had as theme: "Integrating Data Analytics in Enterprise Modeling and Software Development." The seventh BMSD edition (2017) took place in Barcelona, Spain, and the theme was: "Modeling Viewpoints and Overall Consistency." The 2018 edition in Vienna marks the eighth event, with the theme: "Enterprise Engineering and Software Engineering - Processes and Systems for the Future."

We are proud to have attracted distinguished guests as keynote lecturers, who are renowned experts in their fields: Norbert Gronau, University of Potsdam, Germany (2017), Oscar Pastor, Polytechnic University of Valencia, Spain (2017), Alexander Verbraeck, Delft University of Technology, The Netherlands (2017), Paris Avgeriou, University of Groningen, The Netherlands (2016), Jan Juerjens, University of Koblenz-Landau, Germany (2016), Mathias Kirchmer, BPM-D, USA (2016), Marijn Janssen, Delft University of Technology, The Netherlands (2015), Barbara Pernici, Politecnico di Milano, Italy (2015), Hend*erik* Proper, Public Research Centre Henri Tudor, Luxembourg (2014), Roel Wieringa, University of Twente, The Netherlands (2014), Kecheng Liu, University of Reading, UK (2013), Marco Aiello, University of Groningen, The Netherlands (2013), Leszek Maciaszek, Wroclaw University of Economics, Poland (2013), Jan L.G. Dietz, Delft University of Technology, The Netherlands (2012), Ivan Ivanov, SUNY Empire State College, USA (2012), Dimitri Konstantas, University of Geneva, Switzerland (2012), Marten van Sinderen, University of Twente, The Netherlands (2012), Mehmet Aksit, University of Twente, The Netherlands (2011), Dimitar Christozov, American University in Bulgaria - Blagoevgrad, Bulgaria (2011), Bart Nieuwenhuis, University of Twente, The Netherlands (2011), and Hermann Maurer, Graz University of Technology, Austria (2011).

The high quality of the BMSD 2018 program is enhanced by two keynote lectures delivered by outstanding guests: Jan Mendling, WU Vienna, Austria (Title: "Blockchains for Business Process Management - Challenges and Opportunities") and Roy Oberhauser, Aalen University, Germany (Title: "The Convergence of Business, Software Development and IT Operations and the Next Wave"). Further, the keynote speakers and some other BMSD 2018 participants will take part in a panel discussion and also in other discussions stimulating community building and facilitating possible R&D project acquisition initiatives. These special activities will contribute to maintaining the event's high quality and inspiring our steady and motivated community.

We demonstrated for an eighth consecutive year a high quality of papers and we are proud to have succeeded in establishing and maintaining (for many years already) high scientific quality and stimulating collaborative atmosphere. Also, our community is inspired to share ideas and experiences.

In 2018, the scientific areas of interest to the symposium are: (a) Business Processes and Enterprise Engineering; (b) Business Models and Requirements; (c) Business Models and Services; (d) Business Models and Software; (e) Information Systems Architectures and Paradigms; (f) Data Aspects in Business Modeling and Software Development.

In tune with the aforementioned challenges and in line with the BMSD areas, BMSD 2018 addresses a large number of research topics:

- Design Thinking
- Business Processes
  - Enterprise Modeling
  - Business Process Modeling
  - Business Process Variability
  - Business Process Management and Notations
  - Evaluation of Notations
  - Business Process Contracting
  - Business Processes and Interoperability Issues
  - Business Processes and Supply-Chain Issues
  - Process Modeling within Augmented Reality
  - Digital Business Models
  - Visualizations of Business Process Models
  - Inter-Enterprise Collaborations
  - Business–IT Alignment
- Software Specification
  - Software Ecosystems
  - Composition of (IT) Services and Service-Level Agreements
  - Micro-Service Architectures
  - Specification of Context-Aware, Self-Managing, and Value-Sensitive Systems
  - Product Variability
  - Software Development Monitoring
  - Software Defect Management
  - Vehicle Navigation Applications
- Crosscutting Concerns
  - Information Security
  - Privacy
- Analytics
  - Fuzzy Decision-Making
  - Data Analytics
  - Processing of Uncertain Data
- Hot Topics in Technology and Innovation
  - Blockchain Technology
  - Internet of Things

BMSD 2018 received 76 paper submissions from which 35 papers were selected for publication in the symposium proceedings. Of these papers, 14 were selected for a 30-minute oral presentation (full papers), leading to a full-paper acceptance ratio of 19% (compared with 26% in 2017) – an indication of our intention to preserve a high-quality forum for the next editions of the symposium. The BMSD 2018 keynote lecturers and authors are from: Austria, Belgium, Bulgaria, Denmark, Finland, Germany, Indonesia, The Netherlands, Russia, Slovenia, Sweden, Switzerland, Turkey, UK, and USA (listed alphabetically), i.e., a total of *15* countries (compared with 20 in 2017, 16 in 2016, 21 in 2015, 21 in 2014, 14 in 2013, 11 in 2012, and 10 in 2011) to justify a strong international presence. Four countries have been represented at all eight BMSD editions so far: Bulgaria, Germany, The Netherlands, and UK, indicating a strong European influence.

BMSD 2018 was organized and sponsored by the Interdisciplinary Institute for Collaboration and Research on Enterprise Systems and Technology (IICREST) and co-organized by Vienna University of Economics and Business (WU Vienna), being technically co-sponsored by BPM-D. Cooperating organizations were Aristotle University of Thessaloniki (AUTH), Delft University of Technology (TU Delft), the UTwente Center for Telematics and Information Technology (CTIT), the Dutch Research School for Information and Knowledge Systems (SIKS), and AMAKOTA Ltd.

Organizing this interesting and successful symposium required the dedicated efforts of many people. Firstly, we must thank the authors, whose research and development achievements are recorded here. Next, the Program Committee members each deserve credit for the diligent and rigorous peer-reviewing. Further, we would like to mention the excellent organization provided by the IICREST team (supported by its logistics partner, AMAKOTA Ltd.); the team (our gratitude to Aglika Bogomilova and Tania Manova) did all the necessary work for delivering a stimulating and productive event, supported by our Austrian Colleagues Rebecca Runge and Roman Franz. We are grateful to Springer for their willingness to publish the current proceedings and we offer special compliments to Ralf Gerstner for all his support and also to Christine Reiss for her professionalism, patience, and great collaboration with regard to the proceedings preparation. 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 inspiring reading. We look forward to meeting you next year in Lisbon, Portugal, for the Ninth International Symposium on Business Modeling and Software Design (BMSD 2019), details of which will be made available on http://www.is-bmsd.org.

June 2018 Boris Shishkov

Organization

Chair

Boris Shishkov	Bulgarian Academy of Sciences/IICREST, Bulgaria

Program Committee

Hamideh Afsarmanesh	University of Amsterdam, The Netherlands
Marco Aiello	University of Groningen, The Netherlands
Mehmet Aksit	University of Twente, The Netherlands
Paulo Anita	Delft University of Technology, The Netherlands
Paris Avgeriou	University of Groningen, The Netherlands
Dimitar Birov	Sofia University St. Kliment Ohridski, Bulgaria
Frances Brazier	Delft University of Technology, The Netherlands
Ruth Breu	University of Innsbruck, Austria
Barrett Bryant	University of North Texas, USA
Cinzia Cappiello	Politecnico di Milano, Italy
Kuo-Ming Chao	Coventry University, UK
Samuel Chong	Capgemini, UK
Dimitar Christozov	American University in Bulgaria – Blagoevgrad, Bulgaria
Jose Cordeiro	Polytechnic Institute of Setubal, Portugal
Claudio Di Ciccio	WU Vienna, Austria
Jan L. G. Dietz	Delft University of Technology, The Netherlands
Teduh Dirgahayu	Universitas Islam Indonesia, Indonesia
John Edwards	Aston University, UK
Hans-Georg Fill	University of Vienna, Austria/University of Bamberg, Germany
Chiara Francalanci	Politecnico di Milano, Italy
J. Paul Gibson	Telecom and Management Sud Paris, France
Rafael Gonzalez	Javeriana University, Colombia
Norbert Gronau	University of Potsdam, Germany
Clever Ricardo Guareis de Farias	University of Sao Paulo, Brazil
Jens Gulden	University of Duisburg-Essen, Germany
Ilian Ilkov	IBM, The Netherlands
Ivan Ivanov	SUNY Empire State College, USA
Marijn Janssen	Delft University of Technology, The Netherlands
Gabriel Juhas	Slovak University of Technology, Slovak Republic
Dmitry Kan	AlphaSense Inc., Finland
Stefan Koch	Johannes Kepler University Linz, Austria
Michal Krcal	Masaryk University, Czech Republic

John Bruntse Larsen	Technical University of Denmark, Denmark
Kecheng Liu	University of Reading, UK
Leszek Maciaszek	Macquarie University, Australia/University of Economics, Poland
Jelena Marincic	ASML, The Netherlands
Hermann Maurer	Graz University of Technology, Austria
Heinrich Mayr	Alpen Adria University Klagenfurt, Austria
Nikolay Mehandjiev	University of Manchester, UK
Jan Mendling	WU Vienna, Austria
Michele Missikoff	Institute for Systems Analysis and Computer Science, Italy
Dimitris Mitrakos	Aristotle University of Thessaloniki, Greece
Ricardo Neisse	European Commission Joint Research Center, Italy
Bart Nieuwenhuis	University of Twente, The Netherlands
Olga Ormandjieva	Concordia University, Canada
Mike Papazoglou	Tilburg University, The Netherlands
Marcin Paprzycki	Polish Academy of Sciences, Poland
Oscar Pastor	Universidad Politecnica de Valencia, Spain
Prantosh K. Paul	Raiganj University, India
Barbara Pernici	Politecnico di Milano, Italy
Doncho Petkov	Eastern Connecticut State University, USA
Gregor Polancic	University of Maribor, Slovenia
Henderik Proper	Luxembourg Institute of Science and Technology, Luxembourg
Ricardo Queiros	Polytechnic of Porto, Portugal
Jolita Ralyte	University of Geneva, Switzerland
Stefanie Rinderle-Ma	University of Vienna, Austria
Werner Retschitzegger	Johannes Kepler University - Linz, Austria
Wenge Rong	Beihang University, China
Ella Roubtsova	Open University, The Netherlands
Irina Rychkova	University of Paris 1 Pantheon Sorbonne, France
Shazia Sadiq	University of Queensland, Australia
Stefan Schoenig	University of Bayreuth, Germany
Andreas Sinnhofer	Graz University of Technology, Austria
Valery Sokolov	Yaroslavl State University, Russia
Richard Starmans	Utrecht University, The Netherlands
Hans-Peter Steinbacher	FH Kufstein Tirol University of Applied Sciences, Austria
Coen Suurmond	RBK Group, The Netherlands
Bedir Tekinerdogan	Wageningen University, The Netherlands
Ramayah Thurasamy	Universiti Sains Malaysia, Malaysia
Roumiana Tsankova	Technical University of Sofia, Bulgaria
Damjan Vavpotic	University of Ljubljana, Slovenia
Han van der Aa	Humboldt University of Berlin, Germany
Marten van Sinderen	University of Twente, The Netherlands
Alexander Verbraeck	Delft University of Technology, The Netherlands

Barbara Weber	Technical University of Denmark, Denmark
Hans Weigand	Tilburg University, The Netherlands
Roel Wieringa	University of Twente, The Netherlands
Dietmar Winkler	Vienna University of Technology, Austria
Shin-Jer Yang	Soochow University, Taiwan
Benjamin Yen	University of Hong Kong, SAR China
Fani Zlatarova	Elizabethtown College, USA

Invited Speakers

Jan Mendling	WU Vienna, Austria
Roy Oberhauser	Aalen University, Germany

Abstracts of Keynote Lectures

Blockchains for Business Process Management – Challenges and Opportunities

Jan Mendling

Institute for Information Business, Vienna University of Economics and Business, Vienna, Austria
jan.mendling@wu.ac.at

Abstract. Blockchain technology offers a sizable promise to rethink the way inter-organizational business processes are managed because of its potential to realize execution without a central party serving as a single point of trust (and failure). To stimulate research on this promise and the limits thereof, we have written a position paper on the challenges and opportunities of blockchain for Business Process Management (BPM) with various experts in the field. In this talk, I summarize these challenges and opportunities alongside two established frameworks, namely the six BPM core capabilities and the BPM lifecycle, and detail seven research directions for investigating the application of blockchain technology to BPM.

The Convergence of Business, Software Development and IT Operations and the Next Wave

Roy Oberhauser

Computer Science Department, Aalen University, Aalen, Germany
Roy.Oberhauser@hs-aalen.de

Abstract. Market pressure for faster software deployment cycle times gave rise to DevOps. Now BizDevOps is heralded as the next wave of organizational change to more tightly integrate the business into the organization's software development and IT operations culture and processes. This talk will highlight certain implications of this trend for both enterprise and software engineering, and will hypothesize the logical progression beyond BizDevOps as the next wave to affect our future organizational and software systems.

Contents

Short Papers

Full Papers

Business, Contracts, Information

Coen Suurmond(✉)

RBK Group, Keulenstraat 18, 7418 ET Deventer, The Netherlands
csuurmond@rbk.nl

Abstract. In the prevailing mechanical-rational view on organisations, business processes and information systems the engineering approach is the road to solutions. A contrasting view can be found in approaches that start from the social character of organisations and/or the role of natural language in information and communication. In this paper I want to examine this discussion from another and perhaps unusual angle: an analysis of the legal contract as representation of a business agreement. Classical contract law considers a contract as a discrete and fully specified business exchange, while Macneil states that such a discrete contract cannot exist and that every contract has relational aspects based on expectations and trust. For Macneil, a contract is not about some fictional discrete exchange, but rather about a mutual agreement of getting things done in a partially undetermined and uncertain future. This debate in law has meaning both for Business Modelling and for Software Design: Business Modelling should be based on how real business is done and not on some rationalistic idealised view on business, and Software Design should not aim at the fictive monolithic integrated system, but on the contrary provide an open structure that allows for human intervention and that allows for integration with other and heterogeneous information flows.

Keywords: Contract law · Enterprise information systems · Rationality Reasonability

1 Introduction

The role of an Enterprise Information System (further to be designated by EIS) is to support the business of the company, i.e. delivering value to the customer (why does a customer buy the products of this company) and delivering value to the company (both as margin between revenues and costs, and as the provision of specific capabilities for delivering value to the customer). Often the EIS has been regarded as coinciding with the IT systems. Over the decades the IT community has been expanding its territory from delivering software solutions, to requirement engineering, business process design and management, enterprise architecture, and enterprise engineering. Both phenomena are indicators of an engineering viewpoint on companies and on information systems. A typical definition of an enterprise from the viewpoint of the IT community is: “a goal oriented cooperative to be implemented by people and means” [1]. This definition reveals a rational engineering approach where a designed construct is “to be implemented”. A definition of enterprise engineering from the same way of thinking is: “Enterprise Engineering is defined as the body of knowledge, principles,

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 3–17, 2018.
https://doi.org/10.1007/978-3-319-94214-8_1

and practices to design an enterprise" [2]. And Capgemini's Integrated Architecture Framework is described as "a toolbox that contains processes, products, tools and techniques to create all types of architectures which are intended to shape businesses and the technology that supports it." [3]. An interesting point in these definitions is that while they are using the term "design", the term really should be "redesign". Starting and growing enterprises are entrepreneurial acts, the thinking about enterprise architecture and enterprise engineering necessarily is about existing enterprises that are already structured over time both by conscious decisions and by evolving patterns.

In this paper I want to break with the habit to approach the enterprise information system primarily from the rational IT design viewpoint. I want to investigate which general and fundamental requirements for an enterprise information system might be derived from approaching the information system from a fundamental business viewpoint. The basic idea for this paper was triggered by the observations of John Kay about two types of contracts used in business, classical contracts and relational contracts, in his book about the foundations of business success [4]. Relational contracts are not fully specified and the business partners are expected to act according to the agreement as intended, the written text of the contract being an insufficient representation of the agreement. This analysis turned out to be very fruitful in more than one way. Firstly, it shows how business as economic exchange cannot be fully represented in written and formalised texts. Secondly, there are interesting analogies in the thinking about classical contracts and the thinking of the IT community about information systems. In this paper I will start with a discussion about the place of rationality, and about other foundations of human choice and behaviour. This is followed by the main sections about contract law and about doing business. An analysis of kinds of information involved in contracts and business precedes the section where consequences for the design of enterprise information systems are drawn.

2 Place of Rationality

Humans as social animals are expected to behave according to the rules and habits of their social group. Humans as responsible members of human society must be able to give reasons for their behaviour. Humans as rational beings are expected to behave in a consistent way: contradictory reasons are not accepted. Humans as logical beings endowed with the faculty of ratiocination can use logical reasoning and logical models to think things through. Each of the stages in this list is building upon the former stage. The problem with many theories is that the last stage is taken as the standard to gauge human behaviour. Subsequently the theories try to explain the human shortcomings, and to provide methods to deal with them. If, however, we take the second and third stage as fundamental to human society, we will perceive and analyse human behaviour in a different light.

A prime example of reasoning from a rational model can be found in a model that is central to the theory of general economic equilibrium. This Arrow-Debreu model, that mathematically formalises Adam Smith's notion of the invisible hand on market exchanges and shows that "a co-ordinator is not necessary to achieve a co-ordinated outcome" [5]. This model is valid under certain economic assumptions: "all markets are perfectly competitive and the households and firms which trade in them are materialistic and self-regarding" [5]. As Kay nicely formulates at the halfway-point of his book about economics: "The fundamental theorems of welfare economics rest on the assumptions of the Arrow-Debreu model, and if that model were a correct description of how markets work, this book would end here". Kay then continues with another 200 pages about the ways real markets are imperfect. Near the end of his book, in discussing the American Business Model (ABM), Kay writes: "Markets function effectively only if they are embedded in social institutions which are poorly – if at all – accounted for within the ABM" [5]. The presupposed individualistic rational behaviour of economic agents on markets does not exist, a market is a social institution embedded in other social institutions. Newer economic theories from the last decades analyse cases with cooperative economic behaviour [6, 7] and emphasise the embeddedness of economic activities in social institutions [8, 9].

Herbert Simon (in the middle of the last century) and Daniel Kahneman & Amos Tversky (at the end of last century) have studied the rationality of decision making in organisations. Simon formulated his findings as: "Rationality, then, does not determine behavior […] Instead, behavior is determined by the irrational and nonrational elements that bound the area of rationality […] Two persons, given the same possible alternatives, the same values, the same knowledge, can rationally reach only the same decision." [10]. Rationality is bounded by limitations to the capacity of the individual to process information, as well as by the costs of obtaining information (cf. Coase on transaction costs). Kahneman discusses in his recent popular book "Thinking, fast and slow" the findings of the research he started in close cooperation with Tversky [11]. Kahneman describes two modes of thought (Kahneman notes that this distinction has been around in psychology research for a long time), called system 1 and system 2. The psychological experiments demonstrate that much of human decision making would be based on emotional, stereotypic, unconscious, in short: non-rational "System 1 thinking". The "System 2 thinking" is about rationally thinking things through. The intuitive System 1 does the fast thinking, the much slower System 2 "monitors System 1 and maintains control as best as it can within its limited resources" [12]. Both strands of psychological research are perceived to underpin the assumption that to be human is to be "fully rational". Shortcomings to this ideal are either irrational or nonrational.

A corollary of the paradigm of rational individual man is the same engineering approach to both technical and social problems. Friedrich von Hayek has aptly formulated a common trait in the rationalistic and mechanistic viewpoint on our world where solutions for social issues are to be engineered as having: "an exaggerated belief in the powers of individual reason and a consequent contempt for anything that has not been consciously designed by it or is not fully intelligible to it" [13].

If we look at human behaviour from an evolutionary viewpoint, however, humans are basically social organisms living in a social and natural environment. According to Von Uexküll organisms are interacting with their environment in terms of their perceptual world and operational world, or "Merkwelt" and "Wirkwelt" [14–16]. The Merkwelt and Wirkwelt indicate the natural and social space that can be perceived and acted upon by the individual organism, and the organism will develop a repertoire of patterns to deal with recurring situations in their Merkwelt and Wirkwelt. On top of these basic natural mechanisms humans do have a more developed language, consciousness, sense of responsibility, and the faculty of rational thinking, checking for inconsistencies and they are used for thinking through possible behavioural alternatives and also to reflect on and analyse situations.

The research of Adriaan de Groot on thought processes in chess has shown that the intellectual level of top chess players is to be found for an important part in their perceptual competences. Although the game of chess is a highly codified abstract system that seems to belong exclusively to the realm of Kahneman's System 2 thinking, De Groot's research demonstrated that top chess players inhabit a Merkwelt and Wirkwelt that differs significantly from lower level players [17].

In business, experienced people know their business and are skilled in reading all kinds of soft information, like De Groot's chess players. They use a combination of background knowledge, perceptual competences, negotiations and calculations in preparing and closing their deals. In explaining their behaviour they will be able to provide reasons for why they acted as they did, but this does not mean that their decisions were the result of a deliberate reasoning process. Rationality is of course important, but by far not the only thing.

3 Contracts, Business Agreements, and the Law

In a business contract the parties promise to each other to deliver goods, services or money in a mutual exchange under stated conditions. In our modern constitutional state business contracts have both an essential administrative function and an essential business function. The administrative function is that each transfer of goods, services or money must be based on a contract, and the law does not allow just to hand over money to someone without a sufficient specific legal ground or title. The business function is that the parties can rely on the law to enforce their agreement.

In the Treitel, an established treatment of British contract law, Peel defines a contract as "an agreement giving rise to obligations which are enforced or recognised by law. The factor which distinguishes contractual from other legal obligations is that they are based on the agreement of the contracting parties" [18]. Austen-Baker and Qi Zhou provide in their introductory book 'Contract in Context' an example of a common legal definition of contract ("a promise or a set of promises for the breach of which the law provides a remedy"), and then ask two question (1) 'what are the purposes of contract?', and (2) 'what are the purposes of contract law?' [19]. Their answer to the first question is that contracts are about obtaining cooperation by the exchange of promises (a business deal) or by securing protection (think of an insurance contract). The purpose of contract law is to provide enforceability and to provide the conditions under which a contract will be enforced and how. These purposes can be viewed as the provision of a stable and predictable business environment, thus reducing the transaction costs in doing business (cf. Coase later in this paper).

The American Uniform Commercial Code (2017–2018 edition) gives in Sect. 1-201 the following interesting pair of definitions for contract and agreement: "**Contract**, as distinguished from "agreement", means the total legal obligation that results from the parties' agreement as determined by the Uniform Commercial Code as supplemented by any other applicable laws" and "**Agreement**, as distinguished from "contract", means the bargain of the parties in fact, as found in their language or inferred from other circumstances, including course of performance, course of dealing, or usage of trade as provided in Sect. 1-303" (my emphasis). [20] In this definition, the contract is formal and belongs to the legal sphere and is about legal enforceability, and the agreement is factual and belongs to the business sphere. However, the formulation of "reasonable time" in Sect. 1-205 of the same code shows that the distinction between business habits and formal conditions are blurred: "Whether a time for taking action required by the Uniform Commercial Code is reasonable depends on the nature, purpose, and circumstances of the action". So, when a contract specifies an action within "reasonable time", its interpretation and the legal consequences will depend on the soft criteria that belong to the sphere of the general habits for this kind of business agreement.

Traditionally, the legal discourse about contracts is founded on two pillars: (1) parties have freedom of contract and (2) the contract is legally enforceable. The interpretation of the second pillar has been subject to fundamental discussions in law and in legal theory. When disputed in court, should the contract be interpreted literally, or should the context be taken into account? The opposing views are formulated by Hugh Beale in 2013: "Relational theory suggests that contract law should take the context into account unless the parties have agreed that the contract should be treated discretely. English law often seems to start at the other end: the context should be ignored unless the parties have provided for it to be taken into account." [21]. The idea of the discrete contract text that does not need any knowledge of background or context is nicely formulated by Beale as "the Chancery lawyer's ideal that the judge could answer every conceivable question about the terms of the contract without taking his or her eyes from the document" [21].

The conventional idea of the contract is that the contract is completely representing the agreement by the contract parties. If the contract parties would find some condition important enough they should write it down as a proviso in the contract. Ian Macneil has disputed this idea from his first publication in 1960. In his Rosenthal Lectures of 1979, published in 1980 as The New Social Contract, he recapitulates his analysis of the nature of the contract [22]. The classical convention of the fully specified discrete contract is based on a fiction: "[a] discrete contract is one in which no relation exists between the parties apart from the simple exchange of goods". His principal objection to this fiction is that an exchange between parties presupposes communication and social conventions about an exchange, therefore there is some relationship between the parties. His practical objection is that most contracts (1) have a duration, (2) deal with exchanges in a not fully known future, (3) are not fully specified and (4) are based on some reciprocal trust between the contract parties that the other party will fulfil the agreement as meant, and will not wiggle out of his obligations by sticking to the letter of the contract. In classical contract theory, the discrete contract is the norm and relational aspects are either neglected, or treated as aberrations. As Macneil writes: "The structure of the principles of general contract law, instead of reflecting material relations as they actually exist, is based entirely on the nonexistent discrete transaction. Other forms of reciprocity, dominant in the real world, have been eliminated." [23]. And in a formulation that I will use again in the section about Enterprise Information Systems, Macneil writes about the bias of thinking about contracts in terms of classical contract theory: “Contracts are about getting things done in the real world … even if a law-oriented definition encompasses every contract … it will inevitably be perceived as narrower because it immediately tells us to think about law. If we wish to understand contract … we must think about exchange and such things first, and law second” [22]. The analogy between the world of law and the world of IT is that in both worlds business deals are reduced to formal aspects. What the business parties want to achieve by their deal is pushed aside and the deal is reduced to its formal representation in text (contract) or in business objects (IT).

Any economic exchange is based on a mutual commitment to give the other party his or her due. A clear cut example of a discrete transaction would be buying a paper at the newsstand for its advertised price. Buyer and seller know exactly what is exchanged: today’s paper is exchanged for a fixed amount of money. But, as Macneil analyses, this kind of fully specified discrete exchange is embedded in a web of social relations, and the transaction is only possible because the contract partners in this simple transaction rely on the availability and stability of the background societal conventions [22]. As an example: the small Euro-coins are worth 50, 20, 10, 5, 2 and 1 cent. In the Netherlands the smallest coins are not used and cash payments are rounded to the nearest 5 cents. But what means “are not used” in this statement, and what does it mean in a real transaction? If the Netherlands as a country would be isolated from its neighbour countries in the euro-zone, “are not used” would mean “are not in circulation” or “cannot be found in the purse of people in the Netherlands”. But in Germany the small coins are used, and returning from Germany I would have such small coins in my purse. Buying a paper, I could get rid of these small coins by paying for the paper. The newsstand will not like this payment with coins that are not in circulation, at the same time it is clear that the small coins are valid euro coins. The question is: are these small coins legal tender in such an exchange? The point here is that the small coins are part of the exchange if they are accepted by the seller, and his willingness to accept will be based partly on legal structures, partly on the fuss for him to get rid of those coins later on, and partly on his relation with his customers. Will he refuse, will he complain, or will he comply? His reaction will not only be determined by the legality of payment with these coins, but also by the weighing of the seller of his own convenience against being nice to this customer in this moment, and perhaps also by the impact of his behaviour on bystanders at the newsstand. This is a small example of a mechanism that is fundamental for very many business deals. See for example the analysis by David Campbell of a case from 1929 about the sale of timber staves [23]. Two types of specification are discussed in this analysis: measurements of the timber vs fitness-for-use. The general point is here: is a buyer allowed to refuse a delivery because it does not exactly match the specifications in the contract, where accepted business practice would give the seller some leeway? If the answer would be that the buyer has this freedom, he can use this in bad faith. Generally speaking, business deals presuppose some reasonable amount of good faith and business would be worse off when it would be possible to escape from a deal using literal interpretation of contracts.

In an approach that differs from Macneil and addresses other non-discrete aspects of contracts John Wightman has analysed the role of implicit understandings in contracts [24]. He distinguishes three forms of implicitness: (1) “implicit understandings stemming from a shared language, knowledge of the social institution of money, currency, and a shared 'market mentality'"; (2) "implicit understandings which emerge over time between the parties to a particular contract", and (3) "implicit understandings about how commercial relations in a particular sector are carried on". Next in his analysis, Wightman distinguishes two types of contracting: (1) the contracting community model of contractual relations, and (2) the personal consumption model of contractual relations. In the first model the contracting parties have a shared knowledge about the physical nature of the products being traded, about the way the trade is carried on, and about the practices that are used when hitches in performance arise. In the second model complex products (which require specialist knowledge to appraise) are bought by an infrequent and therefore inexperienced buyer who is driven by individual preferences. Wightman continues by discussing when and how the law recognises the implicit elements of a contract, but that part is less interesting for this paper.

4 Doing Business

Our market oriented society is based on the principle that actors can freely engage in economic exchanges that leave each of them better off. The concept of comparative advantages, first systematically formulated and analysed by David Ricardo in 1817 [21], shows that economic exchange can be advantageous to both actors. Hence, doing business should not be reduced to some zero-sum game where one economic actor tries to lure another one into some unfavourable deal. This kind of behaviour exists of course, but normal business is based on economic exchanges that benefit both business partners. Depending on the individual business acumen, the general business context, the specific needs and the possible alternatives in a given context, one or the other can get a relatively better deal at some expense of the other. But in a healthy economic market situation (to be warranted by law and by other institutions) each of the partners has the freedom either to enter or to refuse the business deal, depending on his individual assessment of the gains and the costs of the deal. The gains and costs are to be taken as not only measured by immediate financial effects, but also by future financial effects (such as the revenues of support contracts in IT projects or the selling of cartridges in the printer market) and the imponderable gains such as the development of the portfolio of the company or the change in the market position by winning this new customer.

Businesses get things done by engaging in relations with suppliers and customers (for material goods and services), and with employees (for labour). What exactly is exchanged (and when, and under what conditions), is in many cases not fully specified but each business partner assumes that the other partner will behave according to the mores of the trade and the intention of the agreement. In terms of information this is a significant difference: the first is about general and rather stable background patterns of meaning and interpretation, the second is about information that is specific for the individual contract. Note that this distinction is gradual rather than fundamental: two businesses dealing with each other over some longer time will develop certain behavioural patterns and business terms will be more and more general and implicit. For an individual contract the business partners are only expected to make note of specifications or conditions that differ from the habitual pattern. And the general patterns of the trade of course emerge from patterns between individual businesses over time.

From this nature of business some essential factors that contribute to business success can be derived. Firstly, there is the role of trust and reputation. Enterprises will attract more business when they are perceived as more trustworthy. Incidentally, this is one of the social and extra-legal enforcement mechanisms for sticking to the contract as intended. John Kay has written extensively about this aspect in his book Foundations of Corporate Success, and he uses the term distinctive capabilities to indicate this kind of intangible assets of an enterprise [4].

Secondly there are the transaction costs. As Coase has pointed out, making a contract for some business exchange comes at a cost [27]. Therefore, relying on implicit and stable elements in a contract will save transaction costs. Also, making a gentleman's agreement based on mutual recognition of intentions and expectations will cost less than drawing up a full-blown contract. Repeating business with the same partner can also reduce transaction costs. The relevance of this becomes clear when an enterprise decides to replace existing supply channels by a tendering process (contracting volumes of goods for the duration of a limited amount of time). Generally speaking, tendering drives the prices down and drives the transaction costs, adaptation costs and enforcement costs up. The price effect is very visible, the costs much less and will partly appear only over time. The costs are caused by the characteristics of the tendering process: the specifications must be more detailed (otherwise the bids cannot be compared), and there are initial costs due to the new relations with new suppliers that disturb existing patterns in business processes and generate adaptation costs. Enforcement costs may be expected to be higher for two reasons: (1) the very competitive bidding process drives the prices down resulting in smaller margins for the supplier, and (2) the tendered contract is limited in its duration and the supplier will have less incentives to invest in the continuity of the relationship with the customer. Both forces will erode the cooperative relationship between supplier and customer and the supplier will be incentivised to stick to the specifications and the supplier will have bigger costs in enforcing the contract terms (or, in case of real deviation of contract terms, incur extra costs to deal with the deviant situation).

The third success factor does not belong to the individual business deal but to the general business environment: the role of institutions. Reliable institutions decrease transaction costs, both in closing and formalising a deal and in enforcing a deal. Business partners can rely on existing social mechanisms and do not have to find out their own solutions over and over again. This kind of trust is analysed by the theorists of the New Institutional Economics [8, 9].

To summarise: doing business is about getting things done with a positive margin for the company. Generating and maintaining a healthy margin is not only related to net sales and production costs, but also to the costs of transactions and enforcement. Stable relationships and mutual trust will have a positive influence on driving these costs down. Reputation and inducing a feeling of mutual confidence are important in the sales process, as is trusting the supplier in the purchase process (or in hiring employees). Of course, the statements above are very general statements. Depending on the kind of business the dependency on implicit aspects and relationship aspects will vary. Spot markets are much more discrete than integrated supply chains. But, as Macneil has written, any business is embedded in social structures and complete discreteness is an illusion.

5 Information in Business Agreements

Boisot analyses aspects of information in his work about the Information Space [28]. In this three-dimensional space he distinguishes (1) the degree of codification, (2) the degree of abstraction, and (3) the degree of diffusion. Tacit personal knowledge would refer to information located in this space as not codified, not shared, and concrete. This is the kind of information that cannot be codified and subsequently diffused. However, it might be ‘copied’ or ‘absorbed’ by another person by imitation and guidance, like a shrine worker showing an apprentice how to process different kinds of woods. The demonstration is concrete, and the apprentice must learn not only the skills but also to recognise how to apply which skill in which context. This is an individual abstraction process developing with experience. One could say that the concept of mentoring a trainee in a modern organisation is partly based on this principle. Also, the subtitle of Mintzberg’s book “Managers, not MBA’s – A hard look at the soft practice of managing and management development” refers to the skills needed to deal with this kind of information [29]. On the other side of the spectrum, scientific knowledge about the laws of nature is highly codified, abstracted and diffused. Mainstream theoretical physicists from all over the world would essentially speak the same language and would be able to communicate easily with each other.

Boisot emphasises that although in practice codification and abstraction processes often go together, and although both processes are about economising data and data processing, they are fundamentally different. Codification is about “giving a better definition to form, removing it fuzzy edges and allowing a sharper differentiation and focus”, while abstraction is about “shared attributes, thus avoiding the need for independent description and treatment.” [28].

In doing business the business man will deal with at least three different kinds of information. The first kind is the background information about the general context. This kind of information is rather uncodified and can be more or less abstract. It is acquired through a process of acculturation. Each market place, from the rural town cattle market to the global financial market, has its own habits and norms. Participating in a market means acknowledging those social conventions. Part of this kind of information is about the business practices when hitches in performance arise. This is the kind of information Wightman writes about: “This knowledge is based on knowledge about the physical attributes of the product or the service, including what can go wrong with these products/services in general. But it also includes knowledge about how hitches such as shortfalls in the quantity or quality of goods and services are actually handled, or how late payment or other late performance is regarded” [24]. Wightman writes about ‘knowledge’ and not about ‘information’, but this knowledge governs what information is considered to be relevant and to be supplied by either buyer or seller, and it gives the background and context against which the information is interpreted in a given situation.

The second kind of information is concrete and not codified and is about the particular circumstances of the business deal. It is information about the business partner (What is his reputation? Why does he want to sell?) and about the assessment of business opportunities. The expressions "gut feeling" and "smell a rat" indicate the non-verbal and intuitive character of this kind of information. But this information is also about the particular and concrete aspects of the business agreement: for example considering the holiday of your most trusted quality supervisor before deciding when the first delivery of a new supplier of raw material will arrive.

The third kind of information is highly codified and abstract, and is used in more or less complex business calculations. Revenues and costs of an exchange are calculated in order to assess its financial attractiveness. This is the kind of information economic theories are about (e.g. price function calculations or the calculations for the return on investment). This is the kind of information Boisot writes about in the introductory section of the chapter "The structuring of information". Boisot has an interesting observation to offer about classical economics, the subject of Sect. 2 of this paper: "Economic man, as defined in orthodox economic theory, is only allowed to operate in that part of the E-space where data is both highly coded and abstract. Strictly speaking, therefore, economic man has no past." [28] (E-space is the two dimensional epistemological space in Boisot's model). This observation is yet another critique of the simplified view of classical economics, in contrast to the way business is conducted in a social context where recognition of particular circumstances might be as important as highly codified and abstract information. Developing new business is about combining the interpretation and assessment of particular circumstances with calculations that are based on highly codified and abstract information. Doing routine business is often based on evolved and not sharply codified social patterns of exchange, each business partner expecting from the other consistent behaviour.

In formally writing down the business deal in a contract for two reasons the information will have to be codified. The terms of the contract must relate to law, and the contract must express the parameters of the business deal as clearly as possible. However, as the discussion about contract law in the previous section shows, the written contract will not represent the business deal completely. The relational aspects of the business deal as mutually agreed on by the business partners will in many cases not be fully expressed in the discrete (codified) terms of the written contract. This relational aspects are partly particular to this individual contract (when the business partners both recognise particular circumstances), partly particular to the long-standing relationship and habits between the business partners, and partly general to the kind of trade.

In the paragraphs above we have discerned three kinds of information in finding and closing business deals, respectively belonging to the characteristics of the trade and the market place, belonging to the individual business deal and belonging to the business calculation models. In executing and monitoring the business deals (the domain of the Enterprise Information Systems) both the discrete terms of the written contract and the relational information belonging to the individual contract, the business relationship, and the kind of trade must be taken into account. Satisfying a business deal cannot be reduced to formally fulfilling the contract terms, but must also obey the implicit agreement about the way the business agreement will be executed or implemented.

In business practice, the contract partners know about the (non-)specificity of the contract terms. The quantity might be exact, or an approximation. Delivery time might be exact, or just sometime before 4pm. Sometimes the product specifications will be very precise (measurements of parts for an Airbus), sometimes the specifications are more descriptive (colour and texture of upholstery of furniture). The third kind of information is about the business practices when hitches in performance arise. As Wightman writes: “This knowledge is based on knowledge about the physical attributes of the product or the service, including what can go wrong with these products/services in general. But it also includes knowledge about how hitches such as shortfalls in in the quantity or quality of goods and services are actually handled, or how late payment or other late performance is regarded.” [24] Wightman writes about ‘knowledge’ and not about ‘information’, but this knowledge governs what information is considered to be relevant and to be supplied by either buyer or seller, and it gives the background and context against which the information is interpreted in a given situation.

Business partners expect from each other that the other party will not offload all kinds of problems and irregularities to their business partners. In the initial contract the business partners will seek an agreement about which problems are to be solved to which degree by which partner (sometimes: at which additional cost), later on practices will develop and patterns will evolve. Developments will be monitored by the business partners on a regular basis, sometimes leading to an adjustment of the earlier agreement. And, given these agreements and patterns as background, the business partners expect from each other that deviations from their standardised exchanges will be accompanied by relevant information (the fifth kind of information). For example, when the time window for delivery is normally ‘somewhere in the afternoon’, and the buyer needs the goods at 1pm at the latest, this will be mentioned in the buying order. But also information from the seller: next week we have auditors on the shop floor, so expect some delay in delivery.

To summarise: the information involved in business agreements is partly explicit and discrete, partly implicit, partly about factual specification, partly about modality and intentions. Implicit information for standard situations (including things that ‘normally can go wrong’) can partly be made explicit and discrete in business rules, In some normal situations and in unforeseen circumstances intentions and expectations are to be factored in the assessment of the situation and the decision how to act, in these cases information must be interpreted in context.

6 Enterprise Information Systems

According to the relational approach to contract law it is not possible to represent all aspects of a business agreement in the business contract. Intentions, promises, general habits and specific considerations of the contract parties cannot be fully written down. If this is not possible, a fortiori it will not be possible to represent a business agreement fully in the formal sign system of an IT system. And if an enterprise information system is the organisation and structuring of all information flows that are relevant to the conducting of the business of the enterprise, while an IT system is fundamentally incapable of representing all relevant information of business agreements, then an IT system cannot be more than a partial solution for an enterprise information system. Other, uncodified (or less codified) information will be taken from notes, memos, and oral information about context and intentions of the business agreement. Two main issues with IT systems for representing information are the lack of modality and the excess of precision ("about 3000 kg by the end of next week" will in a typical IT system have the same representation as "exactly 3000 kg on Friday next week").

There are instructive analogies and differences to be drawn from the comparison between contract law and IT systems in relation to business agreements. One analogy was already mentioned above, the inability of both for representing business agreements fully. Both contract law and IT systems are hampered by their own constraints of their sign systems. In contracts it is the 'legalese' as a formalised variant of natural language, in IT systems it is the codification of terms in predefined categories. An important difference here is that in contracts the full register of natural language might be employed to express nuances, intentions, and considerations (not all of which might be legally enforceable, but at least it is written down), while IT systems ask for rigid codification (see the example above of "end of next week" and "Friday next week").

Another analogy is the tendency of IT and legal professionals to take the representation of the business agreement for the agreement itself, thereby substituting business reality by its incomplete representation. While business partners would communicate about the content of their agreement, and perhaps argue about intentions and interpretations, lawyers and IT professionals are discussing about the reduced information in their respective representations. And within companies IT users increasingly take the sales orders in the IT system for business reality. However, here an essential difference between the representation of a business agreement in a contract and in an IT system must be taken into account. While the contract is by itself an essential part of business reality (ultimately the contract is signed by both parties and may be discussed in court), the IT representation is 'just' an unilateral choice of one company and cannot bind the other company.

The DEMO method provides a striking example of the similarity of the approach of classical contract law and an IT approach. The fundamental transaction in DEMO is precisely the kind of discrete event that is central in classical contract law. In a given situation of the world the client successively requests a fact of the supplier, the supplier creates the required fact, the client accepts the result, and the transaction is finished. As in the discrete contract, DEMO assumes that transactions are fully specified and instantaneous. No duration, no unspecified implication and no trust seem to be involved. And especially: all information that is needed for the different stages of preparing and executing the transaction is explicit, complete and atomic.

A business agreement will generally be based on delivering value in more than one sense: (1) the financial value that contributes to the financial health of the parties, (2) the use value of the product or service to be delivered to the customer (this would be the reason for buying), (3) the process value of the mutual adjustment of the processes of supplier and customer, and (4) the relationship value of creating or maintaining the relationship between the business parties that serves the longer term stability of the companies. The awareness and monitoring of each of these value concepts belongs to the enterprise information system, and in this area a lot of concrete and less codified information is involved. When problems arise with satisfying business agreements because of limited resources, this value-related information is key in weighing the options and deciding which agreements to satisfy and which not to. Rigid business rules may decide the simple cases, but often this kind of decisions requires experience and background information about customers, processes and business agreements. And, because a company will have simultaneously many agreements and contracts with many kinds of partners (buyers and sellers, labour (both permanent and temporary), service contracts for maintenance and for cleaning, perhaps covenants with administrative bodies), this kind of decision processes can be rather convoluted.

Because the enterprise information system cannot fully coincide with an IT system, and because the main IT system will always be used in combination with other means of providing information to business processes (more often than not encompassing a plethora of spreadsheets), one of the key design objectives of an enterprise information system should be a well thought-out system of common references. A second key objective should be that in the architecture of the main integrated IT system the key points should be identified where meaningful human intervention in the IT system is possible. On those key points knowledgeable people must be able to combine different kinds of information and must be able to overrule the process flows in the IT system. Together with the system of common references such an approach would allow users to combine and interpret information from different sources and from different sign systems, and to make their own decisions. Such an approach also allows to use the main integrated IT system for stable routine processes in combination with other emergent and less defined information processes. Periodically, the pattern of information flows and information use can be evaluated and the relation between information processing in the IT system and other ways of information processing can be adjusted.

7 Conclusions

The theme of this paper is the opposition between two different views on economics, business, law, and information systems: (1) the rationalistic and mechanistic view of our world where social constructs are to be engineered, and (2) the view on our world as constituted by emerging patterns of interactions by responsible and reasonable humans where social life is not dictated by engineers but supported by engineered machines and devices.

In classical economics homo economicus would operate on perfect markets, unbounded by limitations to either availability of information or capacity for information processing. As is proven time and again this rationalistic approach is a reduction of social reality and cannot cope with actual social behaviour. Two answers to this problem have been tried. The first answer is to stand by the rationalistic base model, to consider actual human reasoning and actual social behaviour as falling short of the rationalistic norm, and to find ways to cope with what are considered shortcomings. The second answer is to take reasonability (accountability) and not rationality as the foundation of the social world. A fundamental part of being reasonable is to be consistent, and consistency is a concept that belongs to the realm of rationality. But humans make choices based on values, and must sometimes choose between contradicting rules.

The two main similarities between contract law and enterprise information systems are (1) their subservient role to the business itself, and (2) the adherence of both to written facts and rules. However, "similar" is not "equal". Contract law is subservient to the business deals that are written into contract, but once the contract is written and signed the contract **is** de business deal as far as the law is concerned. The contract is binding for both business parties. This is an essential difference with the representation of the business deal in the IT system of the individual company. To be sure, this IT representation often is considered within the company as "the business deal to be fulfilled", but this representation is the internal and unilateral matter of the individual business. The second essential difference is the "language" used for representation of the business deal. The contract is written in natural language in a legal context, where terms and rules are interpreted according to their conventional meaning in business and in law. The IT representation is coded and interpreted in a formal language of variables and logical rules. Where natural language allows for subtle distinctions by choosing the words and formulations and the language user can make up his own distinctions and categories, the IT systems forces the user to choose from predefined categories that are coded into the system.

In our design of the main IT system of a company we should acknowledge the fact that such is system is to be considered as an important component of an enterprise information system, but that such a system will encompass information sources and processes that are not all based on IT. We should also acknowledge that the IT community must not try to rationalise the business world and to engineer the enterprise. On the contrary, to be subservient to the business means to provide useful instruments like IT systems that can be used in combination with other instruments. IT is just as the law an essential aspect for doing business, but it should know its place.

References

1. Op't Land, M., Proper, E., Waage, M., Cloo, J., Steghuis, C.: Enterprise Architecture – Creating Value by Informed Governance. Springer, Berlin (2009). https://doi.org/10.1007/978-3-540-85232-2
2. Giachetti, R.E.: Design of Enterprise Systems – Theory, Architecture, and Models. CRC Press, Boca Raton (2010)
3. Van't Wout, J., Waage, M., Hartman, H., Stahlecker, M., Hofman, A.: The Integrated Architecture Framework Expalined. Springer, Berlin (2010). https://doi.org/10.1007/978-3-642-11518-9
4. Kay, J.: Foundations of Corporate Success. Oxford University Press, Oxford (1993)
5. Kay, J.: The Truth about Markets. Allan Lane, London (2003)
6. Ostrom, E.: Governing the Commons. Cambridge University Press, Cambridge (1990)
7. Ostrom, E.: Understanding Institutional Diversity. Princeton University Press, Princeton (2005)

8. Furubotn, E.G., Richter, R. (eds.): The New Institutional Economics. Mohr Verlag, Tübingen (1991)
9. Furubotn, E.G., Richter, R.: Institutions & Economic Theory, 2nd edn. University of Michigan Press, Ann Arbor (2005)
10. Simon, H.A.: Administrative Behavior. The Free Press, New York (1976)
11. Kahneman, D., Tversky, A. (eds.): Choices, Values and Frames. Cambridge University Press, Cambridge (2000)
12. Kahneman, D.: Thinking, Fast en Slow. Allan Lane, London (2011)
13. Hayek, F.A.: Individualism and Economic Order. The University of Chicago Press, Chicago (1948)
14. Von Uexküll, J.: Umwelt und Innenwelt der Tiere. Julius Springer, Berlin (1909). https://doi.org/10.1007/978-3-662-24819-5
15. Von Uexküll, J.: A Foray into the Worlds of Animals and Humans. University of Minnesota Press, Minneapolis (2010)
16. Brentari, C.: Jakob von Uexküll – The Discovery of the Umwelt between Biosemiotics and Theoretical Biology. Springer, Dordrecht (2015)
17. De Groot, A.D.: Thought and Choice in Chess. Mouton, The Hague (1965)
18. Peel, E.: The Law of Contract, 14th edn. Sweet & Maxwell, London (2015)
19. Austen-Baker, R., Zhou, Q.: Contract in Context. Routledge, Milton Park (2015)
20. The American Law Institute: Uniform Commercial Code – 2017-218 Edition. Thomson Reuters, Eagan (2017)
21. Beal, H.: Relational values in english contract law. In: Campbell, D., Mulcahy, L., Wheeler, S. (eds.) Changing Concepts of Contract, pp. 116–137. Palgrave Macmillan, Basingstoke (2013)
22. Macneil, I.R.: The New Social Contract. Yale University Press, New Haven (1980)
23. Campbell, D. (ed.): The Relational Theory of Contract: Selected Works of Ian Macneil. Sweet & Maxwell, London (2001)
24. Wightman, J.: Beyond custom: contract, contexts, and the recognition of implicit understandings. In: Campbell, D., Collins, H., Wightman, J. (eds.) Implicit Dimensions of Contract, pp. 143–186. Hart Publishing, Oxford (2003)
25. Campbell, D.: Arcos v Ronaasen as a relational contract. In: Campbell, D., Mulcahy, L., Wheeler, S. (eds.) Changing Concepts of Contract, pp. 138–165. Palgrave Macmillan, Basingstoke (2013)
26. Roncaglia, A.: The Wealth of Ideas – A History of Economic Thought. Cambridge University Press, Cambridge (2005)
27. Coase, R.H.: The Nature of the firm. In: Williamson, O.E., Winter, S.G. (eds.) The Nature of the Firm: Origins, Evolution, and Development, pp. 18–33. Oxford University Press, Oxford (1993)
28. Boisot, M.H.: Information Space. Routledge, London (1995)
29. Mintzberg, H.: Managers, not MBAs. Prentice Hall, Harlow (2004)
30. Dietz, J.L.G.: Enterprise Ontology. Springer, Berlin (2006). https://doi.org/10.1007/3-540-33149-2

Reconciling the Academic and Enterprise Perspectives of Design Thinking

José Carlos Camposano(✉)

Department of Computer Science, Aalto University School of Science, Konemiehentie 2, 00076 Espoo, Finland
jose.camposanomorla@aalto.fi

Abstract. Design Thinking has become popular in the management and innovation context but remains mostly misunderstood, as a result of broad interpretations and the lack of empirical research on the subject. This paper aims to reduce the gap between the academic and industrial perspectives on Design Thinking, by reviewing publications focused on three aspects: (1) studies aimed at defining the concept, (2) empirical case studies about its use or adoption, and (3) models or methods proposed to overcome its main challenges. The existing literature suggests that multiple definitions for Design Thinking coexist with some commonly understood design practices, both among designer and non-designer practitioners alike. The challenge most frequently mentioned is the clash of existing organizational structures with the flexibility and unpredictability of Design Thinking. This paper outlines two different approaches to address such challenge and proposes a definition that brings together the academic and enterprise perspectives of Design Thinking.

Keywords: Design thinking · Management · Innovation · Enterprise Organizations

1 Introduction

Design Thinking (DT) is often portrayed as a multi-disciplinary human-centered approach to innovation [1]. In recent years, the term has gradually found its way into business and management literature, suggesting that the practice of designers can be brought into other fields or industries to tackle their own innovation challenges. Professional narratives and project portfolios have become important channels to disseminate this DT knowledge, with the most prominent examples found in the stories of successful product development and service innovations of the consultancy firm IDEO, as told by the founder's brother Kelly [21] and the current CEO Brown [22]. In his widely popular book "Change by Design", the latter of these two authors devotes an entire section under the title *"What is design thinking?"*. Despite providing a detailed account of the DT process, desirable characteristics of DT practitioners and examples of benefits achieved by organizations that have used DT, the author falls short of providing a concrete definition of DT *itself*. Thus, the reader can only construct his own definition by going through the entire book and putting together the different characteristics and descriptions provided. Some of them are attributed to the concept directly

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 18–31, 2018.
https://doi.org/10.1007/978-3-319-94214-8_2

(e.g. *"exploratory process"*, *"iterative approach"*, pp. 16–17) while others must be inferred from the text (e.g. *"[D]esign thinking begins with skills designers have learned over many decades… integrating what is desirable… with what is technologically feasible and economically viable"*, p. 4).

In contrast to the image portrayed by popular culture, certain scholars have attributed the "true" origins of the concept to the study of professional designers' work and practice inside the academic community [5]. The current understanding of DT – or lack thereof – has been criticized for relying too much on the perspective and cumulative experiences of its practitioners, who tend to omit certain formal aspects of the design field [15]. These divergences in discourse suggest that the term itself is still rather loose, has different meanings or is often misunderstood [1]. Furthermore, despite the wide variety of company whitepapers, blog posts and management books describing successful DT project cases, the generalizability and replicability of their processes and results remains unclear. Some researchers even argue that there is no theoretical body regarding DT, because the concept is tightly related to practice [5]. Nonetheless, executives and managers maintain a growing interest on the topic, because their organizations face increasingly complex challenges and they urgently need to broaden their range of strategies to address them [3]. It becomes relevant then to find a definition of DT that can be agreed upon and understood by scholars and practitioners alike, serving as a base for future discussions integrating both sides.

The problems outlined above also suggest that the lack of systematic, empirical research on DT is gradually producing a disconnection between theory and practice. Paradoxically, the same situation may generate certain reluctance among scholars to pursue research on the subject, widening even more the gap between up-to-date sources of information for practitioners and the scientific literature. Some notable examples of institutions aiming to address these challenges are the d.school at Stanford and the Hasso-Plattner Institute in Potsdam. Nevertheless, there are strong reasons to expand the efforts into a broader practice-based research community, in order to reduce the divergences between DT practice and theory.

The goal of this paper is to bring closer the academic and enterprise perspectives on DT, by analyzing previous studies about its conceptual understanding and empirical application inside enterprises. The expected contribution of this secondary study is two-fold: First, to identify the focus areas and recurrent themes discussed in existing literature, highlighting any gaps that could guide a future research agenda. Second, to propose a tentative – and more agreeable – definition of DT, which could be used in the future to bring together the different discourses from scholars and practitioners.

In general terms, the concept of enterprise is defined by the Merriam-Webster, Cambridge and Collins dictionaries as systematic, intentional or purposeful activities, highlighting the importance, risk or difficulty of such project or undertaking [32–34]. In a narrower sense, the same word is commonly used to describe commercial organizations with business purpose. In the context of this paper, the concept of enterprises is understood as equivalent to large organizations or well-established private companies whose size, rigidness and maturity brings them closer to more traditional management approaches than those found in start-ups or small and medium enterprises (SMEs). The scope of "large commercial business organizations" has been intentionally chosen to narrow down and focus this study on the adoption or implementation of DT under potentially similar conditions (e.g. assuming that larger companies have more hierarchies, well-defined processes, more complexity, less flexibility, etc.). Furthermore, the participation of prominent or well-known companies in previous studies could be considered a factor which increases the relevance and interest on DT inside academia, thus motivating further research – which is one of the main arguments this paper advocates for.

The structure of this paper is as follows: Sect. 2 provides a description of the research steps undertaken to select and analyze the existing literature, outlining three main questions to guide the review process. Section 3 provides a high-level overview of the key themes found in the selected articles and their coverage of each question proposed. Section 4 provides a detailed description of the findings, which have been grouped according to each one of the review questions. Section 5 reflects on the overall research process and clarifies the limitations of this paper. Finally, Sect. 6 reiterates the main findings of the literature review and presents the contributions of this paper, namely a tentative definition of DT and a future research agenda.

2 Research Process

According to Google Trends, the search interest about DT has consistently increased during the last decade [35]. As shown in Table 1, the number of search results obtained at the beginning of 2018 from three online databases of academic publications using the keyword "design thinking" follows a similar trend, which suggests the scholarly community has been also giving increasing attention to the topic.

Table 1. Number of academic publications listed per year for the term "design thinking"

	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
ScienceDirect	20	26	29	50	85	117	82	115	229	226	238
Scopus	34	55	80	104	148	175	169	272	308	404	413
Web of science	18	32	57	50	82	95	104	127	208	288	233
Total	72	113	166	204	315	387	355	514	745	918	884

As Welch and Piekkari [25] point out, qualitative research strives for depth of insight rather than "strongly representative" samples. Thus, "the larger the N" may not necessarily result in better studies, since the richness of conceptual analysis does not always depend on the inclusion of quantitative data collection (e.g. surveys) or statistical analysis methods. Empirical studies in fields such as software engineering are usually too heterogeneous for statistical summaries, but it is still necessary to consider them together with other studies addressing the same or similar questions, in order to interpret their findings with more confidence [26].

Because of the aforementioned reasons, this paper followed a theory-based rather than population-based approach to data selection and analysis. According to Cruzes and Dybå [26, 27], research synthesis is the collective term for a family of methods used to summarize, integrate, combine and compare the findings of different studies about a specific topic or research question. Among those methods, the thematic analysis entails the identification, analysis and report of the patterns or themes found within the primary research data [26]. To conduct this process, Thomas and Harden [29] developed an approach called "thematic synthesis", which facilitates the translation of concepts between studies, even though they may have not been originally expressed using the same words. This method draws on elements from both meta-ethnography and grounded theory, particularly the development of analytical themes comparable to "third-order interpretations" to enable the reciprocal "translation" between studies, as well as the use of an inductive process of constant comparison to develop such themes [27, 28]. Unlike grounded theory, the thematic synthesis employs primary studies as unit of analysis rather than the raw data from interview transcripts, field notes or memos [26].

The target primary studies for this paper comprised peer-reviewed academic publications, as well as journals that combine research and industry experience, such as "Harvard Business Review" or "Design Management Review". Some authors like Garousi et al. [30] have presented strong valid arguments advocating for the use of non-published, non-peer reviewed sources of information like consultancy whitepapers, industry reports or blog posts (the so-called "grey literature"), in order to give a stronger voice to empirical knowledge from software engineering practitioners. Such types of online contents were not included in this paper, to prevent any possible biases due to limited disclosure of project background details, as well as to avoid publications which had the commercial purpose of endorsing professional DT services.

The search query was executed in the online databases ScienceDirect and Google Scholar, using the base term "design thinking" in combination with various equivalent keywords for "enterprise" (i.e. "design thinking AND enterprise", "design thinking AND large organization" and "design thinking AND industry"). Priority was given to articles with the highest number of citations. The backward snowballing method was used to perform an iterative revision of the title, publication venue and authors of the most relevant articles referenced on each paper. Following the guidelines by Wohlin [31], the abstract of the papers was read first and then any other parts until a definitive decision could be taken to either include or exclude the paper. In addition to the backward snowballing of the reference lists, the Google Scholar profiles of the authors from the selected papers were reviewed to identify any other relevant articles.

The following exclusion criteria were applied: Papers published before 2007 were omitted to reflect only the recent contributions and empirical observations that are most relevant to the current state-of-the-art, influenced mainly by the work of consultants and field practitioners during the last decade. Papers that did not mention explicitly the combined term "design thinking", but only referred to design in a broader scope (e.g. "design practices" or "design principles") or in a different context (e.g. "design research", "user-centered design") were also excluded.

As suggested by Thomas and Harden [29], following a principle applied in other similar methods aimed at building grounded formal theory, the concepts in a thematic synthesis are to be derived from text grounded in the original context where it was constructed. Three review questions (abbreviated Q1, Q2 and Q3) were proposed as a final filter and primary method for grouping the articles, facilitating a comparative analysis of similarities and differences among those publications which aimed for the same purposes. While enabling a clear separation of "data-driven" themes (which are rather descriptive accounts of the primary studies), the review questions also provided the necessary scaffolding to develop a higher level "theory-driven" analysis, without detaching it entirely from the original source. The three review questions are:

1. What definitions of DT have emerged from academia or industry?
2. How have enterprises adopted or implemented DT during the last decade?
3. What are the main challenges of DT and how can enterprises overcome them?

The author skimmed through the most cited articles obtained and read the key sections such as title, abstract or conclusions, in order to ensure that all publications included in this study covered at least one of the three high-level goals presented by the review questions, either implicit or explicitly stated in text: (1) define the concept of DT according to scholars and practitioners, (2) collect empirical evidence about how DT is actually being adopted or implemented in enterprises, or (3) propose ways to overcome the challenges of DT in an enterprise context.

As a result of the search and filtering process described above, 16 publications were selected for the thematic synthesis, which involved reading them entirely to identify their recurrent concepts or themes. The analysis process consisted of three stages, sometimes overlapping: (1) Line-by-line coding, (2) grouping the codes into "data-driven" descriptive themes, and (3) developing higher-level "theory-driven" analytical themes across primary studies. The latter step represents the two main contributions of this paper: A proposed definition for DT and a future research agenda to reduce the gaps found in literature.

3 Research Focus and Recurring Themes of Primary Studies

Table 2 presents a summary of the themes found in the articles, organized according to their frequency of appearance. Such frequency should be not interpreted as the level of importance attributed to any given theme in comparison with the others, because the source publications differed in their scientific rigidity and choice of research methods. The themes were classified and grouped around specific "stages" in the process of understanding and implementing DT, positioning them closer to either thought or action, and serving as an indicator of the themes which are more likely to be found in conceptual discussions (i.e. Q1-type of publications), empirical studies (i.e. Q2-type of publications), or both (i.e. either one of the proposed questions). Additionally, each theme was broadly categorized by "type" as a benefit or challenge of DT, depending on whether the concept was described by the author and/or study participants in a positive or negative context, respectively.

Table 3 presents a summary of the articles reviewed and indicates with an "X" whether they contributed to answer Q1, Q2 or Q3 for this paper. The scientific rigidity of the publications was categorized into three levels (High, Medium, Low), depending on the clarity of the research methods employed and the use of academic references to support key concepts and findings. An additional distinction worth noting in relation to Q2 is that some authors based their conclusions on primary data collected for that

Table 2. Recurring concepts or themes in the selected publications

Theme	Publication(s)	Freq.	Stage of DT	Type
Underlying design process/methods	3, 5, 6, 12, 15	5	Understand	Benefit
Ambiguous or diverse meaning	4, 5, 6, 7, 10	5	Understand	Challenge
Can function alongside “traditional” methods	4, 9, 11, 13, 14	5	Implement	Benefit
Used to understand customers better	4, 8, 11, 12, 16	5	Implement	Benefit
Clash or mismatch with existing culture/processes	1, 2, 8, 12	4	Implement	Challenge
Advantages/benefits of tangible prototypes	7, 10, 13, 16	4	Implement	Benefit
No single process	4, 11, 14	3	Understand or Implement	Challenge or benefit
Acceptance of uncertainty and risk	7, 12, 14	3	Implement	Challenge
Cultural change to embed DT in organization	7, 13, 16	3	Implement	Challenge
Adaptation from design into management context	10, 12, 14	3	Understand	Challenge
DT team composition	12, 13, 16	3	Implement	Challenge
Inapplicability in certain contexts	7, 8	2	Implement	Challenge
Problem-solving beyond customer-facing product development	9, 14	2	Implement	Benefit
Relation between DT and IT development	11, 16	2	Understand and Implement	Challenge
Difficulty to communicate value	8	1	Implement	Challenge
Redefine the problem scope	12	1	Implement	Benefit

particular study, while others relied on indirect sources to explain how DT has been applied in large organizations. The latter cases are marked with an “I” instead of “X”.

4 Findings

The following sub-sections discuss the similarities and differences among the publications inside each group, in order to answer the three review questions proposed.

4.1 Q1: Definitions of Design Thinking in Academia and Industry

Various authors of the selected publications discussed about the many discourses and explanations for the term DT [4–7, 10]. The current understanding in management and

Table 3. Contribution of each article to the proposed review questions

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Scientific rigidity	H	H	M	L	H	H	L	M	L	M	H	M	L	H	M	M
Answers Q1	X	X	X	X	X	X	X			X	X	X		X	X	
Answers Q2	X	X		I		I	I	X	X		I	I	X	X		X
Answers Q3							X	X				X				

industrial context has been compared with the academic discussions unfolding inside the design research community decades before. Along this line, Kleinsmann et al. [6] and Liedtka [10] credit the first usages of the term to Archer [20] and Rowe [23], respectively. However, the author of the present paper notes that Archer did not refer explicitly to the term “Design Thinking” in his 1979 article [20], but did mention it in an earlier publication titled “Systematic method for designers” [19]. On the other hand, Rowe [23] certainly attempted to create a “generalized portrait of design thinking”, by identifying common problem-solving patterns in designers’ work, but his primary focus was to develop heuristics for architecture and urban planning.

The studies reviewed have frequently mentioned that DT has ambiguous or different meanings depending on the context in which it is applied [1, 11] and thus, it should not be condensed into a single definition [10]. For instance, Carlgren et al. [1] found that some practitioners described the concept using general terms related to user-centeredness and innovation, while others referred to it as a mindset, a set of principles, or as a combination of mindset and practical methods. It was often also explained using the attributes *“iterative”* and *“nonlinear”*. Similarly, Lindberg et al. [11] observed that DT was explained either as methodology or as a mindset, with a more homogeneous understanding at the level of its general goals and principles.

Design researchers have adopted a critical stance regarding the simplified explanations of DT used in managerial practice and other industries [15]. Regardless of the exact wording used, most definitions are built around the notion of translating the skills and knowledge of designers into everyday actions [5, 6, 15]. Empirical studies have revealed some shared basic notions of DT, even among diverse non-designer practitioners [1, 9]. Thus, the existence of an underlying set of design practices seems to prevail as the fundamental and most commonly understood characteristic of DT, despite the variety of areas where it is applied.

Both design and management research have privileged the word innovation as the fundamental concept that connects the various domain discourses on DT [6]. However, DT has been also proposed as a general-problem solving approach for more than just product or service innovation, as various case studies have shown [9, 14]. Therefore, innovation could be considered the main goal of DT in the context of customer-facing product and service design, or as a desirable outcome of DT when it concerns more general problem-solving situations.

4.2 Q2: Case Studies About Design Thinking Implementation

During the last decade, various case studies have explored how DT is understood and adopted by large companies. These studies have shown that DT can be applied with clear strategy and goals [13, 16], but also that its value is often not transmitted to stakeholders, who can perceive it as inefficient or as a waste of resources [8]. The seemingly contraposing findings raise attention about the generalizability of the theory provided by these studies. Some consistent results take the form of pre-requisites or conditions that organizations should fulfill, like the commitment and continuous push from several key stakeholders. Another consistent finding is that the success of DT projects and initiatives depends significantly on the mindset of the managers, who need to take less of a pragmatic approach (i.e. perceiving DT just as a toolset) and understand that its principles or ways of working should be embedded in the culture across the entire organization [14].

The selected publications have also mentioned that DT teams should be allowed sufficient freedom and flexibility to experiment, but at the same time they must be effective in communicating their ideas and proposals to get the trust from managers and other stakeholders. In this context, more "tangible" prototypes seem to help in demonstrating progress, gather internal buy-in and keep the relevant team discussions moving forward [7, 10, 13, 16]. An area that could be explored further through empirical research is to identify the specific dimensions that convey the most value to stakeholders, depending on the types of prototypes used. This could be theorized as advice for practitioners on how to build more effective (i.e. persuasive) prototypes.

Research findings also suggest that DT allows to understand better the customers' needs [4, 8, 11, 12, 16], although the benefits of this increased customer-centricity are not always clear. Furthermore, it has been shown that DT can be used in organizations not only for the development of innovative customer-facing products, but also to improve their own internal processes [14, 16]. These findings could be loosely related to the field of requirements engineering or systems analysis.

Although this paper was not intended to establish links between DT and software development practices, it is worth mentioning the scarcity of explicit mentions about IT in the context of DT. The primary studies often covered too many interrelated topics with a broader high-level view of DT as a problem-solving approach to innovation, but came short of explaining how it can be mapped into concrete aspects of IT and software development. A notable exception that shows an emerging branch of research in this field is the study by Lindberg et al. [11], which concluded that DT contributes to IT development, overcoming the limitations of traditional approaches (by allowing developers to give more attention to non-technical aspects) rather than fully replacing them. For related research about the potential impact of DT in IT development practices, the reader is referred to another theoretical discussion by Lindberg et al. [17] as well as a case study by Hildenbrand and Meyer [18] which were not part of this review. Nevertheless, the observed research gaps provide an opportunity to study in more detail the points of complementarity between DT and IT, with findings supported directly by field data.

4.3 Q3: Identifying and Overcoming the Challenges of Design Thinking

Existing literature has analyzed the challenges of DT implementation in enterprises and suggested some approaches to address them. The challenge most commonly mentioned when applying DT was the difficulty to handle the flexibility and unpredictability of DT [1, 2, 8, 12], attributes which may defy the existing structures, roles and responsibilities, particularly in large and strictly rigid companies. Two distinct perspectives to address this challenge have emerged from the articles reviewed: (1) The structure and culture of enterprises clashes inevitably with DT, or (2) DT can co-exist with existing structures, practices or methods until it is assimilated.

In the first scenario, the strategy to overcome the main challenge is to keep a balance between seemingly opposing or contradictory forces: The existing dominant cultures–characterized by a high standardization of processes with predictable outcomes and metrics–on one hand, and the seemingly "chaotic", iterative and overlapping process of DT on the other [2, 4, 8]. Under this perspective, Prud'homme van Reine [12] proposed a framework of nine dilemmas or "tensions" which reflect how enterprises with solid traditional cultures deal with the cultural conflicts brought by the DT practice. The assessment of these nine dilemmas inside organizations could work as a metric to evaluate more objectively the development and effects of DT. In other words, the DT-readiness of both companies and their managers can be estimated by observing the ability to keep these "opposing elements" in balance:

1. Analytic thinking *versus* intuitive and creative thinking
2. Product push *versus* user empathy
3. Focus on functional aspects *versus* focus on aesthetics and emotional aspects
4. Closed *versus* open approaches to innovation
5. Innovation as a structured process *versus* "bricolage" (i.e. developing systematically the "one best" solution *versus* finding the "best possible" solution with the resources available at a given moment)
6. Linear thinking *versus* non-linear, iterative and overlapping steps (i.e. planning *versus* experimenting)
7. Individual creativity *versus* group innovation
8. Egalitarianism *versus* hierarchical leadership (i.e. stimulating participation *versus* forcing guidance toward a decision)
9. Short-term *versus* long-term approach to innovation (i.e. immediate gains *versus* exploration of possibilities).

The second type of strategy found in the primary studies does not describe an opposition scenario between DT and the existing organizational structures, practices or methods. Instead, the strategy is to embed or blend DT into the entire organization, generating a new culture that gives more importance to design. For this purpose, a clear definition of roles inside DT teams based on task distribution [16] or an analysis of DT practitioner profiles based on their goals and vision about innovation [6] may facilitate the selection of adequate people to lead the process of cultural change.

Along the same line of thought, Sato et al. [13] sustain that DT gives flexibility to augment the conventional business thinking and tailor existing models to specific situations. In their publication, the authors combined DT with four different models for organizational change and development (OC&D). This was achieved by mapping the key concepts and principles of each OC&D model into the context of a broader DT "process cycle", composed by the following elements which are continuously reviewed and refined in no particular order:

- *Intent:* Initial stance on the problem by defining the goal, target customer, expected outcome and metrics
- *Discoveries:* Insights revealing "things as they are" and the reasons why
- *Frameworks:* Hypotheses about patterns and relationships between customer needs, organizational capabilities and business requirements
- *Principles:* Prescriptive guidelines or imperatives derived from the frameworks
- *Solutions:* Ideas that best fulfill the principles and become tangible with prototypes.

Some of the findings in the study by Schmiedgen et al. [14] also adhere to the perspective of an orderly or conscious transition toward the adoption of DT, describing four "archetypical places" where DT can be found inside organizations. This idea comes from a previous work by Junginger [24], who presented a bubble-shaped graphic visualization of these stages as a tool to determine where and how is the organization utilizing DT at any given moment. Moving beyond the assessment of individual discrete points in time and observing the entire cultural change process, these "archetypical places" can be also understood as "assimilation stages" to gradually blend DT into the organizational culture:

1. *Periphery:* External resource booked on demand, such as from external consultants
2. *Somewhere:* Applied separately within some internal areas or departments
3. *Core:* Direct contact with the leadership and used for strategic decision-making
4. *Intrinsic:* Established practice and mindset spread throughout the culture of the entire organization, used beyond product and service innovation.

The similar views regarding DT adoption suggest that the process cycle model of Sato et al. [13] could be applied in combination with the four archetypes described by Schmiedgen et al. [14]. On one hand, the first model would allow to monitor the five key components that should be continuously refined while blending DT into the organization culture (i.e. process inputs and outputs). At the same time, the latter can be used as a reference to evaluate how much DT has already pervaded into the organization (i.e. current state of the adoption process).

5 Study Limitations

As a secondary study which relies on the selection and analysis of existing literature, the present paper has various limitations. First and foremost, the data collection and analysis process was carried out by a single researcher. Even though this paper merely analyzed previous studies performed by various authors, coming from different research traditions and with different perspectives on DT, the possibility of relevant theme omissions and accidental introduction of researcher biases remains. By providing a clear identification of the aims and methodological quality of each primary study, the readers of this paper become able to judge for themselves whether the actual contents of the referenced articles match the descriptions provided in this review. However, the validity of the findings presented could be further improved with different triangulation methods. For instance, by following up with complementary studies which include new primary data (e.g. collected through scholar and practitioner interviews), or by examining again the same articles presented in this paper but from the perspective of other researchers, in order to find any similarities or differences in the conclusions. Business book sections and "grey literature" have been excluded from this study to avoid biases, but may certainly constitute valuable sources of practitioner knowledge. Future studies which include such publications should also allow comparisons with data obtained from inquisitive or observational techniques, as it is difficult to fully cover the enterprise perspective by literature reviews alone.

Another limitation from this study is related to the unequal distribution of papers among the three focus areas mapped into the review questions. On one hand, many of the selected publications aimed to answer Q1 and provided sufficient material to compare the multiple definitions given to DT. On the other hand, it was more difficult to identify relevant concepts related to Q2 and Q3. The empirical studies about DT implementation provided too detailed accounts of case-specific challenges, mostly related to the difficulty to grasp the meaning of DT, as well as the results achieved in each particular case context. For the reasons mentioned, Q1, Q2 and Q3 could not be covered with the same thoroughness during this study, thus providing a better explanation of the academic perspective on DT than the enterprise one. Given the exploratory nature and limited scope of this paper (to propose a definition and research agenda for DT) this may not be considered a significant issue. Nevertheless, it is the opinion of this author that further cross-case empirical studies will be needed in the future to provide stronger supporting arguments for Q2 and Q3.

Finally, as mentioned in the introductory section, the word "enterprise" was constrained in this paper to "large commercial business organizations", in order to facilitate the interpretation and analysis of previous studies under a relatively similar context. A recommendation for future research on this same topic is to clarify the scope and tailor it to certain specific types of organizations (e.g. public sector entities, non-governmental organizations, start-ups, etc.), accounting for their complexity, size or mission, among other types of relevant dimensions which could allow to make the findings more comparable between studies.

6 Discussion and Conclusions

DT is a concept that emerged from the discussions in design research and acquired fuzzy and diverse meanings under the participation of non-designer practitioners in industry. The diversity of discourses and explanations for DT has been addressed by various authors but there is still no agreement on a single definition, or even on whether such definition should be agreed upon. On one hand, scholars from the design tradition tend to be more critic about the pragmatic use of the term and defend the value, training and expertise of the underlying design practices. On the other hand, case studies and interviews have shown a basic level of shared understanding among designer and non-designer practitioners, which suggests that a common definition can indeed be built, starting from the essential aspects more often associated to DT.

It is therefore the opinion of this paper's author that any future discussions about DT should at least have some basic generally agreed definition that allows to move the academic discussion forward, improving at the same time the level of understanding about DT in industry. Based on all the primary studies analyzed for this paper and especially building upon earlier findings by Schmiedgen et al. [14], the following definition of DT is proposed:

> *"Design thinking is an approach generally applied to customer-facing product and service innovation, which can be also used as a general problem-solving approach. It relies on a heterogeneous set of design practices and methods, which depending on the background of each practitioner may be understood more as a toolset (i.e. pragmatic perspective) or as a mindset (i.e. idealistic perspective). Rather than following a structured linear process, it is often characterized to be iterative and user-centered, attributes which tend to facilitate the rapid experimentation of ideas through prototypes."*

The primary case studies about the adoption of DT inside large companies showed both positive and negative results, putting into doubt whether their findings are generalizable or specific to the organizations where the research took place. There seems to be more clarity in terms of the pre-requisites for a successful implementation of DT, assigning significant responsibility to the change of mindset in the managers. The author of this paper considers that more cross-case studies are needed in the future using the same set of research questions to compare any relevant themes or dimensions of DT in different organizational contexts.

As a major challenge of DT in large organizations, the reviewed publications pointed out consistently the difficulty to handle the flexibility and unpredictability of DT, in contraposition with the existing structures, roles and responsibilities. This challenge can be understood in two ways: (1) The structure and culture of large organizations naturally clashes with DT, or (2) DT can co-exist with existing structures, practices or methods until it gets assimilated. In the first case, the strategy is to keep a balance between permanently opposing forces. In the second scenario, the strategy is to blend DT into the entire organization to generate a design-centric culture. There is an interesting opportunity to test in future case studies the strategies by Prud'homme van Reine [12] and Sato et al. [13] for the first and second scenarios, respectively.

In terms of the broader context where future research about DT could be positioned, a loose connection has been observed between requirements engineering and the increased customer-centricity of DT. Despite the relative broadness of the search keywords and primary study contents, it was a challenging task to find in the selected publications concrete links between DT and other well-established practices in software engineering. DT was mostly described as a problem-solving or innovation approach from a high-level perspective, without establishing theoretical connections with traditional software development practices. Thus, future research about the complementarity between IT and DT should be conducted with more precise data collection methods and objectives, so the empirical evidence can be traced back into more specific disciplines like requirements engineering or software testing.

The concept of DT carries multiple connotations and a high-level of abstraction. If DT is considered a fuzzy topic and wicked problem on its own right, the lessons learned from DT case studies may be extended to the process of researching DT itself. The concept should be brought down to a level of understanding that allows both academics and practitioners to speak a common language with each other. This is a challenging task, but hopefully also a strong motivation for researchers to start examining the potential impact of DT, or any other "vaguely defined" – yet potentially relevant – emerging categories that could be overlapping with their areas of expertise.

References

1. Carlgren, L., Elmquist, M., Rauth, I.: Exploring the use of design thinking in large organizations: towards a research agenda. Swed. Des. Res. J. **11**(1), 55–63 (2014)
2. Carlgren, L., Elmquist, M., Rauth, I.: The challenges of using design thinking in industry-experiences from five large firms. Creat. Innov. Manag. **25**(3), 344–362 (2016)
3. Dorst, K.: The core of 'design thinking' and its application. Des. Stud. **32**, 521–532 (2011)
4. Drews, C.: Unleashing the full potential of design thinking as a business method. Des. Manag. Rev. **20**(3), 38–44 (2009)
5. Johansson-Sköldberg, U., Woodila, J., Çetinkaya, M.: Design thinking: past, present and possible futures. Creat. Innov. Manag. **22**(2), 121–146 (2013)
6. Kleinsmann, M., Valkenburg, R., Sluijs, J.: Capturing the value of design thinking in different innovation practices. Int. J. Des. **11**(2), 25–40 (2017)
7. Kolko, J.: Design thinking comes of age. Harv. Bus. Rev. **93**(9), 66–71 (2015)
8. Le Glatin, M., Le Masson, P., Weil, B.: Measuring the generative power of an organisational routine with design theories: the case of design thinking in a large firm, Potsdam, Germany (2016)
9. Liedtka, J.: Innovative ways companies are using design thinking. Strategy Leadersh. **42**(2), 40–45 (2014)
10. Liedtka, J.: Perspective: linking design thinking with innovation outcomes through cognitive bias reduction. J. Prod. Innov. Manag. **32**(6), 925–938 (2015)
11. Lindberg, T., Köppen, E., Rauth, I., Meinel, C.: On the perception, adoption and implementation of design thinking in the IT industry. In: Plattner, H., Meinel, C., Leifer, L. (eds.) Design Thinking Research. UNDINNO, pp. 229–240. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-21643-5_13
12. Prud'homme van Reine, P.: The culture of design thinking for innovation. J. Innov. Manag. **2**, 56–80 (2017)
13. Sato, S., Lucente, S., Meyer, D., Mrazek, D.: Design thinking to make organization change and development more responsive. Des. Manag. Rev. **21**(2), 44–52 (2010)
14. Schmiedgen, J., Rhinow, H., Köppen, E., Meinel, C.: Parts Without a Whole? - The Current State of Design Thinking Practice in Organizations. Hasso-Plattner-Institut für Softwaresystemtechnik an der Universität Potsdam, Potsdam (2015). Study Report No. 97
15. Tonkinwise, C.: A taste for practices: unrepressing style in design thinking. Des. Stud. **32**(6), 533–545 (2011)
16. Vetterli, C., et al.: How Deutsche Bank's IT division used design thinking to achieve customer proximity. MIS Q. Exec. **15**(1), 37–53 (2016)
17. Lindberg, T., Meinel, C., Wagner, R.: Design thinking: a fruitful concept for IT development? In: Plattner, H., Meinel, C., Leifer, L. (eds.) Design Thinking: Understand – Improve – Apply. Understanding Innovation, pp. 3–18. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-13757-0_1

18. Hildenbrand, T., Meyer, J.: Intertwining lean and design thinking: software product development from empathy to shipment. In: Maedche, A., Botzenhardt, A., Neer, L. (eds.) Software for People. MANAGPROF, pp. 217–237. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31371-4_13
19. Archer, B.: Systematic Method for Designers. HM Stationery Office, Richmond (1965)
20. Archer, B.: Design as a discipline. Des. Stud. **1**(1), 17–20 (1979)
21. Kelly, T., Kelley, D.: Creative Confidence: Unleashing the Creative Potential Within Us All. Crown Business, New York (2013)
22. Brown, T.: Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation. HarperBusiness, New York City (2009)
23. Rowe, P.G.: Design Thinking. MIT Press, Cambridge (1987)
24. Junginger, S.: Parts and wholes: places of design thinking in organizational life. Strateg. Des. J. **2**(9), 23–29 (2009)
25. Welch, C., Piekkari, R.: How should we (not) judge the ‘quality’ of qualitative research? A re-assessment of current evaluative criteria in International Business. J. World Bus. **52**(5), 714–725 (2017)
26. Cruzes, D., Dybå, T.: Recommended steps for thematic synthesis in software engineering. In: 2011 International Symposium on Empirical Software Engineering and Measurement, ESEM, pp. 275–284. IEEE (2011). https://doi.org/10.1109/esem.2011.36
27. Cruzes, D., Dybå, T.: Research synthesis in software engineering: a tertiary study. Inf. Softw. Technol. **53**(5), 440–455 (2011). https://doi.org/10.1016/j.infsof.2011.01.004
28. Barnett-Page, E., Thomas, J.: Methods for the synthesis of qualitative research: a critical review. BMC Med. Res. Methodol. **9**(1), 59 (2009). https://doi.org/10.1186/1471-2288-9-59
29. Thomas, J., Harden, A.: Methods for the thematic synthesis of qualitative research in systematic reviews. BMC Med. Res. Methodol. **8**, 45 (2008). https://doi.org/10.1186/1471-2288-8-45
30. Garousi, V., Felderer, M., Mäntylä, M.V.: The need for multivocal literature reviews in software engineering: complementing systematic literature reviews with grey literature. In: Proceedings of the 20th International Conference on Evaluation and Assessment in Software Engineering, EASE, p. 26. ACM (2016)
31. Wohlin, C.: Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering, EASE, p. 38. ACM (2014)
32. Definition of Enterprise by Merriam-Webster. https://www.merriam-webster.com/dictionary/enterprise. Accessed 19 Apr 2018
33. Enterprise Meaning in the Cambridge English Dictionary. https://dictionary.cambridge.org/dictionary/english/enterprise. Accessed 19 Apr 2018
34. Enterprise definition and meaning—Collins English Dictionary. https://www.collinsdictionary.com/dictionary/english/enterprise. Accessed 19 Apr 2018
35. “design thinking” – Explore – Google Trends search. https://trends.google.com/trends/explore?date=2007-01-01%202017-12-31&q=%22design%20thinking%22. Accessed 06 Apr 2018

From Strategy to Process Improvement Portfolios and Value Realization

A Digital Approach to the Discipline of Business Process Management

Mathias Kirchmer[1,2(✉)], Peter Franz[1], and Rakesh Gusain[1]

[1] BPM-D, West Chester, USA
{Mathias.Kirchmer, Peter.Franz, Rakesh.Gusain}@bpm-d.com
[2] University of Pennsylvania, Philadelphia, USA

Abstract. There is a growing agreement in academia as well as in practice that business process management (BPM) has become the management discipline for the systematic execution of business strategy. As any other management discipline, it is implemented through a business process itself, the process of process management. Many components of this process of process management are supported through digital technologies. However, there are critical gaps in the digitalization of the BPM process and existing digital components are often not sufficiently integrated. As a result, organizations do not realize the full potential of a BPM-Discipline. This issue is addressed through the BPM-D Application. A key digitalization gap is the transfer of the strategy of an organization into the appropriate portfolio of process improvement initiatives. For top executives this means that they have to be able to take fast well-informed decisions based on strategic priorities. The operational managers need to drive aligned execution activities. This paper discusses an approach towards a process-driven and value-focused portfolio management to create project portfolios that reflect the priorities of the business strategy, combined with a systematic approach to value realization, during a project and especially after project conclusion. As a result, top executive and operational management views are aligned towards the desired outcome. This paper shows how the presented approach is enabled through the BPM-D Application which represents another step towards digital BPM.

Keywords: BPM · BPM application · BPM-Discipline
Business process management · Digitalization · Process of process management
Execution · Strategy · Process design · Process improvement
Project portfolio management · Strategy execution · Business processes
Enterprise architecture · Process modelling · Software development
Value-driven BPM · Value realization

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 32–55, 2018.
https://doi.org/10.1007/978-3-319-94214-8_3

1 Introduction

Business Process Management (BPM) is increasingly seen as a management discipline that has significant impact on an organization (von Rosing et al. 2015). It provides value by transforming strategy into people and technology-based execution – at pace with certainty (Franz and Kirchmer 2012). BPM plays a key role in realizing the full potential of digitalization initiatives (Kirchmer et al. 2016; Kirchmer et al. 2017). The discipline of process management enables ongoing strategy execution and digitalization in a volatile business environment.

The BPM-Discipline is implemented through a process of its own, the process of process management (PoPM). The increasing importance of the BPM-Discipline for the success of an organization requires an appropriate performance improvement of the PoPM. First progress in this area has been made using the appropriate design of the PoPM (Kirchmer 2015). In order to achieve the next performance level, we apply digitalization systematically to the PoPM itself (Kirchmer et al. 2017). Result is the next generation of "digital BPM".

While we focused in the last paper on defining the requirements for the digitalization of the PoPM, we discuss here the resulting digital tool and its application. Hence, we move the focus from the build-time to the run-time of digital BPM.

2 BPM Digitalization and Automation Gaps

Since the discipline of BPM is implemented through a business process itself, this process, the PoPM, needs to be examined to identify where appropriate digitalization and automation is missing to move it to the next performance level and enhance digital BPM. This is done by using a reference model for the PoPM to identify the digitalization and automation status and resulting gaps. Those digitalization and automation gaps provide the basis for the requirements definition for new digital tools.

2.1 The Process of Process Management

BPM operationalizes strategy so that it can be executed through the appropriate combination of people and technology, at pace with certainty (Franz and Kirchmer 2012). This is shown in the BPM-D® Framework in Fig. 1. This framework summaries key aspects of the definition of BPM. The business strategy is executed through an appropriate management of the process lifecycle from design, implementation, execution to control of the process – considering people and digital technologies in each phase. The key aspect of this definition of BPM is the focus on strategy and its execution. While other definitions recognize this aspect, it is in most of the cases not as centrally positioned as in the BPM-D Framework.

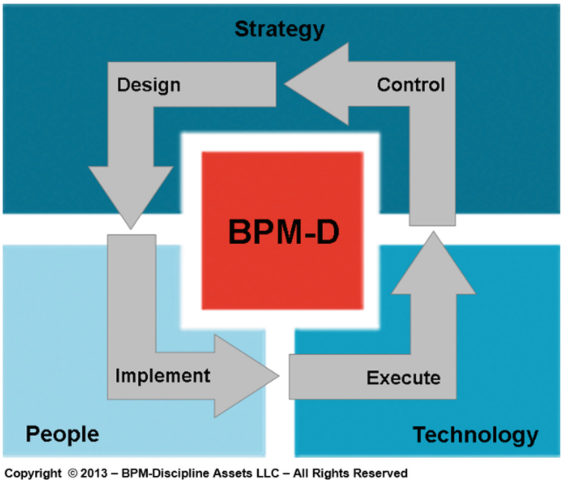


Fig. 1. BPM-D® framework

The BPM-Discipline is implemented, just as any other management discipline: through the appropriate business processes: the "process of process management" (PoPM) (Franz and Kirchmer 2012). This PoPM consists of project-related sub-processes, focused on improving the organization and realizing the targeted value, and asset-related processes, enabling efficient and effective improvements. In both groups we can distinguish planning and realization related sub-processes. The overview of a reference model for the PoPM is shown in Fig. 2 (Kirchmer 2015; Kirchmer et al. 2017).

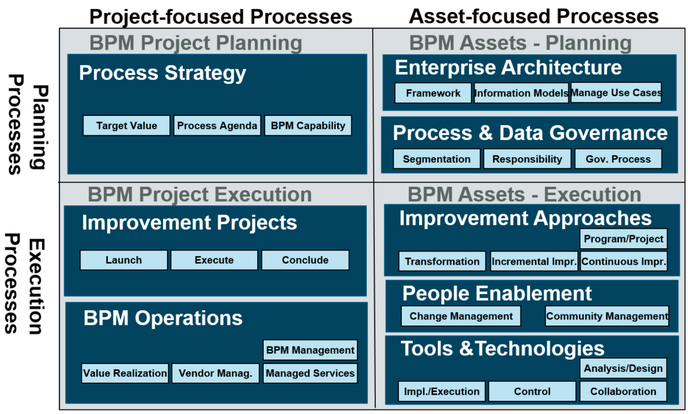


Fig. 2. The BPM-D process framework – reference model for the process of process management

To operationalize this PoPM reference model we have described this business process from all relevant views (Scheer 1998): organization, functions, data, deliverables and control view (Kirchmer 2015). In over 50 business transformation and improvement initiatives we have proven that this PoPM reference model delivers significant value - adjusted and applied in the specific business context of an organization (Kirchmer 2016). It is sufficiently complete and consistent. Key benefits achieved through the application of the PoPM reference model are:

- Reduced risk through re-use of successful practices
- Reduced design time for the PoPM and the organization around it
- Transparent knowledge transfer
- Basis for the ongoing management of the BPM-Discipline
- Basis for ongoing assessments and expansion of the BPM-Discipline through new initiatives
- Information and communication about BPM is more tangible and therefore simplified.

Modelling the PoPM from different views helps to understand relevant capabilities for the BPM-Discipline as well as the business value they provide. It helps to identify how digital technologies can support this management discipline and how this technology supports the current maturity.

The high importance of the BPM-Discipline for strategy execution and enterprise-wide digitalization requires and justifies an accelerated improvement of the PoPM and its application to specific organizations. This can be achieved by moving the digitalizing of the PoPM itself to the next level. Targeted digitalization and the resulting innovation are important enablers for the "Chief Process Officer" (Kirchmer and Franz 2014a) to guide the journey of the BPM-Discipline to deliver best value for an organization. The resulting digital tools make process management more tangible and with that easier to be rolled out. We expand this digitalization approach from using individual, little integrated components of the PoPM to an integrated end-to-end approach across the entire BPM-Discipline.

2.2 Digitalization and Automation Status

In order to identify the status of digitalization and automation of the PoPM we analyzed the different sub-processes of the BPM-D Process Framework. The input from over 200 process initiatives has shown that there are very different maturity levels of digitalization in the various PoPM sub-processes (Kirchmer et al. 2017). The digitalization of the PoPM has focused on the automation of operational processes, including new developments like Robotic Process Automation (RPA) or Blockchain (Scheer 2017; Kirchmer 2017a, 2017b). This has had significant impact on the value the PoPM can provide and the way it approaches process improvement initiatives.

The management of related projects is also well supported through project management systems and related tools to support available process improvement approaches. The PoPM can leverage general solutions in this field, hence project management systems. This is also the case for most of the general BPM Operations for the ongoing management of the PoPM.

A special area of the BPM Operations is the value realization after project conclusion. Existing analytics, process mining and monitoring tools support different aspects of that sub-process. (Scheer 2017; Kirchmer 2017b). However, they are not sufficient to ensure the value realization fully.

Modelling and repository tools as well as specialized enterprise architecture tools cover the Enterprise Architecture related processes. Digital technologies have become mature in this area (Kirchmer 2017b; Franz and Kirchmer 2012). These tools also support the management of process-oriented digital tools and technologies. Some aspects of the people enablement, especially the training and capability building, are also supported through existing digital systems. Examples are e-learning tools and related computer-based training environments.

2.3 Digitalization and Automation Gaps

The development of a strategy-based process management agenda is one important area that is not well supported. An organization only competes with about 15–20% of its processes (Franz and Kirchmer 2012). All the others are commodity processes that do not really impact the competitive positioning if performed at an industry average level.

The commodity processes can be improved, when underperforming, through straight forward initiatives leveraging existing common practices. It is, however, very important for an organization to know its high impact processes, align the process management capabilities with those and define a BPM roadmap consistent with these strategic priorities (Kirchmer and Franz 2014b). These lead to an corresponding process-oriented project portfolio management. The systematic support of the development of a value-driven process agenda is crucial for a successful BPM-Discipline and should be updated with every major adjustment of strategy or market. In our research we have not identified any existing digital tools focused on supporting this part of the PoPM sufficiently, hence this should be part of a new more holistic digitalization approach.

The value-realization after the project and the related process and data governance are not sufficiently enabled through digital solutions either. Existing tools support different functions of those areas, but not the overall control flow and the management of those controls. The enforcement of process standards or the support of collaboration models of different governance bodies is not sufficiently covered though current digitalization approaches. Most activities in this area are still carried out manually with negative efficiency and quality effects or through general workflow approaches, not distinguishing between the operational process and the required governance processes.

Our analysis has shown that the whole "people dimension" of process management is also not provided appropriate digital support in many BPM initiatives. In most process-led transformation and improvement approaches the challenges are less concerning the used technology components, but rather the people involved in the initiative (Spanyi 2003). Since only few processes can be fully automated, people and their skills are often the bottleneck. While there is good progress made with digitally enabled change management approaches (Ewenstein et al. 2015), such as the use of eLearning or different communication tools, the active management of process communities and their integration with change management is still not sufficiently supported. Hence, this is another area for an improved digitalization and automation of the PoPM. Figure 3 illustrates the main gaps of the current digitalization and automation of the PoPM.

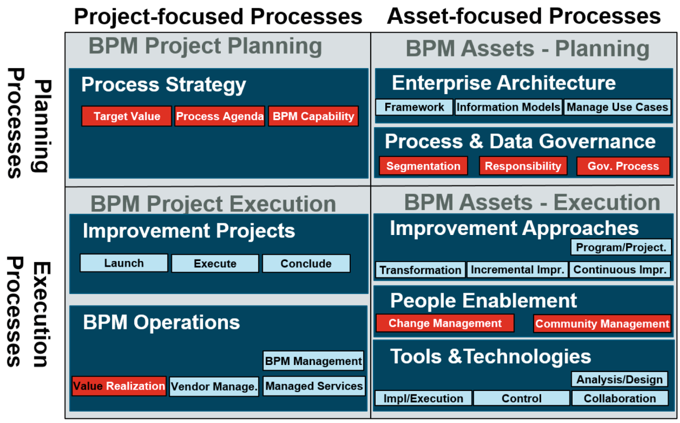


Fig. 3. Digitalization and automation gaps of the PoPM

We also detected a lack of integration between the different existing digital support approaches. While there exist selected integration examples, such as the connection of process modelling and execution tools, there is no overarching integration across the entire PoPM and the underlying digital technologies. A comprehensive digitalization of the PoPM needs to address this topic and deliver the right degree of integration enabling best performance of the overall PoPM. Only this integrated approach leads at the end to digital BPM and delivers the expected value.

3 Process-Oriented Project Portfolio Management and Value Realization

As starting points for the further digitalization of the PoPM we have chosen process-oriented project portfolio management, reflecting the prioritization approach and the BPM agenda, as well as the value realization after project conclusion.

The goal of the process-oriented project portfolio management is to set strategy-based priorities and focus initiatives on high impact low maturity processes to achieve best outcome for the targeted strategy execution. This integrates already on a business level the discipline of process management and the project management approach. BPM identifies and defines necessary improvement projects and manages the resulting project portfolios. Project management executes those portfolios and provides required feedback information. This is visualized in Fig. 4.

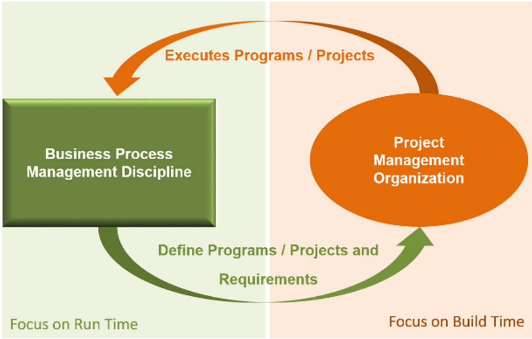


Fig. 4. Process-oriented project portfolio management: integrated approach to business process management and project management

The value-realization after project conclusion is essential to realize the strategy-related effects which often takes longer than the project duration. In this phase there is an integrated approach between the process design with the resulting process models and the overlaying process governance is required. This is visualized in Fig. 5. In traditional organizations this value realization is a challenge since there is often no governance approach after the projects has concluded. Process governance, integrated with process management resolves this situation.

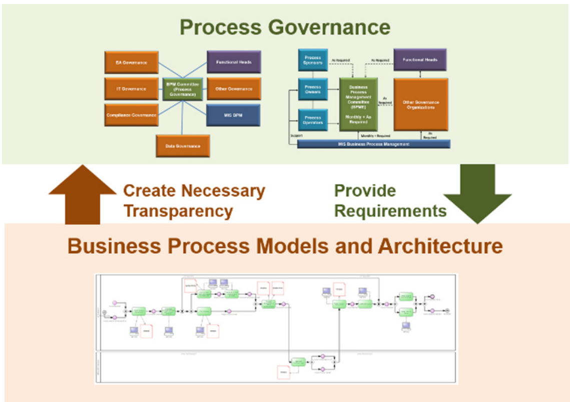


Fig. 5. Value realization based on integrated process design and process governance

Improving process-oriented project portfolio management and the related prioritization in combination with the value realization also addresses major challenges of digital transformation initiatives (Kirchmer et al. 2016). It helps to identify where to best start digitalization, supports the timely decision making, and supports business and IT alignment through common priorities. The focus of projects on specific process-areas

Fig. 6. Key challenges of digital transformations

also simplifies building up appropriate capabilities and the targeted approach delivers quick value, useful to overcome resistance to change. The key challenges are shown in Fig. 6.

In the underlying research a clear correlation between process management maturity and value through digitalization was identified. Process-oriented project portfolio management and integrated value relation simplify the systematic realization of this BPM maturity.

3.1 Defining and Managing Process-Based Project Portfolios

Process-oriented project portfolios are defined and managed based on the identification of high impact processes, related process management capabilities and the required work packages necessary to improve the processes, using the appropriate process management capabilities (Franz and Kirchmer 2012; Kirchmer and Franz 2014b). This requires 5 key steps:

- Identify high impact processes
- Identify required BPM capabilities
- Define resulting value packages
- Assign projects to value packages
- Prioritize projects
- Track value-realization.

To identify high impact processes the strategy of an organization is converted into a value-driver tree. Then the impact of each process on each of the value-drivers is calculated. The top 15–20% processes are generally considered the high impact processes (Franz and Kirchmer 2012). This process segmentation is best done on a level 3 of detail, that means that an organization is described through 150–200 processes. The results of the segmentation can then be summarized and communicated in a heat map based on the operating model description of the organization. In most practical situations there are process areas that are in-between high impact and commodity. In a second step they can be included in one or the other group based on overall management considerations. An example of such a heat map is shown in Fig. 7.

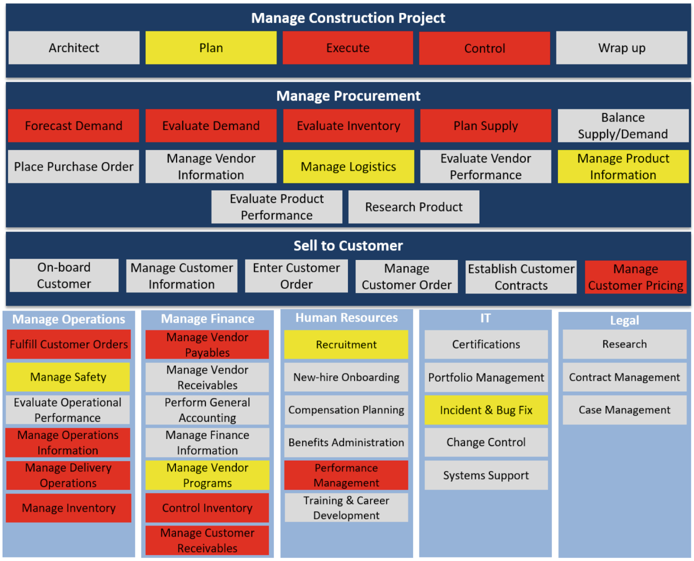


Fig. 7. Heat map identifying areas with most high impact processes.

For the high impact processes the maturity level compared to other organization of the same industry is evaluated. High impact low maturity processes are key targets for improvements and the resulting programs and projects. These processes need to get improved in order to execute the organization's strategy.

A focused BPM maturity assessment is used to identify the required process management capabilities and compares them to existing capabilities to identify the gaps (Franz and Kirchmer 2012). The capability assessment is structured based on the PoPM and the different views on it. The results can again be displayed as a heat map, using the BPM-D Process Framework as basis. Important BPM capabilities with a low maturity level need to be addressed in enabling projects. This ensures to have the capabilities developed, necessary to fix operational high impact low maturity business processes. The portfolio of necessary initiatives gets extended appropriately. Figure 8 shows an example for such a BPM capability heat map.

High impact processes and BPM capability gaps form the foundation for the definition of value packages. A value package defines the improvement activities, involved resources, expected outcomes and a high-level business case for an initiative. It should also show how it relates to the identified key value-drivers.

The execution of a value-package is done through one or several projects and a project can address one or several value packages. The impact of a value package on the execution of the business strategy is calculated by combining the impact on business processes and their impact on the value-drivers representing the strategy. This allows a strategy-based prioritization of value-packages and related projects. In Fig. 9 the example of a value package is shown.

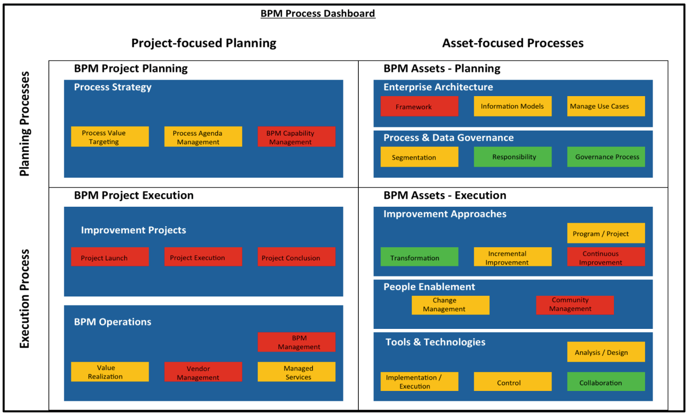


Fig. 8. Heat map showing BPM capability gaps

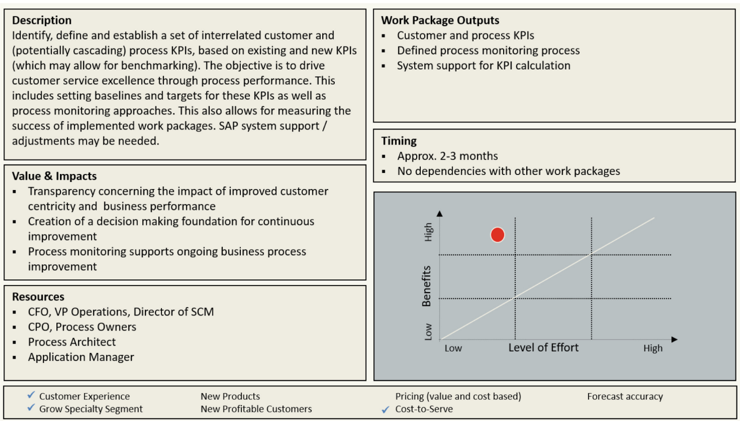


Fig. 9. Value package – example

The realization projects are prioritized based on the impact of the related value packages. This is simple if one project executes one or several complete value packages. In the case that value packages are split between projects, the identification of the project impact may require a management decision. We have not yet identified a more standardized approach for that. However, practice has shown that this only happens very rarely, hence it is not worth to examine this case further at the current point of time. However, we will keep on observing this situation in practice and address the point in the future if necessary.

The project portfolio is managed based on those priorities and the effort it takes to realize a project with the related value packages. High cost of a high impact project may force a delayed start due to a lack of resources. A simple project prioritization is visualized in Fig. 10. A changing business strategy or changing market conditions

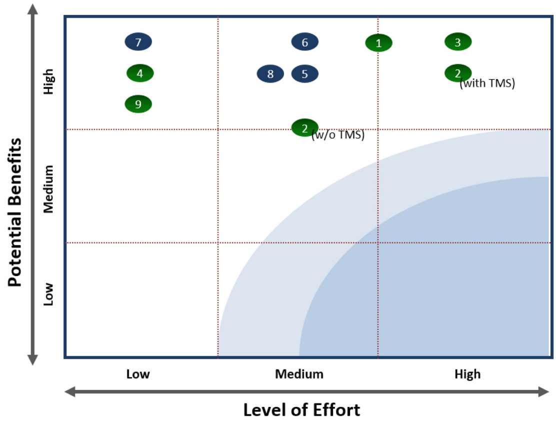


Fig. 10. Project prioritization

affect the high impact processes and the impact of the related value packages and projects. Hence, the project portfolio is continuously managed towards best execution of the current strategy in the current market conditions.

This process-oriented project portfolio management enables a systematic strategy execution, leveraging the discipline of process management. The continuous change and adjustment of portfolios and project prioritization emphasizes the need for digital support.

Once the portfolio is defined the value-delivery status of each project is tracked so that at the end of a project the ongoing value-realization has a clear starting point. It manages the resulting new processes towards the final target outcome.

3.2 Realizing the Value After Project Conclusion

Once a process improvement project is finished, in most of the situations the targeted value is not fully achieved. For example, if the goal was to reduce stock based on a better forecasting and planning approach, it will take time after implementation of this approach until the results are visible. During that time, it is crucial to control the process carefully and keep it on track. Depending on the achieved results over time either the goals are reached, need to be adjusted or additional improvement initiatives launched. Even if the goals of an improvement projects are achieved right after the project or later, a systematic process control is required to keep the new process on track.

Tools like process monitoring and mining deliver information about the project impacts, but they need to be embedded in a systematic process control to drive the value realization. Responsibilities and accountabilities need to be defined and managed appropriately. For this process control three key steps are required:

- Definition of the controls
- Execution of the controls
- Reporting of the control results.

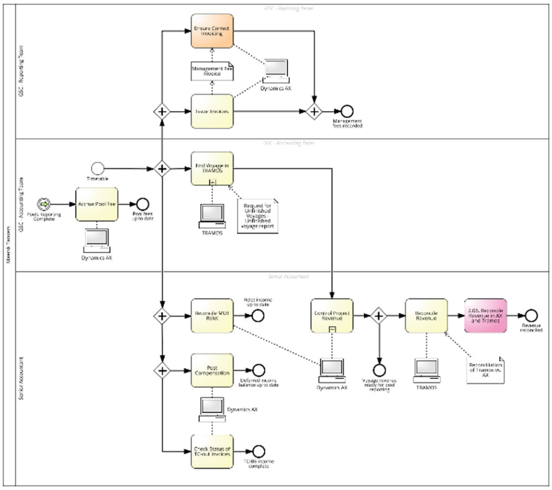


Fig. 11. Process model with integrated process controls

The controls are defined as part of the process design and they are included in a process model. The controls basically represent components of the process governance. Figure 11 shows the example of a process model in BPMN notation (Shapiro et al. 2012), including the relevant control steps. This approach ensures the right controls at the right time during the process execution. It aligns the operational process with process governance. Details of the control activities can be defined in appropriate standard operating procedures (SOPs).

The execution of those controls needs to be triggered during the running business process. The controls are documented to document the evolution of the control results as well as control responsibilities and accountabilities. The documentation should also include progress in the value-realization, for example linked to specific KPIs. In many cases the reports also need to fulfil compliance requirements so that they can be used to obtain appropriate certifications.

4 Digital Approach to Process-Oriented Project Portfolio Management

In line with these objectives, work has progressed on the design and implementation of an integrated BPM-D Application that aims to properly support and digitalize the Process of Process Management (PoPM). The approach, initial implementation and early pilots demonstrate considerable progress regarding the defined objectives.

In order to close the identified digitalization gaps of the PoPM, the development of the BPM-D Application has been launched (Kirchmer et al. 2017). This is designed as a cloud-based solution that allows on one hand the integration of existing process management software tools and on the other hand delivers the functionality to close the digitalization gaps. Key models and planned scope are shown in Fig. 12.

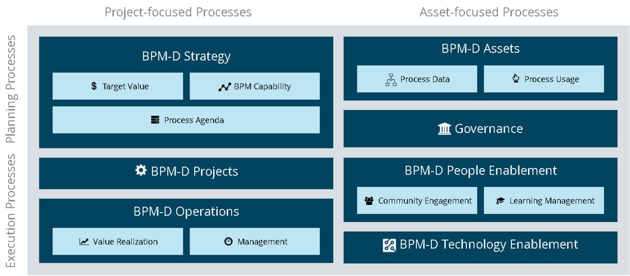


Fig. 12. Modules of the BPM-D application

The BPM-D Strategy modules support the process-oriented project portfolio management:

- Targeting value
- BPM Capability
- Process Agenda.

Related process management master data, such as the process hierarchy or value-drivers, are part of the governance module. This enables the use across all components of the application. The architecture of the BPM-D application is illustrated in Fig. 13. It explains how this digital component enables the integration of the portfolio management into the larger BPM-Discipline activities and their digital support.

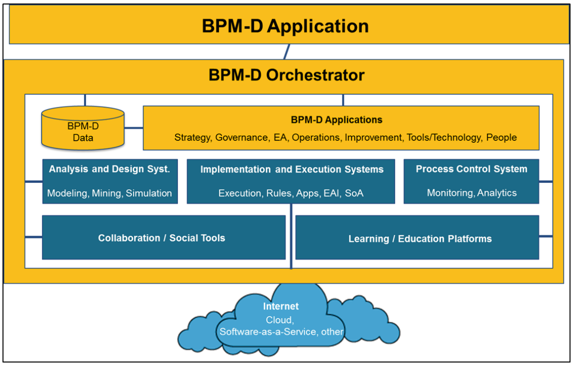


Fig. 13. BPM-D application architecture

4.1 Identifying High Impact Processes and Their Maturity Level

A high impact process is a business process that has significant influence on the execution of the strategy of an organization. Therefore, the BPM-D application allows the capturing of a short summary of the business strategy as basis for the evaluation.

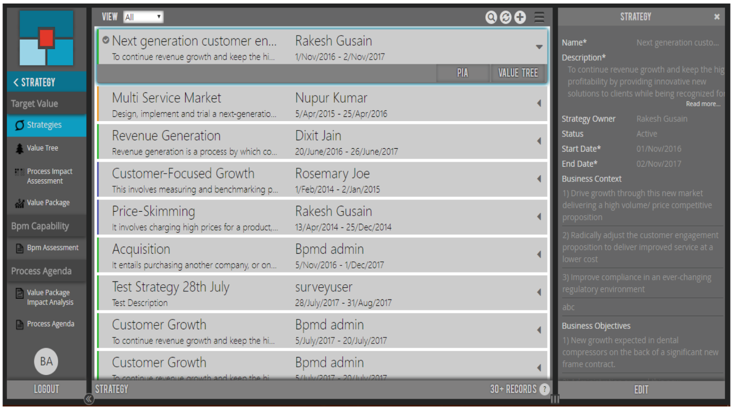


Fig. 14. Strategy definitions in the BPM-D application

This strategy can change based on market conditions, legal regulations or other external and internal factors. Hence, several strategies per organization have to be accessible to ensure an agile ongoing use of the application. Each strategy element is part of the master data. Figure 14 shows the overview for existing strategies in the BPM-D Application.

Those strategies are than, as explained, operationalized through a value-driver tree. This exercise is normally done interactively, for example during a workshop, using the BPM-D Application. Therefore, an easy to use data capturing and immediate graphical representation is important.

The value-drivers themselves need to be weighted to provide a good picture of the strategic intent. This weighting process is supported through appropriate voting-capabilities which result in suggested weights. Digitalization helps here to achieve efficient and effective results, saving significant time for the involved executives. The development of the realization of the value-drivers can be measured through appropriate enterprise performance indicators.

Figure 15 shows the value-driver tree in the BPM-D application. It is connected to a strategy. New strategies require new value-driver trees aligned with the specific strategy. Value-drivers are also considered master data of the BPM-D Application, allowing their broad use and central maintenance. The value-drivers can be managed independently of the overlaying value-driver tree as part of the ongoing adjustment of the strategy.

The value drivers need to be assigned to enterprise key performance indicators (KPI). These indicators describe the enterprise wide status of the strategy execution. They are supported through process KPIs which are discussed below. The use of those KPIs in the overall context of the project portfolio management is important to enable a strategy-based project portfolio management. The management of KPIs and related links to value-drivers can become a key advantage of this digital approach to the PoPM.

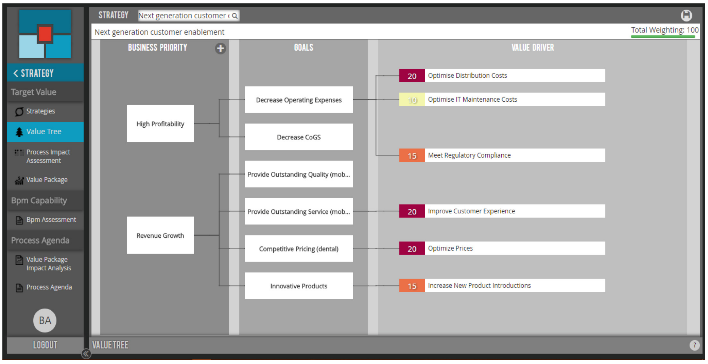


Fig. 15. Value-driver tree in the BPM-D application

The value-drivers are linked to the level 3 business processes through a process impact assessment matrix, as explained above. The processes can be captured in form of a hierarchical function tree or imported from a process repository. Hence, the BPM-D Application integrates the repository into the overall PoPM digitalization approach, creating additional value through the process models, initially captured in a modelling and repository tool. The top level of this process hierarchy can be displayed as the "operating model" of the organization. This representation is used later on to visualize where high impact areas can be found. This visualization is important for the use on an executive level. The switch from a more operational view on the hierarchy to this executive view is another advantage of this digital execution of the PoPM. Figure 16 illustrates the operating model of an organization in the BPM-D Application.

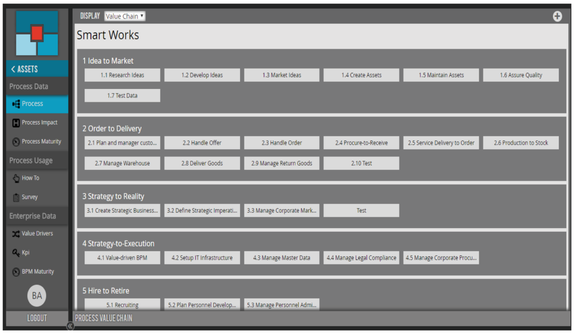


Fig. 16. Operating model of an organization in the BPM-D application

For each process the impact on each value-driver is defined. The weighted total of those impacts represents the total impact of a process on the strategy of the organization. The digital approach allows an immediate visualization of the effect of changing weights of value-drivers, adding or de-selecting value-drivers. This allows the evaluation of the operational effects of different strategy scenarios.

The impact of a process on a value-driver can be described through an appropriate key performance indicator (KPI), used to manage this process. This creates the basis for a process-oriented and strategy-based process management after the improvement project has concluded. Figure 17 shows this process impact matrix.

Fig. 17. Process impact assessment matrix in the BPM-D application

The KPIs behind the impact fields of the matrix are used to drive the process performance. The KPI definitions are also managed through the BPM-D Application. These KPI definitions should be transferred to a process monitoring and analytics system for the ongoing use. This link is not implemented yet, but in the improvement plans. The process KPIs also form the basis for the definition of controls to enable the value-realization after the project. Hence, they support again the logical integration of components of the PoPM.

The impact value (no, low, medium or high impact) can be obtained through interactive workshops, supported through the BPM-D Application, or a survey of key stakeholders and practitioners involved in the processes. Figure 18 visualizes this survey functionality. It simplifies the cross-functional collaboration of involved stakeholders, leveraging the digital internet-based integration.

The high impact processes are evaluated regarding their maturity level compared to industry peers, similar organizations or known good practices. This requires the identification of the current maturity level as well as the target in order to achieve the necessary impact on the value-drivers to get the business strategy executed. This executive decision can either be taken based on existing information or supported through additional analysis. Practice has shown that most executives have a very good idea of current and required maturity level. Figure 19 shows how those maturity levels are captured in the BPM-D Application. The digital support allows again either a capturing during a workshop or in broader surveys. This simplifies the ongoing re-evaluation of this maturity and the management of necessary adjustments.

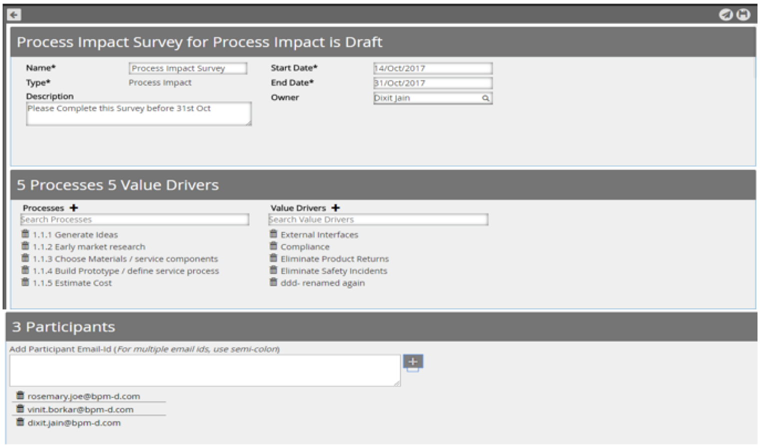


Fig. 18. Survey supported through the BPM-D application

Fig. 19. Current and targeted process maturity levels in the BPM-D application

Result are the identification of high impact low maturity processes on which improvement initiatives should focus. Those priority areas can be visualized using the mentioned operating model's depiction of the process hierarchy. This format has proven to be a well suited executive communication tool. This representation is shown in Fig. 20.

The effects of the change of the relevant value-drivers is shown immediately though changing color codes in the operating model graphic. This allows again the comparison of the effects of strategy adjustments, this time on an executive level.

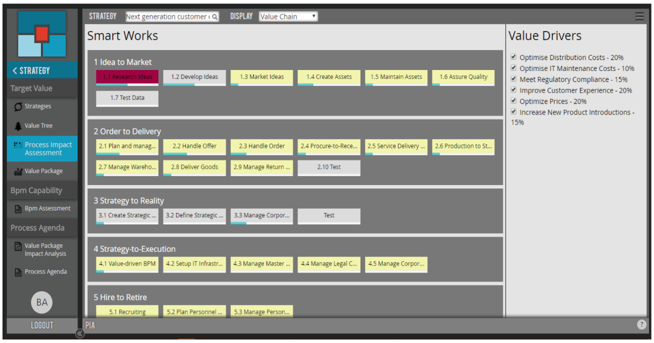


Fig. 20. High impact low maturity processes in operating model format in the BPM-D application

4.2 Identify BPM Capability Gaps

Existing and required BPM capabilities are captured based on the BPM-D Framework (Franz and Kirchmer 2012). This is the maturity assessment of the PoPM, simplified through the use of the BPM-D reference structure and the thereon based digital application. The difference between existing and required capabilities provides the capability gaps that need to be closed during or prior to process improvement projects.

The capability gaps form the basis for the definition of capability improvement initiatives that can be transferred into individual projects. The capturing of the PoPM maturity assessment is done in a similar screen as shown in Fig. 21. It supports a simple easy-to-understand overview over the background of the existing BPM maturity.

Digitalization facilitates the required surveys or workshop sessions. The integration of survey capabilities eliminates working steps required if third party survey tools are used.

4.3 Define Value Packages for High Impact Low Maturity Processes and Related BPM Capabilities

Value packages describe initiatives to address business process improvements, especially of high impact processes, or they close BPM capability gaps to enable the improvement initiatives. Digitalization and integration of this definition of value-packages enables the link of projects to related processes – before going through the administrative steps of defining specific projects.

The impact of those value-packages is calculated based on the impact of the processes addressed. This calculation is not straight forward since it has to be defined using the degree in which a process is addressed by a value package. One package can address one or more processes and a process can be targeted by one or more value packages. We have for the moment decided to have the user define the degree a process is affected by a value package. The digitalization of this work allows the rapid adjustment and management of those dependencies as well as the quick check of different scenarios to create balanced value-packages of manageable scope.

The overall impact of a potential value-package is visualized in table form, as shown in Fig. 21, or again using the operating model format, illustrated in Fig. 22. The value-packages are the basis to define related projects and programs. The management of value-packages and their link to projects is significantly simplified through the use of the BPM-D Application. This is a key enabler of a process-oriented portfolio management approach.

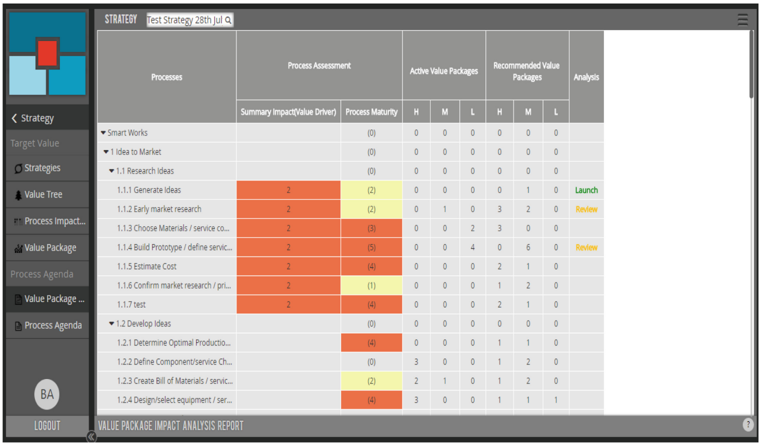


Fig. 21. Impact of value packages in the BPM-D application

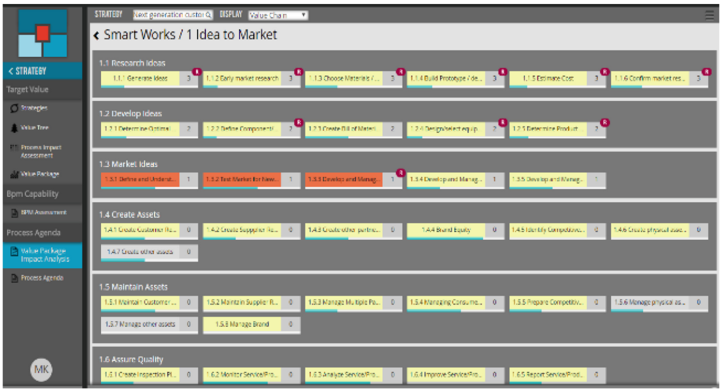


Fig. 22. Operating model with impact of value packages in the BPM-D application

The BPM-D Applications suggests for which processes value-packages should be defined based on the impact and already ongoing initiatives captured in the BPM-D Application before. It also shows where an analysis should be launched to see if initiatives need to be changed or stopped since they target less important business processes or processes addressed by multiple initiatives. This is also illustrated in Fig. 22.

4.4 Transfer Value Packages into Prioritized Project Portfolios

Each value-package is now translated into projects. Therefore, the value packages are evaluated in regard to their overall expected value with consideration of their business impact, complexity and the effort required. This allows to segment those packages into 3 waves:

- Wave 1 value packages need to be addressed in the short terms
- Wave 2 value packages will be executed in a defined mid-term
- Wave 3 value packages relate to future initiatives, no specific timeline is assigned.

The prioritization of value packages is shown in Fig. 23. The digital support of this prioritization enables here, too, a simpler management and adjustment of those decisions. It also helps to facilitate possibly required workshops.

The value packages are then transferred into one or several manageable projects. For waves 1 and 2 this should be initiatives in a limited time line, less than 6, or even better less than 3 months, wherever possible. This definition of smaller projects enables a more agile approach to carrying out improvements in an organization. The BPM-D Application supports that project definition through appropriate digital templates, simplifying this procedure. Wave 3 packages can be assigned to a project right away and operationalized later, when the execution time frame is clearer. Those value-packages are forming a waiting list, managed through the application.

The result is a BPM agenda with projects in the appropriate realization phases. They can now be scheduled and managed in project management system, integrated in the BPM-D Application.

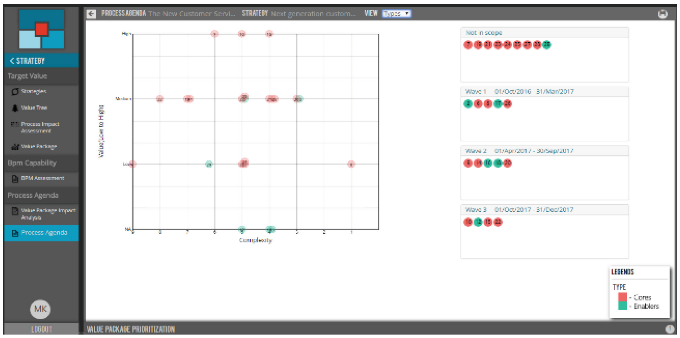


Fig. 23. Prioritization of value packages and segmentation into realization waves in the BPM-D application

Result is a project portfolio based on the impact of involved processes on the organization's business strategy. Each project is directed towards the execution of the overarching strategy. Changing strategy or business environment may lead to changing process impacts and modifications of the project portfolio. Results of the value-realization during and after ongoing projects are captured in the BPM-D Application so that they may also lead to changes in the project portfolio.

5 Digital Approach to Value Realization

The management of the value realization after project conclusion is supported by another BPM-D Application module, part of the overall architecture discussed above. This functionality has already been used successfully for over one year in practice. The core of this value realization is the establishment and management of appropriate controls during the execution of the new and improved business processes.

5.1 Identify and Manage Appropriate Process Controls

The process controls are defined as part of the operational process design, as component of the process governance. They are identified as control activities in the process model describing the to-be process. An example of such a definition is shown in Fig. 11, introduced before. Such a process model is housed in a process repository. The BPM-D Application imports these process models from the repository and extracts the control items. These controls can then be refined, changed and adjusted as required in the BPM-D Application. In case of new controls, these have to be added to the appropriate process models. Figure 24 illustrates the controls management in the BPM-D Application.

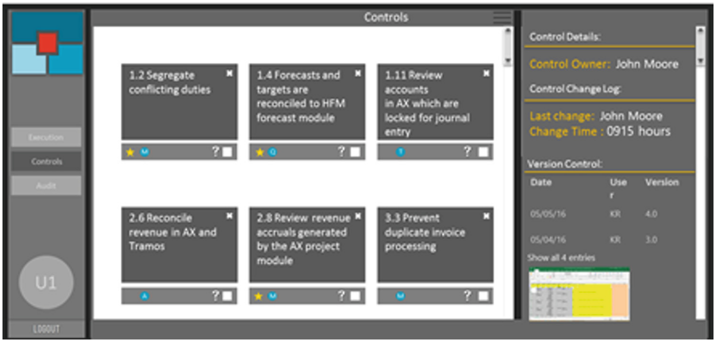


Fig. 24. Managing process controls in the BPM-D application

The controls are then refined in form of process control tasks. Those have to be managed and serve as basis for necessary report. The control tasks can include references to the process KPIs defined as part of the process impact assessment matrix discussed before.

The BPM-D Application uses again content form a modelling and repository tool, hence increases the value of this models. The digitalization of the controls makes their management efficient and, in many cases, even possible. Especially in larger organization a manual management of the controls would be difficult.

5.2 Manage Related Process Control Tasks and Report Results

The control tasks are assigned to specific roles. People logging into the BPM-D Application with such a role are presented the related control tasks. They see the status, can carry out open tasks and report the findings back through the application. The task management in the BPM-D Application is shown in Fig. 25.

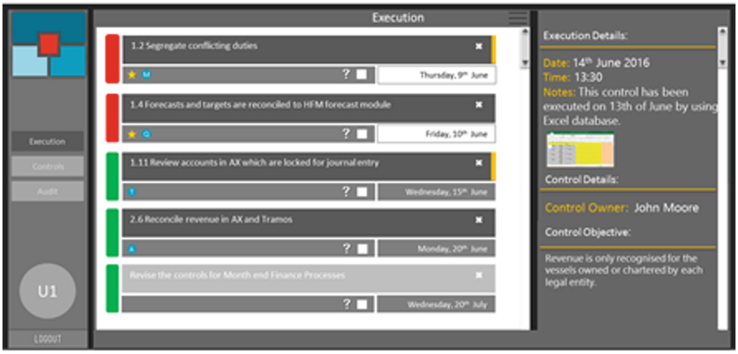


Fig. 25. Task management in the BPM-D application

The role-based management of those control tasks is again basically only possible through appropriate digital support. This enables the appropriate governance to drive value-realization.

The status of the tasks and information about the executing roles can be displayed in appropriate reports. The pilot client of the BPM-D Application uses the software in processes relevant for legal compliance requirements. The delivered reports have been proven to fulfil the formal compliance requirements. An example for an online report is shown in Fig. 26.

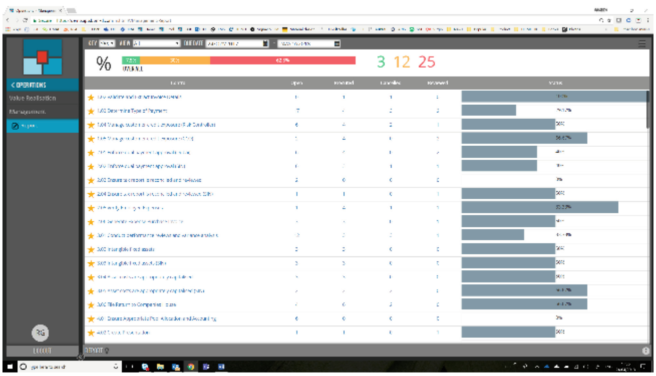


Fig. 26. Reporting compliance task status in the BPM-D application

This reporting through the BPM-D Application is another simplification of the value-realization. The use showed significantly reduced time for the handling while the report quality increased.

6 Conclusion

The implementation and use of the first modules of the BPM-D Application has shown a digital path of moving from the business strategy of an organization to a project-based execution of required initiatives and the following value-realization. The identification of high impact business processes is used as link between strategy and execution. The digital enablers close first important gaps in the digitalization of the process of process management and enable the integration of existing process management tools, the business process repository and project management systems. Compliance relevant reporting is delivered as a side effect.

This digitalization of the process of process management needs now to be again moved to the next level. Therefore, we plan the following next steps:

- The current modules of the BPM-D Application need to be applied at 3-4 pilot organization to get more feedback and incorporate this into the software.
- The formalization of the link of value-packages to processes and projects to value-packages and the related quantification of priorities has proven to be more challenging than expected. We currently use very simple distribution formulas to achieve that. This may need to be refined, also depending on the pilot feedback.
- The integration of process analytics and monitoring systems on the basis of the KPIs defined in the BPM-D Application still has to be designed in detail and implemented. This may also require an extension of the existing functionality of the BPM-D Application to allow the upload of monitoring results to support the value-realization.
- There are still functional and integration gaps regarding the digitalization of the process of process management. These need to be prioritized and the extension of the BPM-D Application planned accordingly.

The digitalization of the management of process-oriented project portfolios and the first component of the following value realization are important steps towards an integrated digital support of the process of process management. This will continue to make the BPM-Discipline itself more efficient and effective which is another step towards systematic strategy execution. It is another step towards digital BPM and the management discipline it supports.

References

Abolhassan, F.: The Drivers of Digital Transformation – Why There is No Way Around the Cloud. Springer, Berlin (2016). https://doi.org/10.1007/978-3-319-31824-0

Cantara, M.: Start up your business process competency center. In: Documentation of The Gartner Business Process Management Summit, National Harbor (2015)

Ewenstein, B., Smith, W., Sologar, A.: Changing Change Management. McKinsey Digital (2015)
Fischer, L.: BPMN 2.0 Handbook – Methods, Concepts, Case Studies and Standards in Business Process Modelling Notation (BPMN). Future Strategies Inc, Lighthouse Point (2012)
Franz, P., Kirchmer, M.: Value-Driven Business Process Management: The Value-Switch for Lasting Competitive Advantage, 1st edn. McGraw-Hill, New York (2012)
Kirchmer, M.: Robotic Process Automation (RPA) – Pragmatic Solution or Dangerous Illusion (2017a). http://insights.btoes.com/risks-robotic-process-automation-pragmatic-solution-or-dangerous-illusion
Kirchmer, M., Franz, P., Gusain, R.: Digitalization of the process of process management – the BPM-D application. In: Proceedings of the Seventh International Symposium on Business Modelling and Software Design, Barcelona, 3–5 July 2017 (2017)
Kirchmer, M.: High Performance through Business Process Management –Strategy Execution in a Digital World, 3rd edition. Springer, New York, Berlin (2017b). https://doi.org/10.1007/978-3-319-51259-4
Kirchmer, M.: Strategy execution in a consumer goods company: achieving immediate benefits while building lasting process capabilities for the digital world. In: Proceedings of BPM 2016, Rio de Janeiro, 18–22 September (2016)
Kirchmer, M., Franz, P., Lotterer, A., Antonucci, Y., Laengle, S.: The Value-Switch for Digitalisation Initiatives: Business Process Management. BPM-D Whitepaper, Philadelphia, London (2016)
Kirchmer, M.: The process of process management – mastering the new normal in a digital world. In: BMSD Proceedings, July 2015 (2015, in publication)
Kirchmer, M., Franz, P.: Chief Process Officer – The Value Scout. BPM-D Whitepaper, Philadelphia, London (2014a)
Kirchmer, M., Franz, P.: Targeting Value in a Digital World. BPM-D Whitepaper, Philadelphia, London (2014b)
McDonald, M.P.: Digital strategy does not equal IT strategy. Harvard Bus. Rev., **19** (2012)
Nixon, N.W.: Viewing ascension health from a design thinking perspective. J. Organ. Des. **2**, 23–28 (2013)
Scheer, A.-W.: Performancesteigerung durch Auto-matisierung von Geschaeftsprocessen. Whitepaper, August-Wilhelm Scheer Institute for Digital Products and Processes, Saarbruecken, Germany (2017)
Scheer, A.-W.: Industry 4.0: from vision to implementation. Whitepaper Number 5, August-Wilhelm Scheer Institute for Digital Products and Processes, Scheer GMBH, Saarbruecken, Germany (2015)
Scheer, A.-W.: ARIS – Business Process Frameworks, 2nd edn. Springer, Berlin (1998). https://doi.org/10.1007/978-3-642-97738-1
Shapiro, R., White, S.A., Bock, C., et al.: BPMN 2.0 Handbook – Methods, Concepts, Case Studies and Standards in Business Process Modelling Notation (BPMN), 2nd edn. Future Strategies, Inc., Lighthouse Point (2012)
Sims, C., Johnson, H.L.: Scrum: A Breathtakingly Brief and Agile Introduction. Dymax, Menlo Park (2014)
Spanyi, A.: Business Process Management is a Team Sport – Play it to Win!. Anclote Press, Tampa (2003)
Swenson, K.D., von Rosing, M.: What is business process management. In: von Rosing, M., Scheer, A.-W., von Scheel, H. (eds.) The Complete Business Process Handbook – Body of Knowledge from Process Modeling to BPM, vol. 1, pp. 79–88. Morgan Kaufmann, Amsterdam (2015)
von Rosing, M., Scheer, A.-W., von Scheel, H. (eds.): The Complete Business Process Handbook– Body of Knowledge from Process Modeling to BPM, vol. 1. Morgan Kaufmann, Amsterdam (2015)

Blockchain-Based Traceability of Inter-organisational Business Processes

Claudio Di Ciccio(✉), Alessio Cecconi, Jan Mendling, Dominik Felix, Dominik Haas, Daniel Lilek, Florian Riel, Andreas Rumpl, and Philipp Uhlig

Vienna University of Economics and Business, Vienna, Austria
{claudio.di.ciccio,alessio.cecconi,jan.mendling}@wu.ac.at, dominikfelix@gmail.com, dominik.haas1993@gmail.com, daniel.lilek90@gmail.com, flori0n.riel@gmail.com, andreasrumpl@gmail.com, philippuhlig1204@gmail.com

Abstract. Blockchain technology opens up new opportunities for Business Process Management. This is mainly due to its unprecedented capability to let transactions be automatically executed and recorded by Smart Contracts in multi-peer environments, in a decentralised fashion and without central authoritative players to govern the workflow. In this way, blockchains also provide traceability. Traceability of information plays a pivotal role particularly in those supply chains where multiple parties are involved and rigorous criteria must be fulfilled to lead to a successful outcome. In this paper, we investigate how to run a business process in the context of a supply chain on a blockchain infrastructure so as to provide full traceability of its run-time enactment. Our approach retrieves information to trace process instances execution solely from the transactions written on-chain. To do so, hash-codes are reverse-engineered based on the Solidity Smart Contract encoding of the generating process. We show the results of our investigation by means of an implemented software prototype, with a case study on the reportedly challenging context of the pharmaceutical supply chain.

Keywords: Blockchain · Ethereum · Smart Contracts · Supply chain Business Process Management

1 Introduction

Integrating processes that extend throughout the supply chain requires multi-party collaborations and an intense exchange of information along multiple channels. The multitude of passages entails data redundancy and lack of full knowledge on how, when and where tasks were conducted, and products used through their life cycle [24]. This becomes an issue of particular concern for inter-organisational processes involving untrusted parties [6]. In a pharmaceutical supply chain, for instance, critical goods are handled and the verifiability of their origin and processing is needed to prevent illegal actions, e.g., the distribution of counterfeit drugs. Multiple actors work together on delivering a product

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 56–68, 2018.
https://doi.org/10.1007/978-3-319-94214-8_4

that needs to have very specific qualities to be fulfilled. Thus *traceability* is required. Classic approaches against counterfeiting, like the one proposed in [25] are not always sufficient, because supply chains usually lack trust, transparency, and documentation. The key properties of the blockchain technology offers a promising solution to this problems without the need of a third party authority.

Blockchain has emerged as an open, distributed ledger that can record transactions between parties efficiently and in a verifiable and permanent way. This is enabled by a combination of peer-to-peer networks, consensus-making, cryptography, and market mechanisms [5]. Blockchains thus ensure data integrity and transparency [22]. Furthermore, they support so-called Smart Contracts, that is, fully executable pieces of decentralised code expressing how business is to be conducted among contracting parties, e.g., transfer cryptocurrencies and digital assets after a condition is fulfilled [7,28]. These characteristics make it particularly suitable to the execution of inter-organisational business processes along supply chains [30].

In this paper we investigate the application of the Ethereum blockchain to enable traceability of inter-organisational business process, focussing on a case study taken from the pharmaceutical domain. We remark that we abstract from the sole product traceability in supply chains [17,29] and aim at extending the concept towards full traceability of the entire process execution [16]. To that extent, we rely on an existing platform for running business processes on the Ethereum blockchain [4,12] and devise a framework to trace the execution information solely based on the transactions recorded on-chain. Our approach is implemented in a software prototype and its usage demonstrated on an exemplifying process.

The remainder of the paper is structured as follows. Section 2 describes the preliminary notions upon which out approach is based. Section 3 illustrates our approach in detail. Section 4 concludes the paper and draws future research plans.

2 Background

An ever-increasing number of organisations both in the private and public sectors identify their business processes as a key asset. Business processes, or *processes* hereinafter, regulate the inter-relation, assignment, and execution of tasks and decisions that ultimately yield an outcome that adds value for a customer [6]. Business Process Model and Notation (BPMN) is among the most prominent modelling languages for processes [21]. The integration of business processes along the supply chain, has been found to contribute both to better operational and business performance [15,30].

A supply chain is defined as the set of activities and independent organisations whose cooperation delivers a product from its production to the end consumer [13,26]. A pharmaceutical supply chain, for instance, encompasses the stages pertaining to a pharmaceutical product from the raw material processing to the end consumer acquisition of the final product. More in detail, it includes

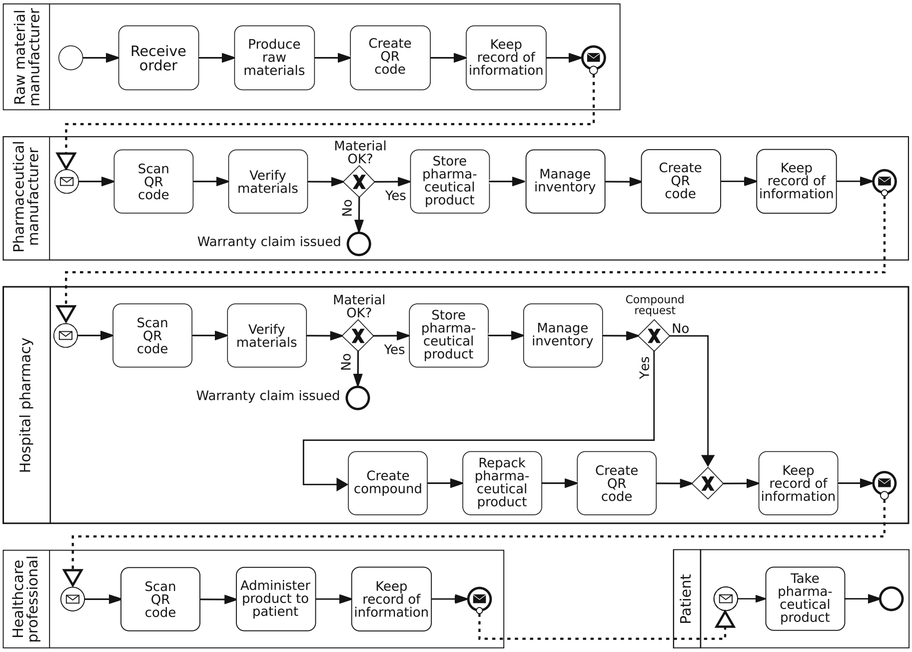


Fig. 1. Simplified pharmaceutical supply chain modelled as a BPMN collaboration diagram.

the following tasks: *(i)* manufacturers produce raw materials, *(ii)* pharmaceutical manufacturers produce the final product, *(iii)* hospitals and pharmacies store and dispense the final product dispatched by wholesalers and distributors, *(iv)* patients take the final product. The coactive work of the many parties involved is paramount as the efforts lead to a successful outcome only if the internal coordination, information exchange, and material flow is efficient. A weakness anywhere in the chain results in an overall failure to achieve the competitive potential of the whole chain. In other words, the chain is only as strong as its weakest link [27]. Standards exist to regulate both the activity flow and the communication between the actors of supply chains. A simplified model of the pharmaceutical supply chain, e.g., is illustrated in Fig. 1 as a BPMN collaboration diagram. It is based upon the model proposed in [10] by GS1, a non-profit organisation that develops and maintains global standards for business communication. The diagram consists of five pools, represented as wrapping labelled boxes, each depicting one of the different parties involved: raw material manufacturer, pharmaceutical manufacturer, hospital pharmacy, healthcare professional, and patient. Within each pool, processes are represented as tasks (rounded boxes) connected by control-flow (arcs), starting and ending with events (circles, drawn with narrow and bold lines respectively). Envelopes represent a message exchange, and dotted arcs connect sending events (black envelope) to catching events (transparent envelope). For instance, pharmaceutical products are delivered to the hospital pharmacy from the pharmaceutical manufacturer after task Keep record of information is concluded. The pharmacy thereafter scans the QR code of the package to register the delivery. Routing choices are drawn as diamonds with a cross. Considering the process of the pharmaceutical manufacturer, e.g., if raw materials are corrupted the process ends with a warranty claim issued, otherwise it proceeds with the production of pharmaceutical product.

A cornerstone of Business Process Management (BPM) is the integration of IT systems in the management of processes to favour their automation. Blockchains can provide an underlying technological infrastructure for business process management.

2.1 Blockchain

The concept of blockchain was introduced in 2008 to support the creation and exchange of the *Bitcoin* cryptocurrency without the need for a controlling central authority [14,20]. While indeed the primary driver of blockchain technology has been cryptocurrencies, it has later found a broader application field outside the sole finance scope. The foundation of a blockchain is the decentralised storage of data that describe transactions. When we think of a traditional bank, the bank is a centralised node of all transactions, the ledger. By contrast, the blockchain network stores the transactions on the different peers that are part of the network, in a *distributed ledger* fashion. This allows for a direct exchange between different participants without a third party that observes the transactions. The transactions are collated in blocks and several blocks together result in a chain. In this way, a blockchain describes a logical sequence of transactions. Adding new blocks in a trustworthy manner is enabled by a combination of peer-to-peer networks, consensus-making, cryptography, and market mechanisms [19]. Blockchains ensure data integrity and transparency, such that the blockchain network stays operational even under byzantine faults. A new block can only be added to the chain if there is a *consensus* on the majority of the network. The most common consensus techniques are Proof-of-Work (PoW) and Proof-of-Stake (PoS) [2]. Ethereum [32] is among the most used blockchain implementations.

All transactions are signed. Nevertheless, the identity of the signee remains anonymous. Consensus and anonymity are facilitated by hashing. A *cryptographic hash function* (henceforth simply *hash function*) is a deterministic one-way function mapping any given input to a fixed-size output. It means that given the same input, the same output is always returned. It is however not feasible to retrieve the original input from the sole output. Keccak [3] is the hash function used in Ethereum. Every block is associated with a hash generated from its content *and* the hash value of the previous block in the list. Hash values thus uniquely represent not only the transactions within blocks but also the ordering of every block. This mechanism is at the basis of the chain. In case somebody tried to alter a transaction, this would change the hash value of its block, and thus break the chain. Different types of blockchains exist, but they can be generally classified by two properties, i.e., verifiability (*private* or *public*), and access grant (*permissioned* or *permissionless*) [33]. The difference between *public* and *private blockchains* is the visibility of the network: in the former the network can be joined and inspected by everyone, while in the latter only by those who are invited. Instead, *permissionless* and *permissioned blockchains* are differentiated by the capability to add blocks to the chain: in the former every peer in the network can add blocks, i.e., participate to the consensus, while in the latter only allowed ones by the network.

Blockchains allow for the execution of Smart Contracts, i.e., software programs that react upon triggers provided by users or generated by the environment, running within the blockchain [7,9,28]. Distributed ledger systems constitute computational platforms then, thus going beyond mere distributed databases. The *Ethereum* blockchain for instance provides a Turing-complete language to encode Smart Contracts, namely *Solidity*[1] [4]. Computational effort comes at a price for Smart Contracts. Operations are associated to a cost measured in *gas*, a unit of work that is meant to be paid in *Ether*, the Ethereum cryptocurrency. In order for a Smart Contract to complete its execution, enough Ether have to be in the wallet of the invoker.

The implication of using blockchain for BPM has been discussed under diverse viewpoints. Mendling et al. [16] highlight the challenges and opportunities of blockchain for BPM in relation to the six BPM core capability areas [23] and in relation to the traditional BPM lifecycle [6]. The first of seven future research directions listed in [16] relates to our work in particular, namely the development of a diverse set of execution and monitoring systems on blockchain. The fundamental problem of trust in collaborative process execution is highlighted in [30]. The authors develop a technique to integrate blockchain into the choreography of processes in order to avoid the need for a central authority and still maintain trust. A first method to compile a process model directly into an Ethereum Smart Contract is presented in [8] by focussing on three areas: initialisation cost for process instances, task execution cost by means of a space-optimised data structure and improved runtime improvements for maximised throughput. The implemented software prototype, Caterpillar [12], is used in the context of our research.

3 Approach

Enabling traceability of inter-organisational business processes implies the capability of tracking the status of ongoing instances and reconstructing the history of its execution. Such a view transcends the monitoring of sole business objects, information artefacts, or carriers, hence transitioning to a more holistic approach. To that extent, in this work we focus on the identification and linkage of all transactions that report on activities of running processes. The blockchain opens up the opportunity of retrieving that information, which is digitally stored and shared by different parties in a decentralised manner.

[1] https://solidity.readthedocs.io.

Table 1. Addresses of Smart Contracts (SCs) deployed by Caterpillar on the blockchain.

	Global Factory SC address:	`0x8cdaf0cd259887258bc13a92c0a6da92698644c0`
	Process Factory SC address:	`0x345ca3e014aaf5dca488057592ee47305d9b3e10`
	Worklist Factory SC address:	`0xf12b5dd4ead5f743c6baa640b0216200e89b60da`
Prc. Instance	**Worklist SC address**	**Process Instance SC address**
1	`0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2`	`0xf2beae25b23f0ccdd234410354cb42d08ed54981`
2	`0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd`	`0xaa8f61728cb614f37a2fdb8b420c3c33134c7f69`
3	`0x22029e89e1d1f79d8e57c9af2fb9bf653bdf4be1`	`0xf21cf97429e6f7338ae989135ff9aa0225719347`

Our approach begins with the encoding of supply chain business process models into executable Solidity programs. To that extent, we resort on the Caterpillar tool [12]. We assume that activities executed by Solidity Smart Contracts are registered as transactions. This is typical in collaborative business processes run on the blockchain [8,30]. Thereupon, we resort on the Solidity encoding of the process model to reverse-engineer the hash-codes and reveal *(i)* the process instance committing the transaction, and *(ii)* the operation signature, denoting the task. We remark here that the latest step requires knowledge on the encoded process model because hash-codes cannot be reverted into original input signatures. In the following, we detail the aforementioned passages.

3.1 Caterpillar: From Process Models to Smart Contracts

Caterpillar acts as a Business Process Management System (BPMS) operating on the blockchain by means of Smart Contracts (SCs). The Caterpillar tool [12] relies on a *Global Factory* SC, acting as a container for all processes and deployed on the blockchain as soon as Caterpillar is launched. Given a BPMN process model like the one of Fig. 1, the tool generates a *Process Factory* SC and a *Worklist Factory* SC. In turn, the Worklist Factory SC generates new *Worklist* SCs. Each of them is associated to a *Process Instance* SC, generated by the Process Factory SC at every start of the process. The Worklist SC routes the execution of process activities as dictated by the workflow depicted in the process model. In other words, the control-flow logic is embedded in every Worklist SC. Every activity corresponds to a function of the Worklist SC, which then forwards the call to the corresponding Process Instance SC. Notice that the Worklist SC registers a transaction for every activity execution. Table 1 shows examples of addresses assigned by Caterpillar to the Smart Contracts described so far.

3.2 Executing Processes Through Smart Contracts

Let us consider that a unit of medicines is handed over from the manufacturer to the hospital pharmacy, as in the example of Fig. 1. A transaction is required to confirm that the hand-over took place. Nowadays, a centralised system would register this information and involved parties require access to that.

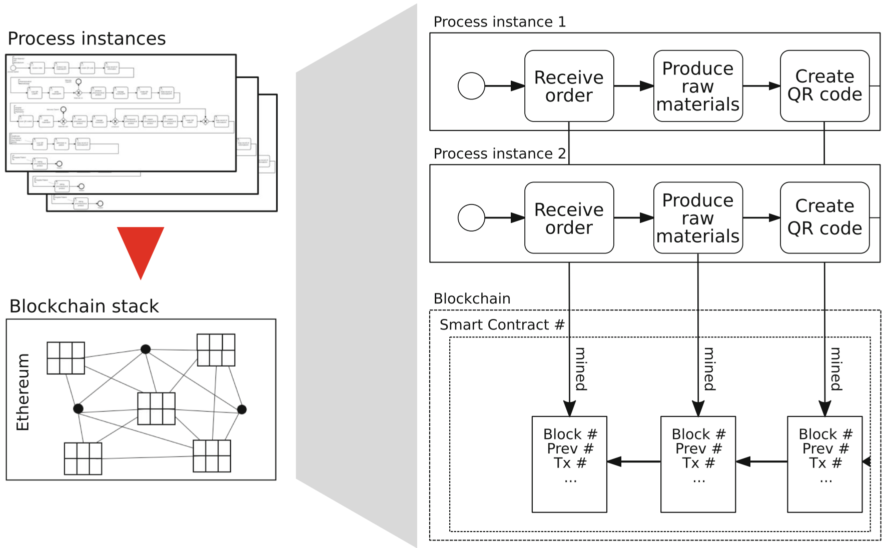


Fig. 2. The execution of process instances on the blockchain

This naturally extends to the blockchain environment, which makes the information readily available to all counterparts in a decentralised manner. Figure 2 depicts how process instances are deployed and executed on the blockchain through Caterpillar.

Once a process instance is initiated through Caterpillar, a new block is added to the chain each time an activity is executed. Inside the block, the transaction reporting on the activity execution is collated. Transactions record, among other things, *(i)* the *contract address* defining the called contract, i.e., the Worklist SC, *(ii)* the *sender address* of the user carrying out the activity, and *(iii)* the *transaction data*, containing the hash-code of the function signature corresponding to the activity. They are the fundamental building blocks for traceability.

3.3 Tracking Activities in the Blockchain

Different process instances may run at the same time. Accordingly the resulting blockchain will be a sequence of blocks originating from different instances. Furthermore, each block is identified by its hash-code, which does not explicitly disclose the process instance that generated the block itself. Therefore we present a procedure to differentiate the process instances in order to allow for their tracking, as depicted in Fig. 3.

Through Ethereum modules it is possible to extract the blocks of a blockchain, along with the individual transactions and all the information held by them. In particular we are interested in the aforementioned *contract address* and *transaction data*. Interestingly, operational transactions generated from the process instance execution are such that the *contract address*es correspond to

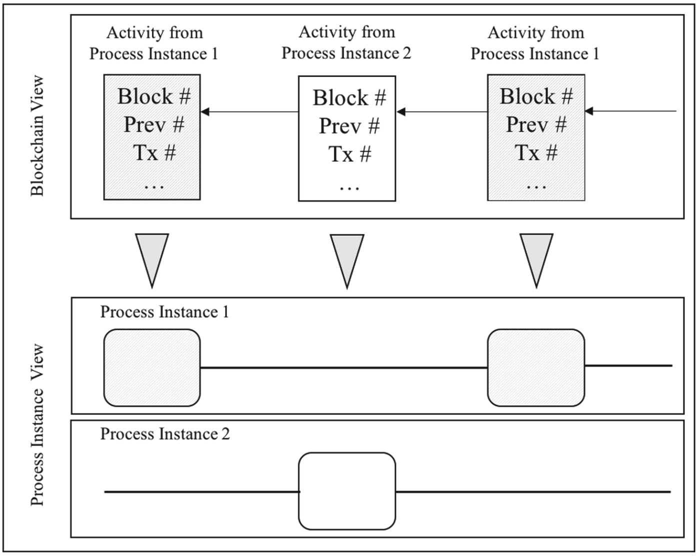


Fig. 3. Extraction of process instances

the address of the router of activities via function calls, i.e., of the Worklist SC, which can be retrieved via Application-Binary Interface (ABI). Therefore, to trace the execution of a process instance, we search for all the transactions resulting from the invocation of the corresponding Worklist SC: to that extent, we look up for matches between the value of the contract address field in the transaction payload and the Worklist SC registered in Caterpillar as depicted in Table 2.

To extract the specific task contained in a block, we fetch the content of the *transaction data* field of every transaction. Because it stores the function signature, it reveals a reference to the executed task once encoded. The *transaction data* consists in a hash-code which can be divided in two parts: the first 4 bytes represent the *function identifier*, while the following bytes enclose the actual parameters of the function. We thus identify the executed activity by parsing the *transaction data* hash-code, so as to compare it with the hashed function signatures.

To sum up, we compare the *contract address* of a transaction to the hash-code of the Worklist SC to identify the process instance, and we match the *function identifier* of the *transaction data* with the hash-code of function signatures to retrieve the activity.

3.4 Example

An excerpt of all addresses assigned to the Smart Contracts (SCs) deployed on the blockchain by Caterpillar is reported in Table 1. We consider three process instances, numbered from 1 to 3 in the table. We recall that to every process instance correspond two deployed SCs, namely a Worklist SC, generated

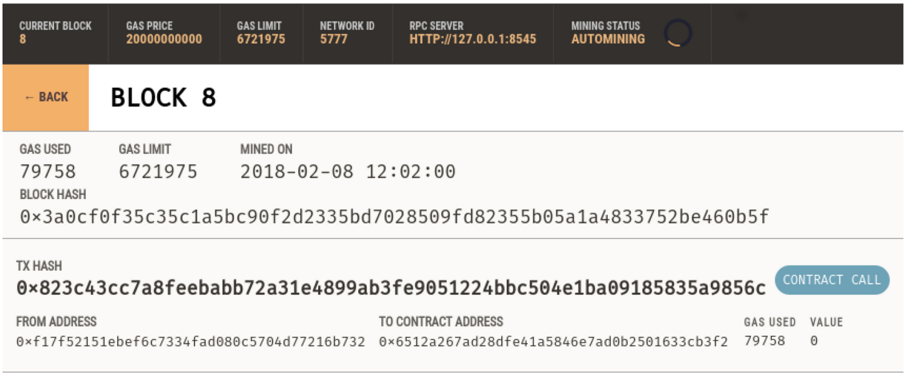
Table 2. Excerpt of recorded transactions and mapping to process instances via matching of the addresses listed in Table 1.

Transaction	Contract address	Instance
1	0x8cdaf0cd259887258bc13a92c0a6da92698644c0	
2	0xf12b5dd4ead5f743c6baa640b0216200e89b60da	
...	...	
7	0x8cdaf0cd259887258bc13a92c0a6da92698644c0	
8	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1
9	0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd	2
10	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1
11	0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd	2
12	0x8cdaf0cd259887258bc13a92c0a6da92698644c0	
13	0x8cdaf0cd259887258bc13a92c0a6da92698644c0	
14	0x22029e89e1d1f79d8e57c9af2fb9bf653bdf4be1	3
15	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1
16	0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd	2
17	0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd	2
18	0x22029e89e1d1f79d8e57c9af2fb9bf653bdf4be1	3
19	0x22029e89e1d1f79d8e57c9af2fb9bf653bdf4be1	3
20	0x0ebe109b4ac5de65d63f7d7e5a856dcd77dc58fd	2
21	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1
22	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1
23	0x22029e89e1d1f79d8e57c9af2fb9bf653bdf4be1	3
24	0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2	1

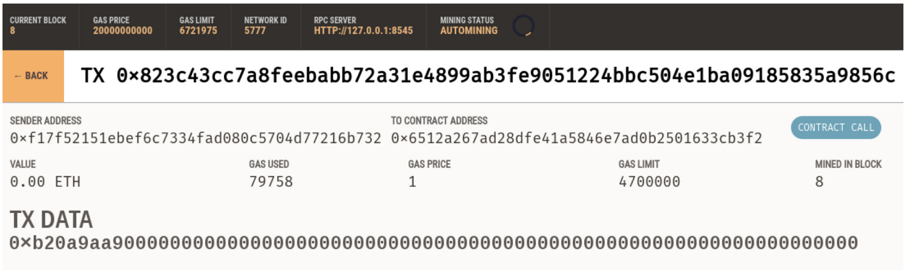
by the Worklist Factory SC, and a Process Instance SC, generated by the Process Factory SC. The hexadecimal address of the Worklist SC of process instance 1 in Table 1 is, e.g., `0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2`, associated to the Process Instance SC having address `0xf2beae25b23f0ccdd234410354cb42d08ed54981`. The Worklist SC is the façade for the execution of tasks: single operations correspond to a task each, and re-route the call to the Process Instance SC once invoked. Considering the example process of Fig. 1, the signature of the function in the Worklist SC corresponding to Receive order is `receive_order(uint256)`. Every execution of Receive order corresponds to a transaction reporting the invocation of `receive_order(uint256)` on the Worklist SC of a process instance.

Figure 4 illustrates the information displayed by the Ganache tool[2] on the transactions recorded in the blockchain. More specifically, the block in Fig. 4(a) contains the transaction reporting a task execution of the example process. Transaction-specific data are illustrated in Fig. 4(b). Notice that the *contract address* field is `0x6512a267ad28dfe41a5846e7ad0b2501633cb3f2`, i.e., the same as the address of the aforementioned Worklist SC. This indicates that the running instance is the one we marked as instance 1 in Table 1. Table 2 shows a list of such

[2] http://truffleframework.com/ganache/.



(a) The block containing the transaction of Receive order.



(b) The transaction in the block of Fig. 4(a) reporting on the execution of Receive order. The *transaction data* field is magnified for the sake of readability.

Fig. 4. Tracing of task execution on the blockchain

mappings between transactions and process instances based on data registered on the blockchain.

To understand what the executed activity is, we analyse the first 4 bytes of the hexadecimal *transaction data* field of the recorded transaction, e.g., `0xb20a9aa9` in Fig. 4(b). The Keccak hash of the signature of the operation corresponding to task Receive order, i.e., `receive_order(uint256)`, is the following: `0xb20a9aa9619b9dc0c6a76bef0c51350699afd93b05757350f8c20de71bc75e18`. We observe that its first 4 bytes are `0xb20a9aa9` too. We thereby recognise that the reported execution corresponds to task Receive order of process instance 1.

3.5 Limitations

The initial results of our approach demonstrate the applicability of a fully automated technique to enable traceability of inter-organisational processes deployed on the blockchain. However, we acknowledge limitations which draw future research endeavours. First and foremost, we remark that the hash-code comparison to identify process and task instances is based on Caterpillar. However, the rationale is rooted in a reasonable design choice which other blockchain-based BPMSs might adopt as well. Indeed, the underlying idea follows the paradigm of self-identification which is inherent to the concept of blockchain, according to which transactions must be signed by the transactor – in this case, contracts enabling the process execution.

Our approach to identify process and task instances requires that interested parties share knowledge on the hash key and hashing function in use. This calls for further investigations, because the fully public access to that information might be undesirable as knowledge on the processes could be disclosed. Those studies would build upon *(i)* methodologies for the specification of interfacing (visible) and internal (hidden) fragments of cooperating processes [1,31] and *(ii)* infrastructural solutions based on permissioned access grants enforced by cryptography of undisclosed information [18].

At the time of writing, the capability of handling BPMN pools in Caterpillar is under ongoing implementation, therefore unique process instance identifiers are used to indicate all operations pertaining to the inter-organisational processes. In a distributed scenario, diverse process instances with different identifiers could run, e.g., one per pool, even on different platforms accessing the shared blockchain. This suggests the implementation of novel techniques based upon record linkage and object matching [11,34] that allow for an automated recognition of matching instances.

4 Conclusions

In this paper we presented a technical solution to ensure traceability through blockchain of inter-organisational business process. Our case study pertains to the pharmaceutical domain, and is based upon the model defined by the GS1 standard.

The results gathered through the investigation via our implemented prototype shows feasibility and correctness of the approach, thus representing a first step towards a comprehensive framework. Our future work to extend the approach includes the overcoming of discussed limitations, as well as the following points. Firstly, it is in our plans to investigate the integration of other BPMSs and blockchain technologies than Caterpillar with Ethereum, e.g., IBM Blueworks Live[3] with Hyperledger[4], or Bonitasoft Bonita BPM[5] with Chain Core[6]. To that extent, we will create dedicated programming interfaces that act as a façade towards the implementation layer, thus allowing for higher-level BPM functionalities. Furthermore, we argue that the exchange of information between the blockchain and the real-world is crucial in a supply chain. Therefore we aim at extending our approach to include agents connecting with the real world from within the blockchain, also known as *oracles*, also considering possible security threats arising from them, e.g., trustfulness of those agents. This would enable the triggering of activity completions via scanning of QR-codes, e.g., as suggested

[3] https://www.ibm.com/cloud/automation-software/business-process-management.
[4] https://hyperledger.org.
[5] https://www.bonitasoft.com.
[6] https://chain.com/technology.

by GS1 [10]. Finally, we aim at the creation of rich event logs to allow for process mining on the blockchain [16].

Acknowledgements. The work of Claudio Di Ciccio was partially funded by the Austrian Research Promotion Agency (FFG) under grant 862950 (Business Process Optimization Toolkit). The work of Alessio Cecconi was funded by the Austrian Research Promotion Agency (FFG) under grant 861213 (CitySPIN).

The authors want to thank Orlenys López-Pintado, Luciano García Bañuelos, and Marlon Dumas, for the valuable advice and precious technical help provided with Caterpillar.

References

1. van der Aalst, W.M.P., Mooij, A.J., Stahl, C., Wolf, K.: Service interaction: patterns, formalization, and analysis. In: Bernardo, M., Padovani, L., Zavattaro, G. (eds.) SFM 2009. LNCS, vol. 5569, pp. 42–88. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01918-0_2
2. Bentov, I., Gabizon, A., Mizrahi, A.: Cryptocurrencies without proof of work. In: Clark, J., Meiklejohn, S., Ryan, P.Y.A., Wallach, D., Brenner, M., Rohloff, K. (eds.) FC 2016. LNCS, vol. 9604, pp. 142–157. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53357-4_10
3. Bertoni, G., Daemen, J., Peeters, M., Assche, G.V.: The making of KECCAK. Cryptologia **38**(1), 26–60 (2014). https://doi.org/10.1080/01611194.2013.856818
4. Dannen, C.: Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners. Apress, New York (2017)
5. Diedrich, H.: Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations. Wildfire Publishing, Sydney (2016)
6. Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management, 2nd edn. Springer, Heidelberg (2018). https://doi.org/10.1007/978-3-662-56509-4
7. Egelund-Müller, B., Elsman, M., Henglein, F., Ross, O.: Automated execution of financial contracts on blockchains. Bus. Inf. Syst. Eng. **59**(6), 457–467 (2017)
8. García-Bañuelos, L., Ponomarev, A., Dumas, M., Weber, I.: Optimized execution of business processes on blockchain. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 130–146. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_8
9. Governatori, G., Idelberger, F., Milosevic, Z., Riveret, R., Sartor, G., Xu, X.: On legal contracts, imperative and declarative smart contracts, and blockchain systems. Artif. Intell. Law, 1–33 (2018). https://doi.org/10.1007/s10506-018-9223-3
10. GS1: Global Traceability Standard for Healthcare (2013). https://www.gs1.org/sites/default/files/docs/traceability/Global_Traceability_Standard_Healthcare.pdf
11. Inan, A., Kantarcioglu, M., Bertino, E., Scannapieco, M.: A hybrid approach to private record linkage. In: ICDE, pp. 496–505. IEEE Computer Society (2008)
12. López-Pintado, O., García-Bañuelos, L., Dumas, M., Weber, I.: Caterpillar: a blockchain-based business process management system. In: BPM Demos, vol. 1920. CEUR-WS.org (2017)
13. Lu, D.: Fundamentals of Supply Chain Management. Ventus Publishing ApS, Frederiksberg (2011)
14. Magazzeni, D., McBurney, P., Nash, W.: Validation and verification of smart contracts: a research agenda. IEEE Comput. **50**(9), 50–57 (2017)

15. McAdam, R., McCormack, D.: Integrating business processes for global alignment and supply chain management. Bus. Proc. Manag. J. **7**(2), 113–130 (2001)
16. Mendling, J., Weber, I., van der Aalst, W.M.P., vom Brocke, J., Cabanillas, C., Daniel, F., Debois, S., Di Ciccio, C., Dumas, M., Dustdar, S., Gal, A., García-Bañuelos, L., Governatori, G., Hull, R., Rosa, M.L., Leopold, H., Leymann, F., Recker, J., Reichert, M., Reijers, H.A., Rinderle-Ma, S., Solti, A., Rosemann, M., Schulte, S., Singh, M.P., Slaats, T., Staples, M., Weber, B., Weidlich, M., Weske, M., Xu, X., Zhu, L.: Blockchains for business process management - challenges and opportunities. ACM Trans. Manag. Inf. Syst. **9**(1), 41–416 (2018)
17. Merminod, N., Paché, G.: Supply management and corporate social responsibility: the challenge of global chain traceability. J. Chain Netw. Sci. **11**, 213–222 (2011)
18. Miklau, G., Suciu, D.: Controlling access to published data using cryptography. In: VLDB, pp. 898–909. Morgan Kaufmann (2003)
19. Mougayar, W.: The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology. Wiley, Hoboken (2016)
20. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008). https://bitcoin.org/bitcoin.pdf
21. OMG: Business Process Model And Notation Specification Version 2.0 (2011). https://www.omg.org/spec/BPMN/2.0
22. Risius, M., Spohrer, K.: A blockchain research framework - what we (don't) know, where we go from here, and how we will get there. Bus. Inf. Syst. Eng. **59**(6), 385–409 (2017)
23. Rosemann, M., vom Brocke, J.: The six core elements of business process management. In: vom Brocke, J., Rosemann, M. (eds.) Handbook on Business Process Management 1. IHIS, pp. 105–122. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-642-45100-3_5
24. Saveen, A., Monfared, R.P.: Blockchain ready manufacturing supply chain using distributed ledger. Int. J. Res. Eng. Technol. **5**(9), 1–10 (2016)
25. Shah, N.: Pharmaceutical supply chains: key issues and strategies for optimisation. Comput. Chem. Eng. **28**, 929–941 (2004)
26. Snyder, L.V., Shen, Z.J.M.: Fundamentals of Supply Chain Theory. Wiley, Hoboken (2011)
27. Sweeney, E.: Towards a unified definition of supply chain management: the four fundamentals. Int. J. Appl. Logist. IJAL **2**(3), 30–48 (2011)
28. Szabo, N.: Formalizing and securing relationships on public networks. First Monday **2**(9) (1997). http://firstmonday.org/htbin/cgiwrap/bin/ojs/index.php/fm/article/view/548
29. Tian, F.: A supply chain traceability system for food safety based on HACCP, blockchain internet of things. In: ICSSSM, pp. 1–6 (2017)
30. Weber, I., Xu, X., Riveret, R., Governatori, G., Ponomarev, A., Mendling, J.: Untrusted business process monitoring and execution using blockchain. In: La Rosa, M., Loos, P., Pastor, O. (eds.) BPM 2016. LNCS, vol. 9850, pp. 329–347. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_19
31. Wolf, K.: Does my service have partners? Trans. Petri Nets Models Concurr. **2**, 152–171 (2009)
32. Wood, G.: Ethereum: a secure decentralised generalised transaction ledger (2018). https://ethereum.github.io/yellowpaper/paper.pdf
33. Wüst, K., Gervais, A.: Do you need a Blockchain? Technical report 2017/375 (2017). http://eprint.iacr.org/2017/375
34. Zardetto, D., Scannapieco, M., Catarci, T.: Effective automated object matching. In: ICDE, pp. 757–768. IEEE Computer Society (2010)

A Blockchain Architecture for Reducing the Bullwhip Effect

Sélinde van Engelenburg(✉), Marijn Janssen, and Bram Klievink

Faculty of Technology, Policy and Management, Delft University of Technology, Jaffalaan 5, Delft, The Netherlands
{S.H.vanEngelenburg,M.F.W.H.A.Janssen, A.J.Klievink}@tudelft.nl

Abstract. Supply chain management is hampered by a lack of information sharing among partners. Information is not shared as organizations in the supply chain do not have direct contact and/or do not want to share competitive and privacy sensitive information. In addition, companies are often part of multiple supply chains and trading partners vary over time. Blockchains are distributed ledgers in which all parties in a network can have access to data under certain conditions. Private blockchains can be used to support parties in making their demand data directly available to all other parties in their supply chain. These parties can use this data to improve their planning and reduce the bullwhip effect. However, the transparency that blockchain technology offers makes it more difficult to protect sensitive data. The dynamics between these properties are not well understood. In this paper, we design and evaluate a blockchain architecture to explore its feasibility for reducing information asymmetry, while at the same time protecting sensitive data. We found that blockchain technology can allow parties to balance their need for inventory management with their need for flexibility for changing partners. However, measures to protect sensitive data lead either to reduced information, or to reduced speed by which the information can be accessed.

Keywords: Blockchain · Blockchain technology · Supply chain management
Information sharing · Information asymmetry · Bullwhip effect
Distributed ledger

1 Introduction

The availability of timely information is crucial for operations in supply chains. If information lags behind the "Bullwhip Effect" (BWE) might occur [1, 2]. The BWE is the effect of the amplification of the demand in the supply chain when there is no good overview of the demand expected in the supply chain. Each node has only information and forecasting based on the demand of the next node, but not an overview of the end-customer demand. The more nodes are between the end-user demand and the supply chain party, the larger the bullwhip effect can be [3]. The final customer demand might be fairly even, but due to the use of batches when ordering, the demand

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 69–82, 2018.
https://doi.org/10.1007/978-3-319-94214-8_5

fluctuates. This results in having larger stocks than needed, excessive production, non-optimal scheduling of production and large warehouses. The risks of having stock are not only that it needs capital, but stock might also become obsolete.

Numerous studies show that addressing BWE requires better information sharing about the demand between supply chain partners [4]. The BWE effect can be avoided by overcoming information asymmetry in the supply chain. Forecasts are based on the information available. By collaboration and sharing the end-customer demand with all parties in the chain, each party will be able to make a more realistic planning of the use of their capacity and the orders that will be produced [5]. By looking at the end-customer demand, variations of forecasts caused by batch orders can be avoided.

Blockchain was originally conceptualized by Nakamoto [6] to store and share transactions of the cryptocurrency Bitcoin. Blockchain technology could be used to share other data instead of transactions, including the demand data that supply chain parties can use to avoid the BWE. Blockchain technology allows for distributed information sharing without requiring an intermediary. Blockchain technology can thus allow parties in a supply chain to directly share data with parties further downstream in the supply chain and thereby reduce information asymmetry. It is possible to limit the parties that can be part of the network and have reading rights, for instance in a private or consortium blockchain [7, 8]. Additionally, once data is stored in a blockchain, it becomes hard to change it and therefore it can be a trusted way to share information among supply chain partners. This makes blockchain an interesting candidate for supporting information sharing to reduce the BWE.

The straightforward access to data that blockchain technology offers has a downside as well. The demand data can be sensitive as sharing of customer data or demand data might harm the negotiation and competitive positions of companies. Sharing that data with a large number of parties, without protecting it, might harm businesses' interests. This might mean that businesses are not willing to share the data required to reduce the BWE via a blockchain, without a form of access control. One solution is to only include parties in the network that are in a specific supply chain. However, this would mean that if parties decide to change supply chains, they have to change blockchain networks as well. This reduces their flexibility and makes it harder to access the data from the new supply chain.

A balance between data accessibility and data protection is required to support information sharing to reduce data asymmetry in supply chains. Without accessibility, there is no reduction of information asymmetry. Without the protection of sensitive data, it is unlikely that businesses will be willing to share data [9]. However, the dynamics between the transparency offered by blockchain technology and the protection of sensitive data are not well understood. Therefore, it is unclear whether blockchain technology will in fact be feasible for reducing information asymmetry in supply chains. This work provides an initial exploration of these dynamics.

In the next section, we discuss the related work. In Sect. 3, we describe the research approach. In Sect. 4, we establish the requirements for an architecture that supports the sharing of demand data to reduce the BWE based on literature. Establishing these requirements helps to get deeper insight into the needs for data accessibility and data protection. In Sect. 5, we provide a design for a blockchain architecture in which we attempt to balance accessibility with the protection of data. Section 6 provides an illustration of the architecture. In Sect. 7, we provide an evaluation of the architecture. In this evaluation, we reflect on how well we could balance data accessibility with data protection and what are the difficulties in striking a balance.

2 Related Work

Research on using blockchain technology to store and share other types of data than transaction data is on the rise in a variety of domains. Blockchain is proposed to support different processes in the domains of supply chain management and business process management as well [10–18]. Examples are tracing goods throughout their lifecycle [10, 12], conflict resolution in supply chains [14], crowd lending [15], business-to-government information sharing [11] and supply chain integration [13].

The literature mentions various benefits of blockchain technology. Mentioned particularly often is that blockchain allows for transparency and traceability and thereby provides trust without requiring an intermediary party [10, 12, 13, 16, 18]. Examples of other benefits ascribed to blockchain technology are robustness by decentralization, practical immutability of stored data, anonymity of nodes, and high data integrity [10, 12, 13, 15, 16]. The disadvantages mentioned in literature are the difficulty of changing data once it is stored, scalability issues, privacy and confidentiality issues, the unclear legal status of smart contracts, and wasted resources [12, 14–16].

Access control and protecting sensitive data is a recurrent theme in the related literature on blockchain technology. The proposed blockchain designs often incorporate some form of access control. Usually this involves encryption to protect sensitive data or making the blockchain network private (see e.g., [10, 11, 13, 14, 16]). However, rarely, an analysis is made of the impact of the chosen form of access control on the perceived benefits of using blockchain technology. Yet, such impact is likely, especially as data protection is at odds with the transparency offered by blockchain technology. Insight into the dynamics between solutions for protecting sensitive data and data accessibility is required to establish whether blockchain technology can provide a balance that is suitable for a proposed application.

3 Research Approach

Blockchain technology relies on distributed ledgers, encryption, Merkle tree hashing and consensus protocols [19]. While these technologies themselves are not new, their specific combination in blockchain technology is. As discussed in the previous section, due to the novelty of blockchain technology, there is no existing knowledge on the dynamics between solutions for protecting sensitive data and data accessibility that we could use as a starting point for our investigations. Therefore, this work is of an explorative nature.

In this work, we design a blockchain architecture for the sharing of demand data in supply chains to support the reduction of the BWE. We focus on balancing access control with data accessibility in our design. The tensions between the need for access control and data accessibility are especially important in this domain, as on the one hand the demand data is sensitive. On the other hand, however, to make reduction of the BWE possible, high data accessibility is required.

As there is a clear tension between requirements for access control and data accessibility when sharing demand data in supply chains, it is interesting to look at the extent to which they can be met in a design based on blockchain technology. More specifically, the difficulties we come across when making design decisions and the extent to which we are able to strike a balance can provide an initial insight into the dynamics that play a role. For the evaluation of the architecture, we thus focus on the considerations taken into account in the design decisions and the extent to which we are able to provide the appropriate level of access control and transparency.

4 Requirements

The architecture for supporting information sharing to reduce the BWE, should ensure the accessibility of demand data and at the same time protect sensitive data. The demand data needs to be accessible, as this accessibility reduces information asymmetry. Protecting sensitive data is necessary, as businesses otherwise might not be willing to share [9]. To determine what the requirements are for reaching this objective, it needs to be known what type of information needs to be shared. Additionally, we need to know what data is sensitive to businesses and how it should be protected.

Requirement 1 for the architecture is that it should support the sharing of inventory levels, work in progress levels, order data and demand data. The BWE is prevalent in traditional supply chains in which parties can base their forecasts only on purchase orders from the previous party in the supply chain [5, 20, 21]. Various studies show that the BWE is reduced when additional information is shared [5, 21, 22]. Sharing market demand data reduces the BWE [5]. The BWE can be further mitigated when inventory levels and work in progress levels are shared as well [5]. This should be done in such a way that information asymmetry is diminished and that all parties in the supply chain can base their forecasts on the same data.

Requirement 2 is that demand data and inventory and work in progress levels should only be accessible to parties in the same supply chain and identities should be anonymized where possible. Demand data can be sensitive. First of all, the identity of parties (like customer names) can be sensitive when information is shared vertically, i.e., with other parties in the same supply chain upstream or downstream. A party that shares data about the identity of their buyer might be bypassed in the supply chain when their producer starts selling directly to their buyer [23, 24].

In addition, information can be shared horizontally, i.e. with parties at the same level in the same supply chain or with other supply chains. Many companies operate in multiple supply chains which complicates information sharing. Other businesses might be competitors that could use this data to their advantage or be used by others to approach customers.

Finally, because of competition law not all information can be shared as this might result in cartel formation. For instance, a business could have a strategy for dealing with their inventory that reduces costs. This might provide them with a competitive advantage. If another business has their inventory levels, they could learn from this, causing the competitive advantage to be reduced or lost. In addition, while it is beneficial for a business to share data to reduce the BWE vertically, there is no clear benefit of sharing it horizontally. Considering the risks, this should thus be avoided.

5 A Blockchain Architecture for Reducing the Bullwhip Effect

In this section, we present a blockchain architecture for diminishing the bullwhip effect. We focus on balancing the requirements in the previous section. We illustrate and evaluate the architecture in subsequent sections.

A blockchain is a distributed ledger in which data is stored in a series of blocks. Nodes in a blockchain network each have a copy of the blockchain [6]. New data that is added to the blockchain is distributed throughout the network [6]. It is then collected into blocks and added to the blockchain by linking the new block to the last block in the chain [6]. Parties in the network can accept the new block according to a consensus mechanism [6]. They express acceptance by adding new blocks on top [6].

The overall architecture presented here consists of several elements: (1) a blockchain network consisting of nodes operated by supply chain members, (2) a data architecture for the format of data in the blockchain, (3) a data sharing architecture for the sharing of data among supply chain partners, and (4) a data access architecture for providing supply chain partners access to certain data.

5.1 The Blockchain Network

The right to read, write or contribute to consensus of nodes can be restricted. The difference between open, consortium and private blockchains is that respectively everybody, a limited set of parties or one centralized organization can control the consensus process and write new data to the blockchain [7]. For all types, it is possible to have public reading rights [7]. However, for private blockchains, it is also a possibility to restrict who can be a node and read the data and thus make the reading rights private [7].

For our design, we are concerned with data accessibility and data protection. In other words, with who can read what data. As a simplification, consortium and private blockchains are sometimes both called "private" (see e.g., [7]). Since our main concern is who has reading rights, and not who controls the consensus process, we will use the term private for our blockchain in this sense as well.

In our design, we will limit what parties can be a node and have reading rights to the data in the blockchain. As we will further discuss in the evaluation, limiting access to the network to parties in a supply chain might be too restrictive. Therefore, in the design, businesses in a certain industry can be part of the network for that industry.

5.2 Data Architecture

In a blockchain, each block consists of a header and a body [6]. In the body, the actual data is stored. The header contains a Merkle root that is unique to the data that is stored in the body [6]. This means that if the data in the body is (maliciously) changed, the Merkle root does not match the data in the body anymore. Furthermore, the header contains a unique hash of the header of the previous block in the chain [6]. Thus, if the header of the previous block is changed, the hash of the header in the next block does not match anymore. Consequently, changing data that is stored in a block requires its header and the header of all subsequent blocks to be changed to avoid detection. This makes it harder to modify data that is stored in a blockchain.

In the case of our design, the data that should be in the body of the block is order data, market demand data, data on the inventory level and on the work in progress level of businesses. These types of data are different, as orders can be viewed as an interaction between businesses. Conversely, inventory and work in progress levels signify internal statuses of businesses.

To store an order in a block, at least the following data elements are required: (1) ID of the retailer, (2) ID of the supplier, (3) type of goods that are ordered, (4) the quantity of goods that are ordered, (5) the date at which the goods will be delivered, and (6) a unique order number. Purchases could be arranged in long-term contracts as well. Such a contract can be stored in a similar manner as a simple order, with added data elements with information from the contract that is relevant for forecasting, such as agreements on repetitive orders. Furthermore, the date that is stored for a contract will signify the end of the contract instead of the date of delivery.

To improve the reliability of the data both businesses can sign the order or contract with their private key. Just as in the case of the wallet of users in Bitcoin, the ID of the businesses can be used as a corresponding public key. Other parties can check whether the parties have indeed signed the agreement using this public key as part of the consensus mechanism (see Sect. 5.3).

Not all data in a purchase order or contract is required to be added as data elements for businesses to base their forecasts on. For instance, data on pricing does not seem to provide an additional benefit for reducing the BWE and is highly sensitive. Hence, such data should not be stored in the blockchain. However, businesses could benefit from storing a proof of existence of the full purchase order and contract in the blockchain. For such proof of existence, merely a unique hash of the document is stored and signed by parties and not the document itself. This can provide businesses with a proof of what was agreed upon, which might improve trust. This benefit could incite businesses to add their orders and contracts at an early stage, which allows other parties to have the data at an earlier stage and to start forecasting earlier in the process.

The inventory levels and work in progress levels can be added to the blockchain as well. For this, at least the following data elements need to be added: (1) ID of the business, and (2) inventory or work in progress level. The party that adds the data can sign it to signify that they added the data themselves.

The data that parties add to the blockchain can be encrypted, with the exception of the ID's of parties and the end-dates of contracts. The latter data elements are necessary to determine who is in what supply chain. This, in turn, is needed to determine what data is relevant to parties and whether they should have access. While not being encrypted, it is not necessary to link the actual businesses to an ID.

5.3 Data Sharing Architecture

Nakamoto [6] describes 6 steps for running a blockchain network. Some of these can be left out when other data than transactions are shared. Most notably, the steps necessary to provide proof of work might not be required [11].

Based on this, the sharing of data via the blockchain can be as follows:

1. A party collects data on orders, contracts and inventory and work in progress levels.
2. The party encrypts all data elements, except for their own ID's and the dates of the end of contracts.
3. The party and other parties involved sign the data.
4. The data is distributed throughout the network.
5. A node adds the data to a block and they add the block to the chain.
6. The new block is distributed throughout the network.
7. Parties check whether the data is actually signed by the appropriate parties using their ID's (public key).
8. If they accept the data, they add a new block on top.

5.4 Data Access Architecture

As data is encrypted in the design, the appropriate parties need to be able to decrypt it in order to access it. To obtain a key, a party has to request it from a key distribution component. This request should contain the ID of the party that shared the data via the blockchain and specify what data access is requested to. The key distribution component will determine whether the data requested is from a party downstream the goods flow of the party requesting the key in the same supply chain. Only if this is the case, it will provide a key. Data access is thus flexible and depends on the context of what parties are in a supply chain with each other at a certain moment.

Who is in the same supply chain depends on the contracts between parties. The current supply chain a business is in can be viewed as a chain of businesses that have contracts with each other that have not ended yet. All data to determine this is available without encryption, viz, the ID's of parties and the end dates in the contracts. The key distribution component should thus have its own copy of the ledger.

Figure 1 provides an example of businesses in an industry that are connected via contracts that have not ended yet. In this example, all businesses, except for retailer A2, are in the supply chain of business D. A2 is not in their supply chain, as there is no path of contracts from D to A2. This means that D can get access to data from all parties, except for A2. For instance, business B1 is downstream in the same supply chain of C1 and thus C1 can have access to data from B1. Businesses C2 and C3 are in the same supply chain, but not downstream from each other.

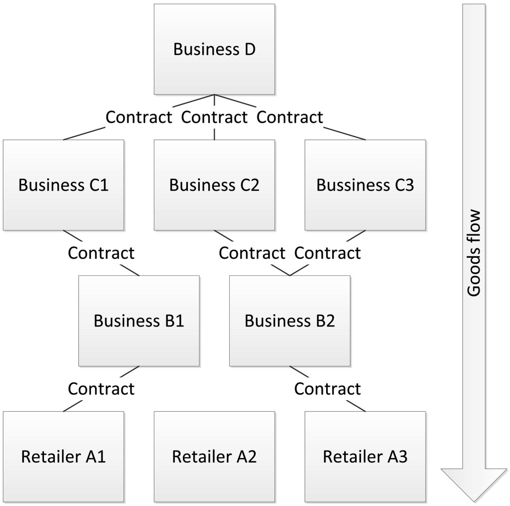


Fig. 1. Businesses in an industry connected via (still in force) contracts in supply chains

To determine who is in a supply chain downstream of a certain business, the key distribution component can do the following:

1. Determine the ID of the business.
2. Search the blockchain for contracts that have not ended yet where the business with this ID is the supplier.
3. List the IDs of the businesses that are purchasing goods in these contracts. These parties are in the supply chain.
4. Determine what parties are in the supply chain of the purchasers (induction).
5. Continue until you arrive at a set of businesses that do not have contracts.

6 Illustration

For the illustration, we consider a scenario with a typical user activity. In this scenario, business A manufactures cars. They use an audio system in their cars produced by business B. Business A has closed a contract with business B in which they agree to buy the audio systems from business B for a certain price. Business C is a retailer that actually sells the cars to consumers. They are in a contract with business A. Each of these parties is a node in the blockchain network via which they share data, as described in Sect. 5.1.

For the illustration we assume that the data about the contracts is already stored in the blockchain by business A. An activity diagram for the scenario is shown in Fig. 2. The storing of the contract by business A is left out here as well to improve clarity.

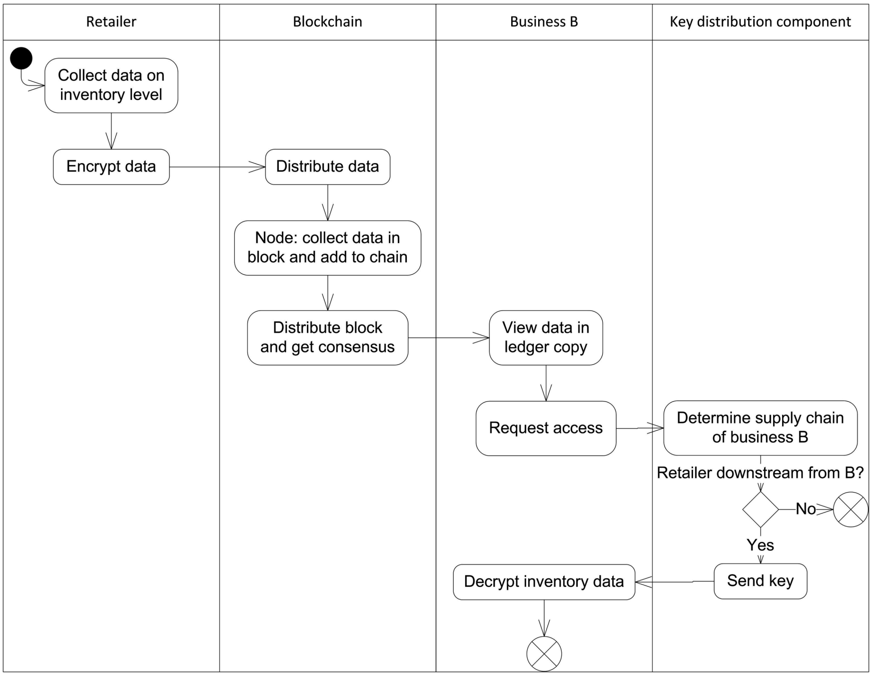


Fig. 2. UML activity diagram for the blockchain architecture

In the scenario, the retailer first collects new data about their inventory level and put it in the format described in Sect. 5.2. They then encrypt the data and sign it with their signature. Subsequently, they insert the data to the blockchain together with their ID. The inventory level data is then distributed throughout the blockchain network and shared via the steps provided in Sect. 5.3.

Business B receives the data in a new block in their copy of the blockchain. They want access to this data and they request access from the decision control component. The decision control component then tries to decide whether the retailer is in the supply chain downstream from business B according to the steps provided in Sect. 5.4. The retailer is indeed downstream in the supply chain from B. As the contracts are stored in the blockchain, the key distribution component can establish this. Then, the key distribution component provides a key to business B for decrypting the data on the inventory level. Business B can use the data to determine their inventory strategy.

7 Evaluation

In Sect. 4, we established two requirements for the architecture, (1) it should support the sharing of inventory levels, work in progress levels, order data and demand data such that all parties in the supply chain base their forecasts on the same data, and (2) that demand data and inventory and work in progress levels should only be accessible to parties in the same supply chain and identities should be anonymized where possible. Here we discuss the extent to which we were able to meet each of the requirements, and more importantly, what are the difficulties in balancing them. Based on this analysis, we derive what are the dynamics between data accessibility and data protection when using blockchain technology to reduce the BWE.

In the design, parties have access to the data that they require to reduce the bullwhip effect. Demand data, inventory levels and work in progress levels can be accessed via the blockchain from parties downstream. These are exactly the parties that the data should be obtained from. When a party downstream the supply chain adds new data, all parties upstream have the same level of access to it, regardless of the number of parties that are in between. This is a clear advantage of using a blockchain.

In addition, when parties enter into a new contract with businesses, they can add the contract to the blockchain. Based on this, they can determine who is downstream in their supply chain in the same way as the key distribution component does. This allows them to determine what data is relevant to them. In addition, as the end dates of contracts are stored in the blockchain, parties can determine when others leave their supply chain. Data from these parties is no longer relevant. The ability to determine what data is relevant, further improves accessibility. In this way the context is taken into account.

By sharing their data via the blockchain, parties know that it can be accessed by all parties upstream in their supply chain. This means that parties do not have to make information sharing arrangements with individual parties. This makes accessing the appropriate data easier as well.

There thus is an advantage to storing data about contracts in the blockchain. Another advantage of blockchain technology is that everybody can have equal access to the data necessary to establish what parties are in a contract, namely their IDs and the end date of the contract. In the architecture, the consensus mechanism is based on checking whether parties have signed the data. Once data is accepted, it is hard to change. This property might provide an incentive to store the contract information, possibly together with a signed proof of existence of the full contract. Namely, storage in the blockchain could provide businesses with a certain amount of proof that they are in fact in an agreement with another business. So, not only is it beneficial to store contract information in the blockchain to improve accessibility, the blockchain technology also provides an incentive to do so.

Thus, access is provided to the data required to diminish the BWE. However, the data is encrypted and a key should be obtained. This reduces the speed and ease with which the data can be accessed.

Parties downstream can choose not to get into long term contracts with others, but to only work with single orders or switch contracts often. In this way, they might block access to data from parties downstream. This might mean that parties that want to share the data and benefit from the data sharing cannot do so, due to this other party. This might impact the relationships between businesses. They depend on each other to make their data available and to benefit from sharing. Businesses could deal with this by looking at the duration of contracts in the supply chain of a business that they are considering getting into a contract with and taking this into account in their decision.

The easiest way to ensure that only parties in the same supply chain can access data, is by only allowing members of the same supply chain to be part of the blockchain network. However, if a party wants to change supply chains, this would mean that they need to change blockchains as well. Having to change from information sharing system can have a quite negative impact on accessibility of data as well.

On the other hand, purchasing flexibility is viewed by some businesses as part of their competition strategy and can be a reason that they still are involved in a traditional supply chain, despite the risk of BWE [20]. As such, supply chain partners change as old parties might disappear and new ones might enter the supply chain. A solution allowing for more flexibility is the one in the architecture, namely making the blockchain only available to parties in the same industry. This allows them to change supply chains, without having to change the system that they use to share data.

In the architecture, confidentiality is guarded by encrypting the data and only allowing parties downstream in the same supply chain access. These are exactly the only parties that should have access to protect the sensitive data according to requirement 2. The extent to which an appropriate level of protection is offered thus depends on the security of the key distribution component and the quality of the encryption.

In Sect. 4, we discussed that identities of businesses should be protected, to prohibit parties to be bypassed in the supply chain or to know each other's trading partners and agreements. In the design, businesses will know the ID of other businesses that they are in direct contact with, but they do not need to know who is behind the ID's of the other parties in their supply chain, or even the blockchain. To a certain extent, it might be possible to derive this from looking at the number of purchase orders, or contracts that certain IDs have with each other. This issue will be bigger when it is, for instance, known that there are only a couple of parties that make certain product in an industry. Conversely, when there is a high number of small businesses involved, it is much harder to determine what party is behind a certain ID.

In addition, if businesses frequently change their ID, accessing data for other parties becomes limited, or even impossible. The IDs are necessary to establish in what supply chains businesses are and thus what data is relevant to them and what data they should have access to. When a party changes ID, it cannot be established in what supply chain they are and their data cannot be accessed. For instance, when a party uses different IDs for a contract and for data on their inventory level, they make the data on the inventory level inaccessible for the party that they are in a contract with, as now it is not possible to establish that they are in the same supply chain. We thus cannot fully protect the identities of the parties in the supply chains without severely reducing accessibility.

In the end, there is not a single party that controls access to the data, but the access is controlled by all parties downstream in the goods flow from the party that wants access. A party can provide parties upstream with access by adding the data and their contracts. However, they might not know who the parties upstream are exactly and they cannot provide one party upstream with access, while not the other. This means that they need to trust businesses that they do not know to keep their data confidential.

8 Conclusions and Suggestions for Further Research

In this paper, we investigated the feasibility of blockchain technology for reducing the BWE. In an exploratory effort, we focused on balancing data accessibility with data protection. We first established the requirements for a blockchain architecture for reducing the BWE. We then used the design and evaluation of a blockchain architecture for reducing the BWE as an analytical tool to obtain the required insight.

We found that information sharing using a blockchain has some clear advantages when it comes to providing businesses with access to data. First of all, as data sharing is distributed, parties in a supply chain can have equal access to data from other parties, even when they are further downstream. Blockchain also allows for storing contract data that parties can use to establish what data is relevant to them, without intermediacy of others. There is a clear incentive for parties to store these contracts in a blockchain as well.

Blockchain thus can offer high transparency. However, in the supply chain management domain, it is of paramount importance to only provide access to data to the appropriate parties. We were unable to find a design in which all sensitive data was fully protected, in particular the IDs of the businesses. The reason for this is that the IDs are necessary to identify the data that is relevant to businesses and to arrange access control. Further research could focus on finding other strategies that do not require sharing the IDs of the businesses. In addition, the level of protection of the other sensitive data depends on the quality of the encryption used.

If businesses' data is not adequately protected, they could respond with strategies that reduce data accessibility, e.g., not sharing certain sensitive data at all or frequently changing IDs. In addition, protecting sensitive data requires that parties perform additional steps to get access. This reduces the speed by which data can be accessed.

The fundamental conflict between data accessibility and data protection seems to be magnified when using blockchain technology. To reduce information asymmetry and to benefit from the improved reliability everybody in a network should have equal access to data. Equal access for all parties is in direct conflict with providing different parties in the network with different levels of access. Currently, the way to solve this seems to be by either sharing some additional data not via the blockchain, such as keys or the actual contracts, or by making it only include parties that can have the same level of access. Both possibilities reduce data accessibility. Furthermore, both solutions require additional 1-on-1 connections outside of the blockchain or a third party intermediating after all to distribute keys or establish identities of parties. This could result in the usual disadvantages of intermediation and having various 1-on-1 connections that blockchain seems to avoid at first sight.

Further research is necessary to determine whether there are other solutions that do not harm data accessibility in this way and that avoid relying on additional connections and relying on third parties. In addition, further study is needed to determine what balance between data accessibility and data protection are acceptable to businesses. Additional practical insight might be gained as well by evaluating the architectures in practice.

References

1. Lee, H.L., Padmanabhan, V., Whang, S.: Information distortion in a supply chain: the bullwhip effect. Manag. Sci. **43**, 546–558 (1997)
2. Lee, H.L., Padmanabhan, V., Whang, S.: The bullwhip effect in supply chains. Sloan Manag. Rev. **38**, 93–102 (1997)
3. Fiala, P.: Information sharing in supply chains. Omega **33**, 419–423 (2005)
4. Bray, R.L., Mendelson, H.: Information transmission and the bullwhip effect: an empirical investigation. Manag. Sci. **58**, 860–875 (2012)
5. Cannella, S., Ciancimino, E.: On the bullwhip avoidance phase: supply chain collaboration and order smoothing. Int. J. Prod. Res. **48**, 6739–6776 (2010)
6. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System (2008). https://bitcoin.org/bitcoin.pdf
7. Buterin, V.: On public and private blockchains. https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/
8. Pilkington, M.: Blockchain technology: principles and applications. In: Olleros, F.X., Zhegu, M. (eds.) Research Handbook on Digital Transformations, pp. 227–253. Edward Elgar Publishing, Cheltenham (2016)
9. Fawcett, S.E., Osterhaus, P., Magnan, G.M., Brau, J.C., McCarter, M.W.: Information sharing and supply chain performance: the role of connectivity and willingness. Supply Chain Manag. Int. J. **12**, 358–368 (2007)
10. Abeyratne, S.: Blockchain ready manufacturing supply chain using distributed ledger. Int. J. Res. Eng. Technol. **5**, 1–10 (2016)
11. van Engelenburg, S., Janssen, M., Klievink, B.: Design of a software architecture supporting business-to-government information sharing to improve public safety and security. J. Intell. Inf. Syst. (2017)
12. Tian, F.: An agri-food supply chain traceability system for china based on RFID & blockchain technology. In: 2016 13th International Conference Service System and Service Management, pp. 1–6 (2016)
13. Korpela, K., Hallikas, J., Dahlberg, T.: Digital supply chain transformation toward blockchain integration. In: Proceedings of the 50th Hawaii International Conference on System Sciences, pp. 4182–4191 (2017)
14. Weber, I., Xu, X., Riveret, R., Governatori, G., Ponomarev, A., Mendling, J.: Untrusted business process monitoring and execution using blockchain. In: La Rosa, M., Loos, P., Pastor, O. (eds.) BPM 2016. LNCS, vol. 9850, pp. 329–347. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_19
15. Schweizer, A., Schlatt, V., Urbach, N., Fridgen, G.: Unchaining social businesses - blockchain as the basic technology of a crowdlending platform. In: 38th International Conference Information System, pp. 1–21 (2017)
16. Mendling, J., Weber, I., van der Aalst, W., vom Brocke, J., Cabanillas, C., Daniel, F., Debois, S., Di Ciccio, C., Dumas, M., Dustdar, S., Gal, A., Garcia-Banuelos, L., Governatori, G., Hull, R., La Rosa, M., Leopold, H., Leymann, F., Recker, J., Reichert, M., Reijers, H.A., Rinderle-Ma, S., Rogge-Solti, A., Rosemann, M., Schulte, S., Singh, M.P., Slaats, T., Staples, M., Weber, B., Weidlich, M., Weske, M., Xu, X., Zhu, L.: Blockchains for business process management - challenges and opportunities. arXiv:1704.03610, vol. 9, pp. 1–16 (2017)
17. López-Pintado, O., García-Bañuelos, L., Dumas, M., Weber, I.: Caterpillar: a blockchain-based business process management system. In: CEUR Workshop Proceedings, vol. 1920, pp. 1–5 (2017)

18. van der Aalst, W.M.P., De Masellis, R., Di Francescomarino, C., Ghidini, C.: Learning hybrid process models from events: process discovery without faking confidence. In: International Conference on Business Process Management (2017)
19. Tasca, P., Tessone, C.J.: Taxonomy of blockchain technologies. Principles of identification and classification. arXiv Preprint arXiv:1708.04872
20. Matthias, H., Stephen, D., Jan, H., Johanna, S.: Supply chain collaboration: making sense of the strategy continuum. Eur. Manag. J. **23**, 170–181 (2005)
21. Dejonckheere, J., Disney, S.M., Lambrecht, M.R., Towill, D.R.: The impact of information enrichment on the bullwhip effect in supply chains: a control engineering perspective. Eur. J. Oper. Res. **153**, 727–750 (2003)
22. Chatfield, D.C., Kim, J.G., Harrison, T.P., Hayya, J.C.: The bullwhip effect-impact of stochastic lead time, information quality, and information sharing: a simulation study. Prod. Oper. Manag. **13**, 340–353 (2004)
23. Klievink, B., van Stijn, E., Hesketh, D., Aldewereld, H., Overbeek, S., Heijmann, F., Tan, Y.-H.: Enhancing visibility in international supply chains: the data pipeline concept. Int. J. Electron. Gov. Res. **8**, 14–33 (2012)
24. van Stijn, E., Hesketh, D., Tan, Y.-H., Klievink, B., Overbeek, S., Heijmann, F., Pikart, M., Butterly, T.: The data pipeline. In: Global Trade Facilitation Conference 2011, pp. 27–32 (2011)

VR-BPMN: Visualizing BPMN Models in Virtual Reality

Roy Oberhauser(✉), Camil Pogolski, and Alexandre Matic

Computer Science Department, Aalen University, Aalen, Germany
{roy.oberhauser, camil.pogolski}@hs-aalen.de,
alexandre.matic@studmail.htw-aalen.de

Abstract. One impact of the digital transformation of industry is an increasing automation of business processes (BPs) and the accompanying need for business process modeling (BPM) and comprehension. The subsequent increased number of processes and process variants to cover all cases, their deeper integration with information services, and additional process structural complexity affects comprehensibility. While virtual reality (VR) has made inroads in other domains and become readily accessible as a graphical interface alternative, its potential for addressing upcoming BPM comprehension challenges has not been sufficiently explored. This paper contributes a solution for visualizing, navigating, interacting with, and annotating business process modeling notation (BPMN) models in VR. An implementation shows its feasibility and an empirical study evaluates the effectiveness, efficiency, and intuitiveness versus alternative model depiction modes.

Keywords: Virtual reality · Business process models · Visualization BPMN

1 Introduction

The digital transformation sweeping through society affects businesses everywhere, resulting in an increased emphasis on business agility and automation. Business processes (BPs) or workflows are one significant automation area, evidenced by the $2.7 billion market for Business Process Management Systems (BPMS) (Gartner 2015). Business process modeling (BPM) is commonly supported with the standardized notation BPMN (Business Process Model and Notation) (OMG 2011) also known as BPMN2. A primary objective of BPMN is to provide a comprehensible process notation to all business stakeholders while providing complex process execution semantics. Furthermore, BPMS vendor lock-in is avoided in that organizations can retain their intellectual assets in the form of models across vendors and into the future, while the vendors can focus their efforts on a single common notation. Yet despite BPMN's goal for comprehensibility, with a growing number of processes and process variants that attempt to cover more business process cases, their deeper integration with information services, and additional process structural complexity, the comprehensibility of the ensuing BP models for all stakeholders can be negatively impacted.

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 83–97, 2018.
https://doi.org/10.1007/978-3-319-94214-8_6

Contemporaneously with this BPM automation trend, virtual reality (VR) has made inroads in various domains and become readily accessible as hardware prices have dropped and capabilities improved. Moreover, the VR (mobile, standalone, console, and PC) revenues of $2.7bn (2016) are forecasted to reach $25bn by 2021 (Merel 2017) or $15bn by 2022 with an installed base of 50 m–60 m (Digi-Capital 2018). VR is defined as a "real or simulated environment in which the perceiver experiences telepresence" (Steuer 1992), a mediated visual environment which is created and then experienced. However, the potential to leverage such an immersive VR capability for comprehending and annotating BPMN models has been insufficiently explored. As BPMN models grow in complexity and deeper integration to reflect the business and IT (information technology) reality, and additional information and annotations are needed to help understand them, an immersive BPM environment could provide the visualization capability to still see the "big picture" for structurally and hierarchically complex and interconnected diagrams and provide a motivational boost due to its immersive feeling and the visual rendering of a BP into a 3D space that can be viewed from different perspectives.

This paper contributes a solution concept which we call VR-BPMN for visualizing, navigating, interacting with, and annotating BPMN models in VR. Capabilities include teleportation and fly-through navigation, depicting subprocesses using stacked hyperplanes, drawing annotative associations between BPMN elements, coloring model elements, and textual element tagging with mixed reality (MR) keyboard support. The evaluation investigates its effectiveness, efficiency, and intuitiveness versus alternative BPMN model depiction modes (paper and PC). We assume the models involved have already been constructed to represent some business reality and thus in this paper we do not delve into details on how the models were created or what business reality goal some model interaction intends to achieve, but rather how the models are visualized and comprehended in VR.

The remainder of this paper is structured as follows: Sect. 2 discusses related work. In Sect. 3 the VR-BPMN solution concept is described. Section 4 then provides details on the prototype implementation. The evaluation is described in Sect. 5 and a conclusion follows in Sect. 6. An Appendix contains various figures.

2 Related Work

Work related to VR-BPMN includes the process visualization and virtualization areas. As to process visualization techniques, (Du et al. 2012) provide a survey, concluding that 3D can improve the layout and can increase the information content of process models. Work related to process visualization includes (Betz et al. 2008), who described an approach for 3D representation of business process models based on Petri nets with organizational models, showing that a 3D process representation can facilitate process-specific information access. (Hipp et al. 2015) described and empirically evaluated various business process visualization alternatives: bubbles, BPMN3D that maps data objects associated to a BPMN element to a "third dimension", network, and thin lines; however, no 3D space nor an implementation is described. (Emens et al. 2016) created a dynamic business process visualization prototype in 2D that can be run in a web browser. In the area of domain-centric process automation, (Westner and Hermann 2017) focus on a virtual training and round-trip engineering scenario using VRfx. BPMN was not used and each 3D object was manually placed. (Holzmüller-Laue et al. 2013) is a web-browser based visualization focused on life science workflows that combines a business process execution simulator based on BPMN, task visualization of operations, and a lab robot simulation.

With regard to virtual worlds, (Brown et al. 2011) investigated collaborative process modeling and communication, implementing a 3D BPMN modeling environment in the virtual world Second Life, and also used the Open Simulator (Brown 2010). Extending this with augmented reality (AR), (Poppe et al. 2011) involves a collaborative virtual environment in Second Life for collaborative process sketching where remote participants are represented as avatars as well as the process in BPMN is projected onto a real space. The 3D Flight Navigator (Effinger 2013) was implemented in Java with OpenGL, and projects parts of BPMN collaboration diagrams onto fixed hyperplanes and provides a heads-up display for navigating the process. As to conferences, neither BPM 2017 in Spain nor BPM 2016 in Brazil shows paper topics directly related to VR or virtual reality. No major BPMS vendors currently sell VR variants of their products. These older studies and the lack of current research involving currently available VR capabilities is needed to determine what value VR can or cannot add.

In contrast to the above, VR-BPMN provides a VR-centric BPMN visualization that utilizes a standard game engine (Unity) and common off-the-shelf VR hardware (HTC Vive), includes comprehensive BPMN support including automatic layout and stacked 3D hyperplanes for subprocesses, visual annotation capabilities (association, coloring, tagging), and MR keyboard interface support.

3 Solution Concept

BPMN models consist of Business Process Diagrams (BPDs) that are composed of graphical elements consisting of flow objects, connecting objects, swim lanes, and artifacts (OMG 2011). Our VR-BPMN solution concept (Fig. 1) focuses on four primary aspects that are affected by VR:

(1) *Visualization*. Since the graphical elements are only specified in 2D, it is thus unclear exactly how these should or could be visualized or mapped into 3D space. While many visual options and metaphors are possible, our view is that diverging too far from the specification would reduce the recognition and standardization afforded by the BPMN specification. We thus utilize different 3D block-like shapes with sides for differentiating elements by type, while retaining the standard BPMN symbols which are fixated onto the sides of those 3D elements to permit perception from different angles. One challenge in 3D space in contrast to 2D space is that one can never be sure if there is not an element hidden behind another element at any particular vantage point if the element is opaque. If one makes the element partially transparent, then it can become confusing as to which element one is focusing on. We thus chose to make the elements opaque in order

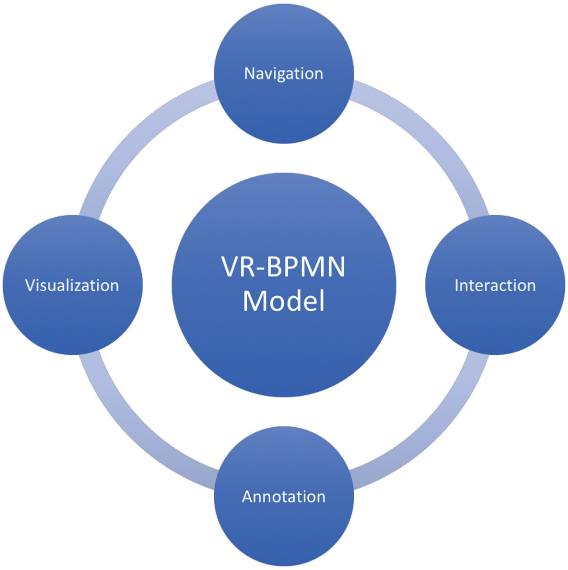


Fig. 1. The VR-BPMN solution concept.

to avoid this visual confusion, and by briefly adjusting one's perspective one can visually check that nothing is "hiding" behind an element. Additionally, visualizing text is an issue in VR due to the relatively low resolutions currently available and the distance to the text. Also, labels for BPMN elements can differ widely in length yet should not interfere with understanding the BPD structure. We thus place labels above the elements (like billboards), make them partially transparent in order not to completely hide any elements behind the label, the labels automatically rotate towards the camera to improve legibility from various angles, and for dealing with longer labels we constrain the maximum billboard width (to reduce overlapping), raise the billboard height, and reduce the text font size. For visualizing subprocesses, hyperplanes are used to take advantage of the 3D space with the subprocess projected onto a plane beneath its superprocess and connected via a glass pyramid to its superprocess. This spatial layout choice is somewhat intuitive since the conceptual relationship of sub and super can be mapped to a 3D space with above and below. However, for certain use cases when comparing subprocess and superprocess elements and their flows in detail, users may wish to have them in the spatially vicinity, so we allow the user to toggle the pyramid height on or off which raises the subprocess to the superprocess plane level.

(2) *Navigation*. The immersion afforded by VR requires addressing how to intuitively navigate the space while reducing the likelihood of potential VR sickness symptoms. Two modes are supported: teleporting permits a user to select a destination and have be instantly placed there (by moving the camera there) - while perhaps disconcerting, it may reduce the likelihood of the VR sickness that can occur when moving through a virtual space. Alternatively, a birds-eye view with

gliding controls is provided, enabling users to fly through the VR space and get an overview of the entire model. With a button on a controller, one can switch between the birds-eye-view and the teleport mode.

(3) *Interactions*. The BPMN specification does not specify exactly how users are to interact and interface with graphical BPMN visual elements. In our VR concept, user-element interaction is done primarily via the VR controllers supplemented by a MR keyboard. We use drag-and-drop element interaction for placing a connective annotation. Rather than only having access to a virtual keyboard that requires cumbersome pointing at each letter and clicking with the controllers, an *MR keyboard* provides access to a real keyboard that can be used for text input by projecting the webcam video stream onto a plane.

(4) *Annotations*. As one is immersed in VR with the headset on, ready access to other external sources of information is hindered and removing the VR headset disrupts the immersion experience. Annotations (a type of BPMN Artifact) can assist users by providing missing information and for comprehending larger or more complex models with notes or details not readily apparent in the model. Thus, we place an increased value in Annotations for placing additional information into the context of the model solution concept. In the BPMN specification textual annotations are signified with a square left bracket floating somewhere apart from any element and an association as a dotted line can be drawn to an element. This unnecessarily clutters the virtual space and dotted lines are more difficult to detect since other objects can be behind them. Hence, our alternative concept of *tagging* enables textual annotations to be placed on any element (including swimlanes or a plane), with the tags situated directly above the element label. Moreover, tagged elements have a colored ribbon indicator on top of their label and the color can be used to indicate some property (such as importance, user, etc.). Tag visibility can be toggled on or off in order to reduce textual clutter if tags undesired. Furthermore, an *association annotation* permits elements to be visually associated via a colored dotted line to be easily visually discernable (by default fluorescent green) and color choice permits a persistent visual differentiation and grouping of elements from others. An association is placed via drag-and-drop, taking one element and dropping it on another and the association itself can also be tagged in addition to the elements, it automatically avoids collisions with objects between the associated elements, and it can extend across hyperplanes.

4 Realization

The Unity game engine was chosen for VR visualization due to its multi-platform support, direct VR integration, popularity, and cost. Blender was used to develop the visual BPMN model elements. For testing with VR hardware, we used the HTC Vive, a room scale VR set with a head-mounted display with an integrated camera and two wireless handheld controllers tracked using two ‘Lighthouse’ base stations.

Visualization. Common BPMN2 elements were realized with various 3D shapes, with the BPMN symbol in black and white placed on the sides in order to be readily perceived (see Fig. 2). Elements are labeled with white text on a semitransparent dark background (rather than white) so as not to detract too much from the primarily white-based BPMN element symbols. Element layout placement is scaled based on the provided BPMN XML layout attributes. Subprocesses as depicted as stacked hyperplanes connected via colored semi-transparent pyramids to their superprocess, as shown in Fig. 3.

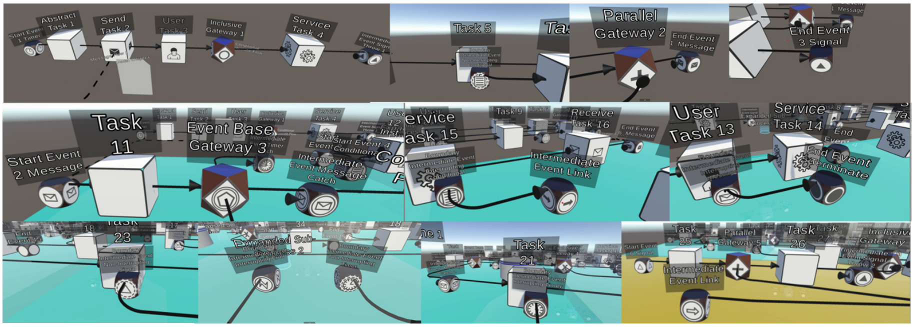


Fig. 2. Collage of various VR-BPMN BPMN2 element screenshots.

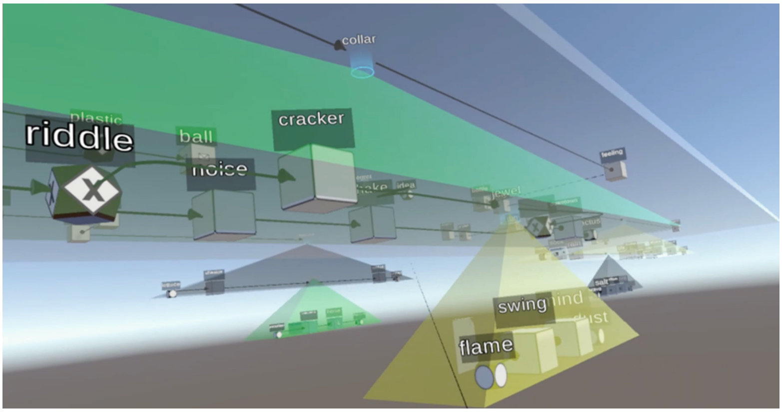


Fig. 3. Layered hyperplanes showing subprocesses as semi-transparent colored pyramids for an obfuscated process example. (Color figure online)

Navigation. With the right menu button on the right controller, one can switch between birds-eye-view and the teleport mode (see Fig. 4). The teleport mode is controlled by the right trackpad by aiming at the target location and pressing the trackpad. The birds-eye-view works with both trackpads, with the left trackpad controlling altitude und the right one forward, backward, left, and right movement. Via the left menu button, a small version of the BPMN diagram, similar to a minimap, is shown to quickly identify the overall location.

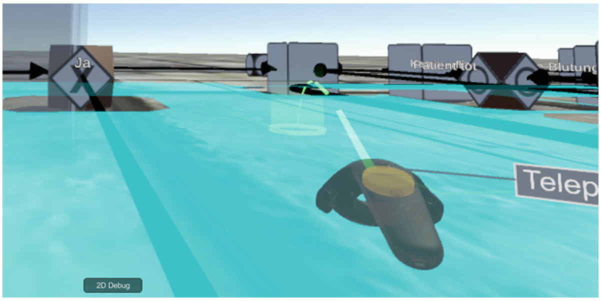


Fig. 4. Teleport interaction example showing VR controller and chosen destination highlighted as green cylinder. (Color figure online)

Interaction. Interaction is done primarily via the VR controllers. By pointing with the controller on an object with a tag and pressing and releasing the trigger button, tag visibility can be toggled on or off. By pointing at an object and pressing and holding the trigger, one can drag-and-drop the object for instance onto another object to place an association annotation. Text input is currently used for adding tags to the elements of a model. Rather than only having access to a virtual keyboard that requires point and click with the controllers, an *MR keyboard* provides access to a real keyboard that can be used for text input by projecting the webcam video stream onto an object's material (see Fig. 5).

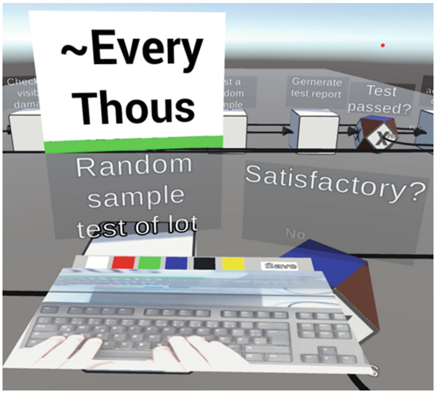


Fig. 5. Tagging in MR mode with tag color palette and real keyboard visible. (Color figure online)

Annotations. By default, all annotation types are given a fluorescent green color to differentiate them clearly from the actual model, but the color can then be customized. Users also can add an annotative association between any elements (see Fig. 6) of the model or even connect several processes. This is done by dragging and dropping from one element to another via a VR controller. Tagging permits users to annotate a selected BPMN element with any additional textual information, the font size is adjusted automatically to fit the available space. A colored ribbon on top of a label indicates that one or more tags exist, and the color is chosen when the tag is placed using a color palette (Fig. 5). Tag visibility can be toggled by selecting an element label. When tags are visible, these are placed on top of the labels on an opaque white background with black text to easily differentiate them from the labels below (Fig. 5). To color either an element ("Compare contents with order" element red in Fig. 7) or tag (Fig. 7 shows yellow, blue, and green tags) the user point with a VR Controller on the target to be colored and selects a color from a predefined color palette (Fig. 5). Figure 8 shows colored swimlanes.

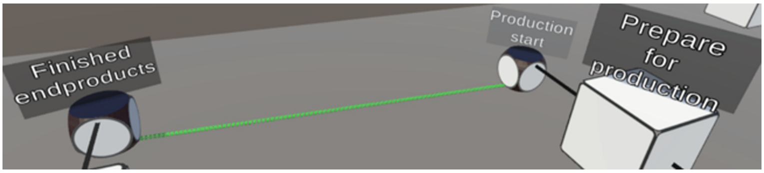


Fig. 6. User added connection annotation (green) between two processes. (Color figure online)

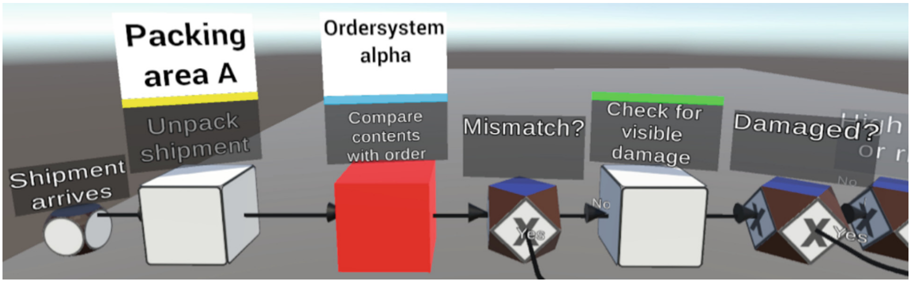


Fig. 7. Tags have a white opaque background and different tag ribbon colors; "Check for visible damage" has a hidden tag; "Compare contents with order" shows a colorized element. (Color figure online)

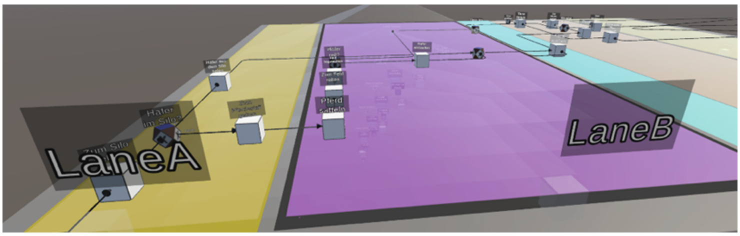


Fig. 8. Overview of a model in the birds-eye-view, showing lanes in different colors. (Color figure online)

5 Evaluation

The VR-BPMN prototype validated the feasibility of our VR-BPMN solution concept. Our empirical evaluation then investigated to what degree BPMN-based process analysis, comprehension, and interaction are affected by a VR environment. For this, we compare VR-BPMN with two other commonly available visual depiction modes: (1) paper-based BPMN analysis and (2) a common PC-based BPMN modeling application tool (Camunda Modeler). To avoid a bias for certain BPM paradigms or tools due to previous experience (such as experienced professionals), we used a convenience sample of master students with little to no prior BPMN experience. The experiments were supervised, a brief training was provided including BPMN anti-patterns, and a debriefing followed. In order not to skew process comprehension due to subject foreign language competency differences, BPMN labels were mostly in the native language of the subjects (German). To avoid interfering with the VR environment and skewing task durations due to subjects reading instructions or answering questions via VR interaction or requiring VR glasses removal, all task questions were asked and answered verbally and noted by a supervisor.

5.1 Paper-Based BPMN vs. VR-BPMN

To compare the comprehension of BPMN processes printed on paper vs. VR-BPMN, a convenience sample of eight Computer Science (CS) students was selected. Four fictional processes were grouped in process pairs, each pair being equivalent in structural complexity but differing in domain nomenclature and slightly in structure in order to avoid familiarity from affecting efficiency, while maintaining equivalent complexity to reduce the treatment variables. The Emergency Patient Treatment process was paired with the Farm process, both consisting of 5 subprocesses, 24 activities, and 6 gateways. An Invoice process (5 activities, 2 gateways, 2 endpoints) was paired with the Mario Game process (5 activities, 4 gateways, 1 endpoint). Whichever process was used on paper was paired with the equivalent in VR-BPMN. To avoid an ordering bias, half started initially with paper. No subject saw the exact same process twice. The task given was to analyze and explain the process (walkthrough).

Figure 9 shows task durations for VR-BPMN vs. paper sorted by the shortest VR times, with the mode order given in brackets. VR-BPMN average task duration was 5:07 vs. 3:36 for paper, a 42% difference. We note that, except for subject 5, the second mode (VR or paper) was faster than the first.

5.2 Tool-Based BPMN vs. VR-BPMN

To evaluate both interaction and process comprehension in VR-BPMN (V) versus a representative common PC-based BPMN modeling tool (Camunda Modeler) (C), a convenience sample of seven Computer Science (CS) students was used. One student experienced VR sickness symptoms and the associated measurements were excluded. The order of which tool to start with first was randomly selected, with three starting with V and three with C. Two BPMN processes were used, the first was a “Student Exam BPM” in German consisting of 5 processes, 18 activities, 5 start events, 7 end

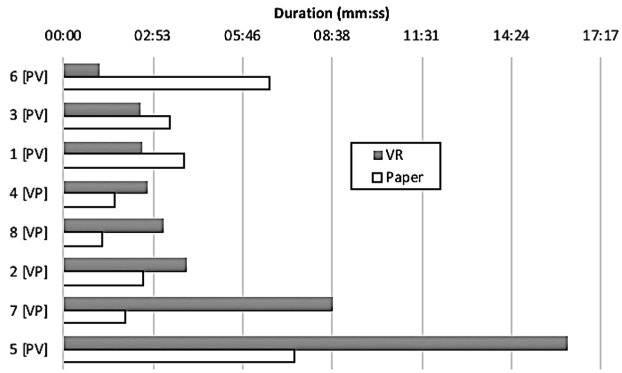


Fig. 9. Subject task duration for paper BPMN model vs. VR-BPMN sorted by shortest VR duration; order given in brackets, V = VR, P = paper.

events, and 6 gateways and is shown in VR-BPMN in Figs. 11 and 12 in the Appendix, the second was a quality-related production processes model in English shown in Fig. 13 and in VR-BPMN in Fig. 14 in the Appendix. Four equivalent tasks were timed for each tool. The tasks were:

(1) Student Exam model: find all BPMN modeling errors,
(2) Quality model: connect all elements that deal with testing,
(3) Student Exam model: Connect the end nodes with the appropriate start nodes, and
(4) Quality model: which tasks must be completed before "Fill out checklist and complaint form."

Task 1's duration was divided by the number of errors found; for Task 2 and 3 the total duration was divided by the the number of connections made (since making these annotations requires extra tool interaction). Results are shown in Table 1, with the expected numbers shown in column 2. Since the domains and processes were unexplained and open to interpretation, the number of connections differed. All Task 1 and 4 elements were correctly identified, so comprehension effectiveness was equivalent. The average of per errror and per connection durations for all four tasks in V was 254s vs. 319s for C, making V 21% faster. Since errors might be clustered in the same process area in Task 1, omitting Task 1 from the sum yields 162s for V vs. 187s for C, making V 14% faster. Five of the subjects indicated a preference for Camunda over VR-BPMN.

Figure 10 shows the total task durations for V vs. C sorted by the shortest V times, with the mode order given in brackets. We note that the second mode is almost always faster than the first, except for subjects 4 and 2 whose duration differences between both modes are minor (17% and 14% respectively) relative to the others' differences. This could be due to a cold-start effect, and in future work we intend to provide a warm-up task in each mode before timing a task.

Table 1. VR- BPM (V) vs. Common BPMN Tool (C) durations in seconds for the four tasks.

Task	Subject	1	2	3	4	5	6	Average
V1	Expect:2	2	2	2	2	2	2	
	Duration (s/error)	38	59	159	26	129	138	92
V2	Expect:5	4	4	8	8	10	7	
	Duration (s/connection)	35	49	43	14	33	35	35
V3	Expect:5	4	4	5	5	4	6	
	Duration (s/connection)	62	54	85	65	79	31	63
V4	Duration (s)	67	72	60	84	39	62	64
V Total (s)								254
C1	Expect:1	1	1	1	1	1	1	
	Duration (s/error)	93	123	118	74	218	163	132
C2	Expect:5	4	4	11	10	9	5	
	Duration (s/connection)	35	23	14	10	10	87	30
C3	Expect:5	4	5	5	6	6	5	
	Duration (s/connection)	30	56	31	30	32	48	38
C4	Duration (s)	43	34	40	140	356	102	119
C Total (s)								319

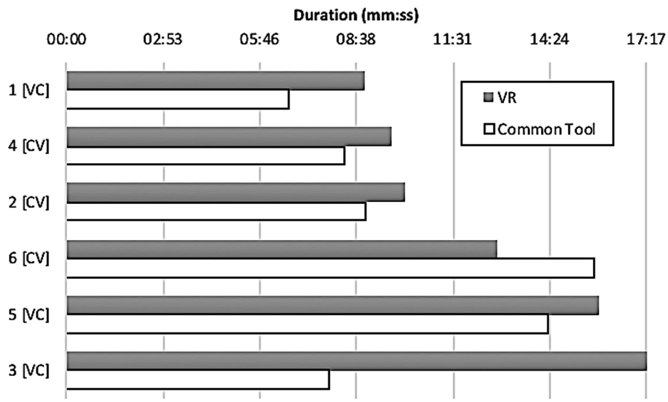


Fig. 10. Subject task duration for a common BPMN tool vs. VR-BPMN sorted by shortest VR duration; tool order given in brackets, V = VR, C = common BPMN tool.

5.3 Discussion

Since BPMN intends to be comprehensible for all stakeholders, and to reduce preexisting personal preferences and biases of BPMN professionals from biasing the results, we believe the use of novices for this experiment was appropriate. Although the sample size was small and did not consist of BPMN professionals, these measurements can provide indicators for future investigation.

The effectiveness of VR-BPMN was shown to be equivalent to the paper and PC tool modes, indicating that all tasks could be performed and the processes analyzed and comprehended.

As to efficiency, VR was 21% faster when taken on a per-error and per-connection standpoint for tasks vs. a common BPMN tool. VR was 14% faster if the error-finding Task 1 is omitted (leaving annotations and comprehension). For the paper mode, VR exhibited a 42% overhead, yet we note that the second mode (VR or paper) was always faster than the first (except for subject 5. The use of VR is more complex than paper, yet with additional VR training, we believe the apparent VR efficiency overheads could be reduced. Since the second mode was often faster than the first, we conjecture that the cognitive burden to become focused on the BPMN task and context (to be cognitively "in flow"), irrespective of the mode, are affecting the durations particularly in the initial mode and can be viewed as a type of overhead. It is likely unique and dependent on the subject's current mental alertness and motivation. It may be analogous to taking an exam, where efficiency increases after one or more questions have been answered. Thus, an efficiency comparison with VR based purely on the total durations can be misleading. Although we did include a training, in future work we plan to insert an additional warm-up round after switching modes to acclimate users within each environment before giving them actual timed tasks. 1 subject was affected by VR sickness, and we will explore possible improvements.

Across all 14 subjects, the intuitiveness of the VR-BPMN interface was rated a 4.2 on a scale of 1 to 5 (5 best). However, a preference for the BPMN tool was indicated by 5 of the 6 subjects. Users tend to prefer what they are familiar to, especially when they don't have much exposure time to the new option. We surmise that VR interaction is not as yet a fundamental competency compared to mouse or paper use, and with further VR experience user expectations may adjust. As VR applications adopt standard VR interaction patterns and expectations (such as button functionality becoming standardized in the VR market like the mouse left and right button), VR use will become more intuitive and natural for users, reducing current overheads. Comments by subjects included that VR model clarity was affected by hidden objects. We thus recommend that BPMN models intended for VR have large spacing between elements to reduce this issue. Further, they complained that the low resolution on current VR-headsets make reading text labels more difficult in VR vs. on 2D monitors or paper. But they also commented that using VR-BPMN was fun, and we conjecture that it can provide a motivational factor towards comprehension.

6 Conclusion

This paper contributed a solution concept called VR-BPMN for bringing BPMN models into VR to provide an immersive BPD experience, and addresses the visualization, navigation, interaction, and annotation aspects affected by VR. A prototype based on the Unity game engine und using the HTC Vive demonstrated its feasibility. Our empirical evaluation showed that its effectiveness for process comprehension in VR was equivalent to non-VR modes (paper and 2D tools). For efficiency, VR-BPMN was 14–21% more efficient than a common PC BPMN tool on a per-connection and per-error basis, yet VR-BPMN was 42% slower than paper use for equivalent tasks. While VR-BPMN was found to be quite intuitive (4.2 out of 5), nevertheless 83% preferred the common PC tool when given a choice of one or the other; hindrances in VR included text legibility and hidden objects in 3D space.

While our results showed that the VR-BPMN solution concept is feasible, that our interaction interfaces are intuitive, and that it can be as effective and efficient as BPMN tools, for broader attractiveness additional capabilities are needed. Future work includes addressing VR sickness, improving the clarity of the models (especially text), integrating additional interactive and informational capabilities, as well as a comprehensive empirical study using BPMN professionals.

Acknowledgements. The authors would like to thank Peter Bolek, Andreas Friedel, Sabine Hager, Matthias Nutz, Philipp-Daniel Wendt for their assistance with the design, implementation, and evaluation.

Appendix

See Figs. 11, 12, 13 and 14.

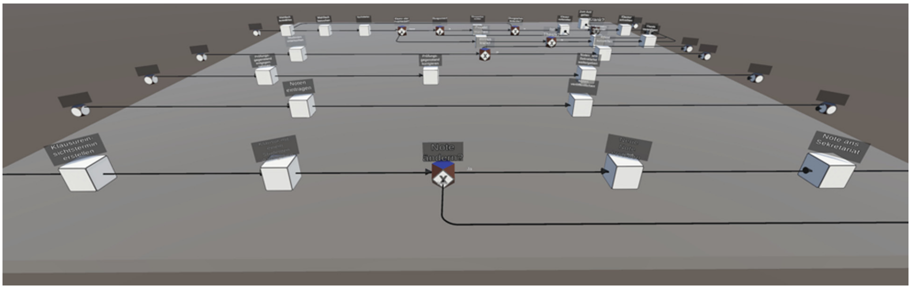


Fig. 11. The Student Exam processes shown in VR-BPMN.

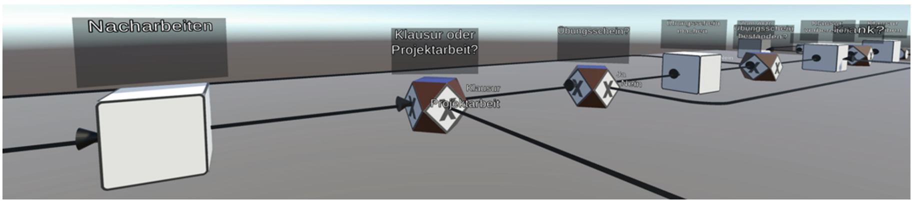


Fig. 12. Close up of Student Exam process in VR-BPMN.

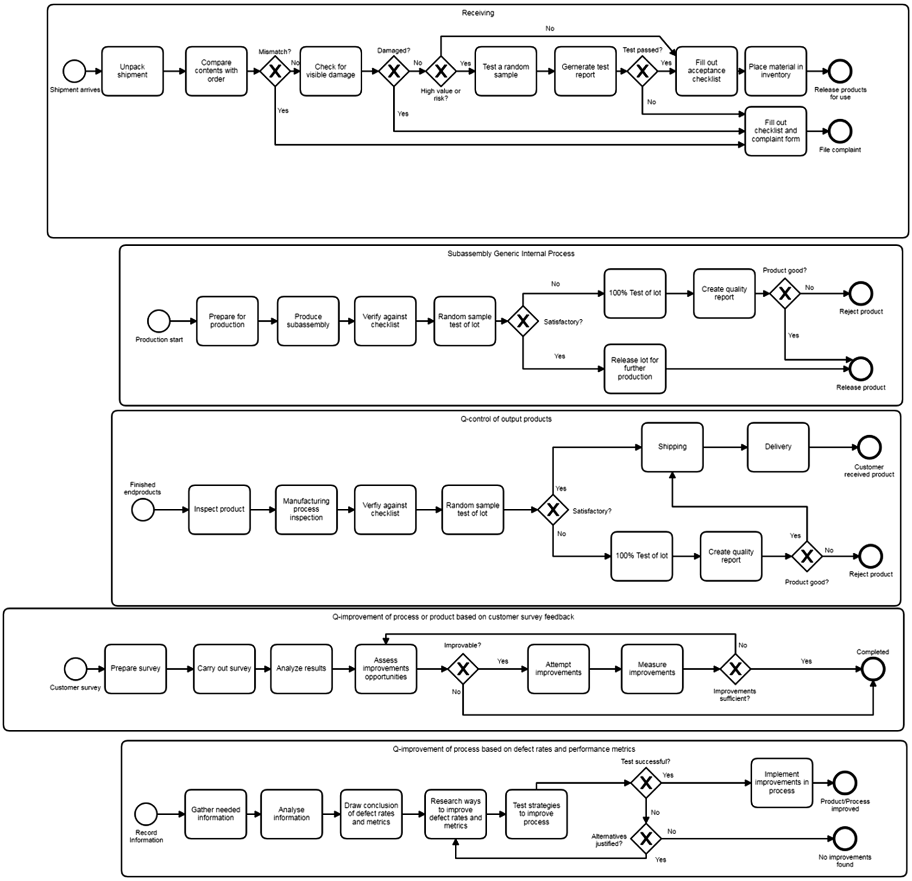


Fig. 13. The Quality Production Processes model as BPMN.

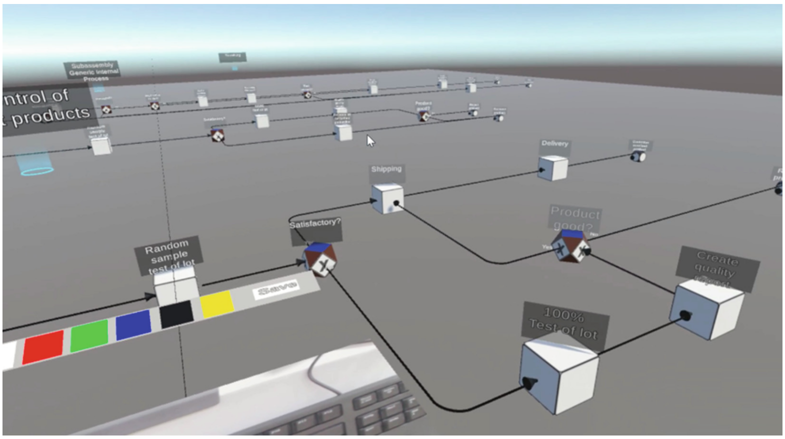


Fig. 14. The Quality Production Processes model in VR-BPMN.

References

Betz, S., Eichhorn, D., Hickl, S., Klink, S., Koschmider, A., Li, Y., Oberweis, A., Trunko, R.: 3D representation of business process models. In: MobIS, pp. 73–87 (2008)

Brown, R.A.: Conceptual modelling in 3D virtual worlds for process communication. In: Proceedings of the Seventh Asia-Pacific Conference on Conceptual Modelling, vol. 110, pp. 25–32. Australian Computer Society, Inc. (2010)

Brown, R., Recker, J., West, S.: Using virtual worlds for collaborative business process modeling. Bus. Process Manag. J. **17**(3), 546–564 (2011)

Digi-Capital: Ubiquitous $90 billion AR to dominate focused $15 billion VR by 2022 (2018). https://www.digi-capital.com/news/2018/01/ubiquitous-90-billion-ar-to-dominate-focused-15-billion-vr-by-2022/#more-2191. Accessed 07 Mar 2018

Du, X., Gu, C., Zhu, N.: A survey of business process simulation visualization. In: International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE), pp. 43–48. IEEE (2012)

Effinger, P.: A 3D-navigator for business process models. In: La Rosa, M., Soffer, P. (eds.) BPM 2012. LNBIP, vol. 132, pp. 737–743. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36285-9_74

Emens, R., Vanderfeesten, I., Reijers, H.A.: The dynamic visualization of business process models: a prototype and evaluation. In: Reichert, M., Reijers, H.A. (eds.) BPM 2015. LNBIP, vol. 256, pp. 559–570. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42887-1_45

Gartner: Gartner Says Spending on Business Process Management Suites to Reach $2.7 Billion in 2015 as Organizations Digitalize Processes (2015). https://www.gartner.com/newsroom/id/3064717. Accessed 07 Mar 2018

Hipp, M., Strauss, A., Michelberger, B., Mutschler, B., Reichert, M.: Enabling a user-friendly visualization of business process models. In: Fournier, F., Mendling, J. (eds.) BPM 2014. LNBIP, vol. 202, pp. 395–407. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-15895-2_33

Holzmüller-Laue, S., Schubert, P., Göde, B., Thurow, K.: Visual simulation for the BPM-based process automation. In: Kobyliński, A., Sobczak, A. (eds.) BIR 2013. LNBIP, vol. 158, pp. 48–62. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40823-6_5

OMG: Business Process Model and Notation (BPMN) Version 2.0. OMG (2011)

Poppe, E., Brown, R.A., Recker, J.C., Johnson, D.M.: A prototype augmented reality collaborative process modelling tool. In: 9th International Conference on Business Process Management, pp. 1–5 (2011)

Steuer, J.: Defining virtual reality: dimensions determining telepresence. J. Commun. **42**(4), 73–93 (1992)

Merel, T.: The reality of VR/AR growth (2017). http://techcrunch.com/2017/01/11/the-reality-of-vrar-growth/. Accessed 07 Mar 2018

Westner, P., Hermann, S.: VR|ServE: a software toolset for service engineering using virtual reality. In: Sawatani, Y., Spohrer, J., Kwan, S., Takenaka, T. (eds.) Serviceology for Smart Service System, pp. 237–244. Springer, Tokyo (2017). https://doi.org/10.1007/978-4-431-56074-6_26

Process Modeling Within Augmented Reality

The Bidirectional Interplay of Two Worlds

Marcus Grum(✉) and Norbert Gronau

University of Potsdam, 14482 Potsdam, Germany
mgrum@lswi.de

Abstract. The collaboration during the modeling process is uncomfortable and characterized by various limitations. Faced with the successful transfer of first process modeling languages to the augmented world, non-transparent processes can be visualized in a more comprehensive way. With the aim to rise comfortability, speed, accuracy and manifoldness of real world process augmentations, a framework for the bidirectional interplay of the common process modeling world and the augmented world has been designed as morphologic box. Its demonstration proves the working of drawn AR integrations. Identified dimensions were derived from (1) a designed knowledge construction axiom, (2) a designed meta-model, (3) designed use cases and (4) designed directional interplay modes. Through a workshop-based survey, the so far best AR modeling configuration is identified, which can serve for benchmarks and implementations.

Keywords: Augmented reality · Process modeling
Simulation process building
Generalized knowledge constructin axiom · Meta-model · Use cases
Morphologic box · Industry 4.0 · CPS · CPPS · Internet of things

1 Introduction

Faced with a first process modeling approach augmenting common process models [3], the augmentation process is still uncomfortable and limited to various issues, such as the following examples show: First, a modeling is based on a rather demonstrative than complete set of modeling objects. It is not clear, which kinds of modeling objects are required to realize the full potential of an AR modeling. Second, it is not clear what the full potential of an AR modeling is connected to. First attempts show potentials in regard to non-transparent processes, a spacial positioning following a Cartesian definition, and identify attractive modeling contexts, such as knowledge-intensive processes, highly communicative behaviors and Industrie 4.0 scenarios [3]. But these are assumingly not the only potentials. Third, an AR modeling is based on a prototype character of workflows. It is not clear, which kinds of modeling activities shall be realized as before, and which

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 98–115, 2018.
https://doi.org/10.1007/978-3-319-94214-8_7

kind of modeling activities can be enabled by the use of AR hardware. Fourth, it is not clear by which kinds of modeling operations AR modeling activities can be carried out and are carried out best. Each concrete AR hardware provides individual characteristics and a standard in AR hardware is not available, yet.

Especially the interplay of the more or less paper based 2D world of a process modeling and the 3D AR modeling world is attractive to overcome previously mentioned issues and progress the AR modeling.

In this contribution, an *AR modeling* is referred to more than the simple enrichment of 2D modeling shapes and the positioning within the real world. It considers the process of the model construction, its visualization as static models and dynamic simulations, and the optimization of created process models, which can be realized by all: It can be realized with help of ordinary computer systems and modeling tools in the common 2D modeling world, it can be realized with help of AR hardware and modeling tools to be created in the 3D AR modeling world, or it can be realized with help of an integration of both.

Hence, the following research will focus on modeling with help of AR technology with the intention to answer the following research question: “How can processes be modeled within the augmented world?” This paper intends not to draw an all-embracing description of concrete, technical realizations of those novel process modeling techniques. It intends to set a first step to an integration of both modeling worlds. Hence, sub research questions are:

1. “How can an AR modeling be specified systematically?”
2. “How can process modeling be realized best in both worlds?”

Based on the assumption that each model creator wearing AR glasses or using other AR devices has a proper reality, the challenge lays in the synchronization of realities during a modeling cooperation process including numerous model creators. The original scientific contribution of this research therefore is an attempt to synchronize individual realities of an AR modeling by the creation of an *AR modeling framework* and the identification of a best configuration.

The research approach is intended to be design-oriented as Peffers proposes [13], such that the paper is structured as follows: The second section presents a foundation and underlying concepts, the third section derives objectives and presents a methodology for the specification of a bidirectional interplay of the common 2D modeling world and the 3D AR modeling world. Those are separated from the design of required artefacts, which will be presented in the fourth section. Their demonstration presented in the fifth section shows the application of designed artefacts. This is evaluated in the sixth section. Then, the final section concludes the paper.

2 Theoretical Foundation and Underlying Concepts

The first sub section presents approaches for model definitions, so that an interpretation for an AR modeling can be selected. Then, meta-model approaches are collected, so that a foundation for the meta-model design is available. Finally, basic control concepts are provided, which will be used for an AR modeling.

2.1 Model Definition Approaches

Some authors see the reason for the diversity of the term "model" in the history of its definition because definitions are based on separate thinking traditions [15, p. 2]. In accordance to Thomas, model definitions can be categorized by the following categories [20, p. 8]:

First, *Stachowiak's common model theory*, who defined a model to be the realization on an at least quintary predicate relation, which refers to the model x of the original y for the model user k within the period t and the intention z [17, p. 118].

Second, *axiomatic model definitions*, that are based on mathematical definitions using the field of mathematical logic, set theory, propositional calculus, predicate logic, etc. [19].

Third, *mapping-oriented model definitions*, which assume models to be mappings of the reality [7, p. 321]. Hence, the performance of the model creator is restricted to the selection of attributes being mapped to the model.

Fourth, *construction-oriented model definitions*. Those assume the reality not to be existent: Since each model creator perceives the reality from its own perception, it constructs its own reality [21, p. 9]. Hence, the creation of models (mentally or explicated) is highly creative and interpreted as construction process [20, p. 25].

Faced with the assumption of each model creator wearing proper AR glasses to construct its own reality, a construction-oriented model definition is attractive. This kind of definition will be the foundation for the *knowledge construction axiom* designed in Sect. 4.1.

2.2 Meta-model Foundation

In principle, meta-models of modeling languages provide taxonomies, that classify modeling objects following certain criteria [5, p. 66]. In literature, meta-models can be found, which provide perspective-oriented modeling taxonomies or approach-oriented taxonomies.

Perspective-oriented taxonomies list modeling items in regard to a certain modeling perspective. The following perspectives can be identified: *Functional*, *activity-oriented*, *behavioral*, *organizational*, *informational information flow-oriented*, *resource-oriented*, *knowledge and knowledge - flow - oriented* and *business process context* perspective ([9, p. 1533], [18, p. 3310]).

Approach-oriented taxonomies list modeling items in regard to one of the following approaches: *Activity-oriented approaches*, *role-oriented approaches*, *object-oriented approaches* and *speech-act-oriented approaches* ([8], [5, p. 67]).

In this contribution, a perspective-oriented meta-model will be provided in Sect. 4.2, since visualizations of AR modeling objects can focus only on relevant items of one of many perspectives. Hence, for the augmentation irrelevant objects can be suppressed perspective-wise.

2.3 Control Concepts

This sub section presents a collection of basic control concepts, which will serve for the operationalization of AR operations. Although a variety of concrete AR operations can be derived from given concepts, this contribution focuses on a first attempt and limits itself to one AR operation per control concept.

Buttons as graphical software elements serve as shortcut for functionality provided by software. Further, *guided processes* present graphical elements in an inherent manner, such that a sequence of simple activations of buttons, augmented representations, menu views, etc. collects information required by programs. Wherever possible, *workflows* realize mechanisms in the background. An activation can be realized easily by a *cursor* and a *touch pad activation*. Alternatively, a movement of the arms can be tracked by a camera, such that a *computer vision* recognizes movements specified in advance. A selection can be realized by the focus of eyes on an object (eye focus analysis). An activation can then be detected by an in advance specified eye blinking pattern (image or video analysis) [14]. Being recorded by a microphone, voice-based instructions can be recognized in regard to a specific context efficiently (speech recognition) [10] and serve as control command. Even EEG electricity can be used for a thought detection, such that instructions are tagged automatically similar to Koelstra et al. [6].

3 Objectives and Methodology

Following the DSRM approach [13], this section identifies objectives independent from a design. Then, a methodology is presented that satisfies methodological objectives. These are separated from the design and its demonstration, so that artefacts can be created and then, the fulfill of requirements can be evaluated. Following a methodological foundation, designed artefacts give evidence in a demonstration in regard to their functioning.

3.1 Objectives

Aiming to prepare a bidirectional interplay of the 2D modeling world and the 3D augmentation world, this section presents a set of requirements that has to be considered in the realization of artefacts. Requirements are presented category-wise. The first category refers to modeling languages in general, the so called meta-level of modeling languages. The second category is connected to the usage context of modeling languages, the here called scenario creation. The third category refers to AR hardware characteristics and serves for the selection of appropriate AR hardware. A fourth category focuses on methodological requirements.

In regard with a meta-level of modeling languages, the following objectives have been identified:

– The modeling shall support the creation of various process domains (knowledge-intensive, business, production processes, etc.) including state-of-the-art systems.

- The modeling shall support the creation of process simulations.
- The modeling shall support the use of several modeling languages.

With respect to the scenario creation, the following objectives have been identified:

- The modeling shall consider typical modeling scenarios.
- The modeling shall consider both modeling worlds (2D modeling and 3D AR).
- The modeling shall consider each model creator to have an own reality.
- The modeling shall synchronize model creator specific realities.

Focusing on the hardware selection (AR glasses or tablets), the following criteria were relevant additionally to AR technique inherent requirements such as the positioning within an area, performance issues, etc.:

- AR techniques shall provide a touch pad.
- AR techniques shall provide a microphone.
- AR techniques shall support the connection with further systems (eye tracking systems for eye focus analyses, online computer vision systems, EEG electricity detection systems for thought detection, etc.).
- AR techniques shall have WLAN access.
- AR techniques shall have Internet access.
- AR techniques shall support common basic operations.

Each identified objective of those three domains has been relevant for the design of the bidirectional interplay of the 2D modeling world and 3D augmentation world and serves as input for the following sections. Following a certain methodology, the following requirements have been identified for the methodologically backed-up creation of an AR modeling framework:

- Artefacts shall characterize an AR modeling by quantified parameters.
- Artefacts shall present an overview of possible solutions.
- Artefacts shall be expandable easily.
- Artefacts shall be constructed iteratively.
- Artefacts shall support a validation by empirical research.
- Artefacts shall support a validation by implementation and use in projects.

Based on the latter six requirements, a methodological foundation focuses on a morphological analysis.

3.2 Morphological Analysis

Following Zwicky, the *morphological analysis* is suited to explore all the possible solutions to a multi-dimensional, non-quantified complex problem for various domains, such as for instance in anatomy, geology, botany and biology [23, p. 34]. It is widely accepted and the history of morphological methods is summarized by Ritchey [16].

Being part of the general morphological analysis, the morphologic box, the so called *Zwicky box*, is constructed in five iterative steps [22]: First, *dimensions* of the problem are properly defined, which refer to relevant issues. Second, a spectrum of values, the so called *parameters*, are defined for each issue. Then, by setting the parameters against each other in an n-dimensional matrix, the morphologic box is created. Here, each cell of the n-dimensional box represents one parameter and marks a particular state or condition of the problem complex. Hence, the selection of one parameter of every dimension, the so called *configuration*, represents a solution of the problem complex. A fourth step scrutinizes and evaluates possible solutions in regard to the intended purpose. In a fifth step, the optimal solution is practically applied. If necessary, insights from the application are considered in previous steps.

4 Design of the Bidirectional Interplay

The design of a bidirectional interplay of the 2D modeling world and the 3D AR modeling world is based on the following parts. First, a *knowledge construction axiom* is presented. This is considered within a second sub section in the design of a *meta-model*. The third sub section presents *use cases* required for an AR modeling. Then, the interplay of both modeling worlds is characterized by the design of *directional interplay modes*. All of them can be considered as sub artefacts designed for the creation of the main artefact called *framework for the bidirectional interplay*. Their creation sequence, artefact relation and theoretical foundation is visualized in Fig. 1, so that the scientific contribution indicated by an asterisk (*) and theoretical foundation can be recognized.

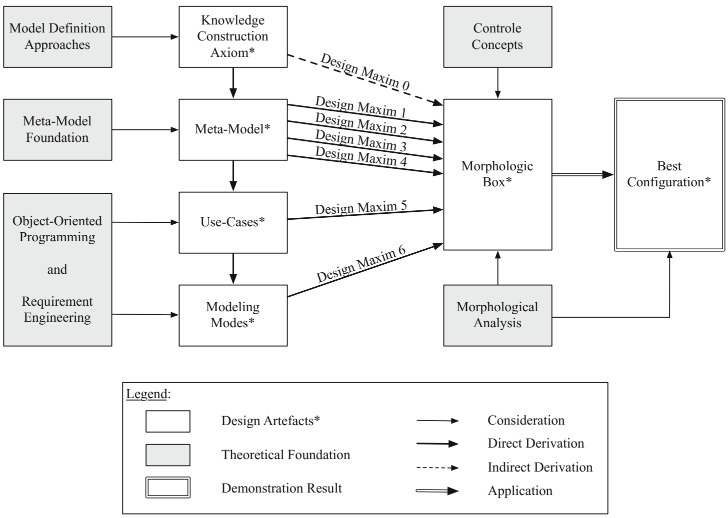


Fig. 1. Artefact creation.

In this figure, it becomes clear that the morphologic box can be considered as main artefact. *Design maxims* are derived from each sub artefact and serve for the construction of a morphologic box representing the *framework for the bidirectional interplay* of the 2D process modeling world and the 3D AR modeling world. Following the proceeding of a morphological analysis, its application serves as demonstration in accordance to Peffers et al. [13] and identifies a best configuration.

4.1 Knowledge Construction Axiom

Following a construction-oriented model definition, the reality is created by the perception of a subject (see Sect. 2.1). Since the production and company-wide processes in general are based on concepts constructed by subjects, any material transportation, conversation, work realization, etc. is based on the application of knowledge.

Knowledge-intensive processes can be operationalized on base of the SECI model [11] using *conversions*. The Knowledge Modeling and Description Language (KMDL) is an example for a modeling language using conversions [2]. Here, four kinds of conversions (*internalization*, *externalization*, *combination*, *socialization*) conceptualize the knowledge creation. Although the SECI model was originally meant as concept for knowledge creation, it can be generalized for value creation processes, which is denominated as *knowledge construction axiom* from here on. A generalization can be applied as follows:

Intentional material and data manipulations incl. their transfers are interpreted as combinations, where the corresponding objects are enriched to explicit knowledge in interpreting them within the context of the current activity. This refers to North's requirement of the enrichment by a *meaning* and *contextual embedding* [12, p. 41]. The enrichment can be carried out either by CPS or computer systems that have been enabled through information objects, such as computer programs, transportation orders, etc. Alternatively, they have been enabled by humans or CPS through knowledge objects, such as for example the intended production, transportation by themselves.

Non-intentional conversions, such as real-world processes (e.g. weather) and physical laws (e.g. gravity) are neglected based on the following interpretation: Even the interpretation of real world processes is created with intention and only by the subjective perception of the individual. Here, the observation of physical phenomenas (as-is values are collected) and its comparisons with to-be values, which are based on the beliefs and expectations of the individual, leads to the adjustment of the individual's beliefs and their making explicit (e.g. in form of physical laws). Hence, their use within process models is always associated with the use of information objects.

Enabled through this generalization, the knowledge construction axiom will be considered in the design of a meta-model for modeling languages and implicitly in the design of the morphologic box (Design Maxim 0).

4.2 Meta-model

Following the idea to identify AR-suited modeling activities, which can be beneficial for many modeling language approaches, the following presents a meta-model for modeling languages.

Since the meta-model integrates a great number of foundational concepts and abstracts over a range of modeling languages, a common understanding of modeling is addressed and a community-wide acceptance supported. The meta-model further tries to provide a state-of-the-art modeling concepts, that considers state-of-the-art concepts, such as Industry 4.0 components. The meta-model can be seen in Fig. 2.

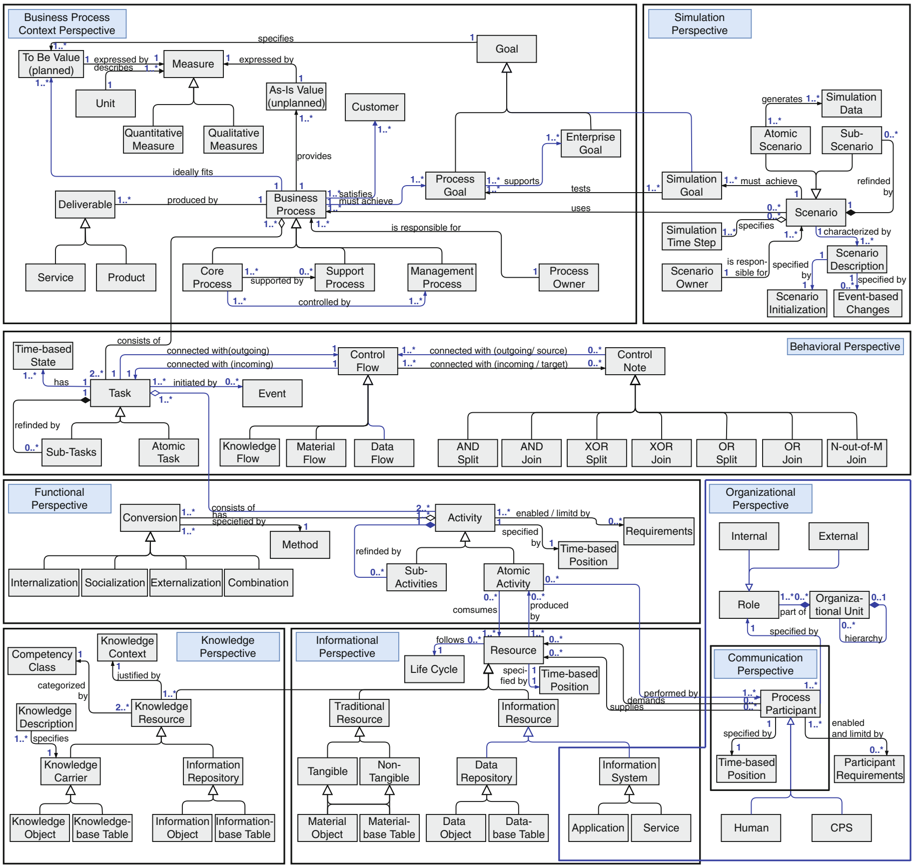


Fig. 2. Meta-model of a bidirectional AR modeling.

In this figure, one can find gray objects representing modeling objects. In accordance with Booch, objects are related by the following kinds of relations:

using relations are represented by arrows, *aggregations* and *compositions* by blanc and black diamonds, *inheritances* are represented by blanc triangles [1]. Required cardinalities can be found next to corresponding relations. Perspectives identified in Sect. 2.2 are boxed. Their title is highlighted in blue. Based on the model of List and Korherr [9], the meta-model presented here complements a knowledge perspective, a simulation perspective and a communication perspective, which were identified in literature. While additional objects and relations are drawn, others were only updated. Required cardinalities complement the meta-model too, which simplifies the implementation. In interpreting Fig. 2, the following independent design maxims can be derived:

Design Maxim 1. Considering the great amount of modeling objects, one can identify only some modeling objects being connected with "time-based positions". The focus of the modeling within the AR world shall lie on those activities.

Design Maxim 2. One can see that hierarchies of elements are established within complex task structures, complex activity structures and complex scenario structures. Hence, their organization is best realized within the 2D modeling world.

Design Maxim 3. The organization of knowledge objects, information objects, material objects, data objects and organizational units including humans and CPS can result in great hierarchies. Therefore, all those modeling objects are well suited for the organization within the 2D world, but can easily be completed by the positioning within the real world via AR techniques.

Design Maxim 4. Since the simulation is carried out in the background but leads to attractive visualizations of time and position based modeling objects, only the positioning of time-based objects is attractive for a scenario creation within the 3D AR world. Further, the simulation controlling is essential in both worlds.

Summing up, the meta-model serves for the derivation of a collection of design maxims. Those will be the foundation for the design of a bidirectional AR modeling.

4.3 Use Cases

In order to identify relevant dimensions for the bidirectional AR modeling, required operations were identified by the construction of a use case diagram. The use case diagram can be seen in Fig. 3 and considers perspectives identified in Sect. 4.2.

With focus on the model creator or the so called *actor*, who would like to create a model using an arbitrary modeling language, use cases within the 2D modeling world are visualized on its left and use cases within the 3D AR world are collected on its right side. While the 2D world is accessed by a common 2D modeling tool, which is here *Modelangelo* on a desktop or laptop [2], the 3D AR world is accessed by a common 3D AR modeling tool, which is here the

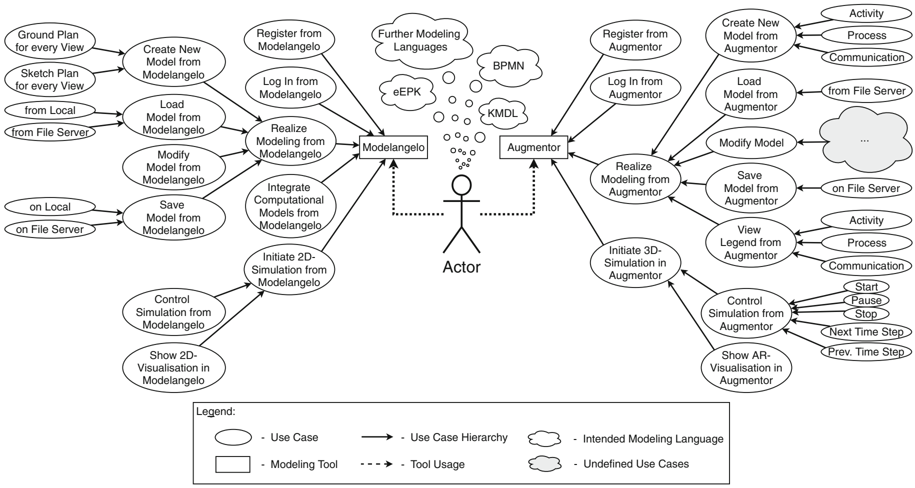


Fig. 3. Common use cases for a bidirectional AR modeling.

Augmentor on a tablet or AR glasses [3]. Since both are built to provide various modeling languages, the programs are suited for the generalization of common use cases for a bidirectional AR modeling.

The use case diagram of Fig. 3 shows basic operations, such as register, load and save activities, which can be found in both modeling worlds. They support working on a *local platform* and the use of *cloud services*. The model creation within the 2D modeling world considers the positioning of modeling objects within the AR world using a *ground plan* and *sketch plan* as well as the extension of modeling objects with 3D models following concepts of Grum and Gronau [3]. The focus of this contribution shall lie on the construction of models within the AR world, since a modification in the 2D modeling world is widely spread. In Fig. 3, this is represented by a gray cloud showing dots and is specified in a later section. The following design maxim can be derived:

Design Maxim 5. Use cases identified in the use case diagram presented here are interpreted as basic operations and will be required in both modeling worlds.

Summing up, the use case diagram serves for the derivation of a design maxim in regard to operations required for an AR modeling in both worlds. Those will be the foundation for the design of a bidirectional AR modeling integrating the 2D world and the 3D AR world.

4.4 Directional Interplay Modes

Dependent on the intended modeling situation, four kinds of *AR modeling modes* can be identified: First, the *no-interplay-mode*, which allows a modeling only within the 2D modeling world or within the 3D AR modeling world. Second, a *one-person-mode*, that realizes a modeling in both modeling worlds for a single person. Third, a *collaboration-mode*, that realizes a modeling in both modeling worlds, while only one person is modeling and others give feedback based on visualizations. Fourth, a *muliple-device-mode*. Here, various model creators are modeling cooperatively using different kinds of modeling devices. The modeling modes can be seen in Fig. 4.

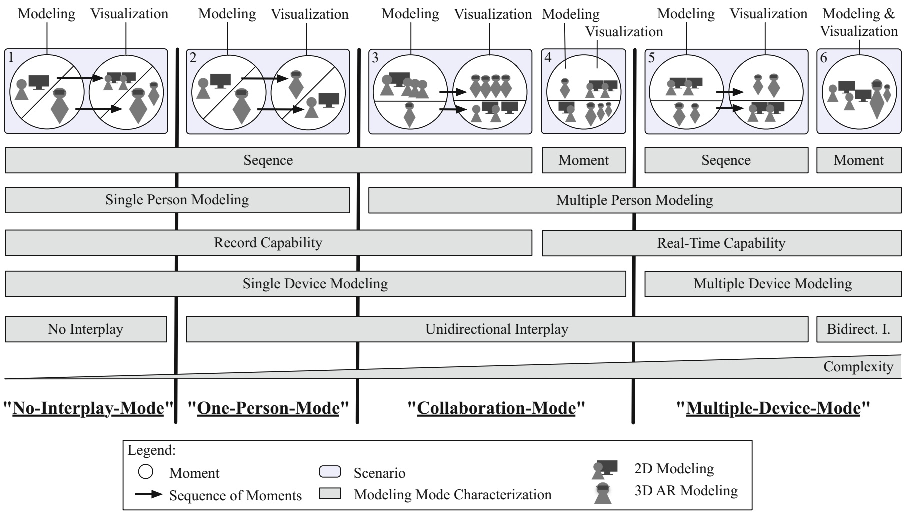


Fig. 4. Common modeling modes for an AR modeling.

The modeling modes are connected to one of six *modeling scenarios* (visualized on the very top). The first scenario refers to two moments. In a first moment, the modeling is carried out either in the common 2D modeling world using desktops and laptops, or in the 3D AR modeling world using AR devices. Then, in a second moment, constructed models are visualized within the same modeling world. The second scenario refers to the model construction in one modeling world and the visualization in the consecutive moment within the other modeling world. The third scenario refers to the model construction in one modeling world and the visualization for many individuals within the other modeling world. The fourth scenario integrates the first and the second moment of the third scenario in one moment. The fifth scenario enables several model creators to work on one model within the same modeling world and to visualize this model for many individuals in a second moment within the other modeling world. The sixth scenario integrates the first and the second moment of the fifth scenario in one moment and connects both modeling worlds bidirectionally.

The realization of each scenario demands different implementation requirements, which are visualized by gray rectangles within the figure. The following independent design maxim can be derived:

Design Maxim 6. Implementation requirements refer to both modeling worlds and characterize an AR modeling.

Since implementation requirements characterize an AR modeling, they will serve as dimensions for the morphological box and are specified in the following: The first dimension refers to the issue that some modeling scenarios demand for a *temporal dissolution* of modeling activities, such as the modification and visualization: Some scenarios only demand a sequential realization of modeling activities (*sequence*) and others demand to realize them simultaneously in one *moment*. The second dimension refers to the number of persons (*single persons* or *multiple persons*) currently modeling with a desktop, laptop, tablet or AR glasses. The third dimension refers to the question, if a *real-time capability* is required, or activities can be *recorded* and used in later sequences. The dimension is called *technical time capabilities* here. The fourth dimension refers to the number of devices (*single device* or *multiple devices*), which are required for the model creation. This focuses on modeling systems as active systems, which are separated from passive systems only visualizing information from active systems. The fifth dimension refers to the *interplay of the modeling worlds* having three parameters: Modeling worlds can be disconnected, such that models are exchanged only within the corresponding world (*no interplay*). Alternatively, modeling worlds are connected in one direction, such that an *unidirectional interplay* realizes exchanges from one world to the other. Only a *bidirectional interplay* realizes the exchange of modeling activities from 2D modeling world to 3D AR modeling world and vice versa without limitation.

Faced with the six modeling scenarios, the complexity and requirements for an implementation rise with every number. Hence, a stepwise implementation from the first to the last modeling scenario is recommended.

Summing up, *AR modeling modes* presented here serve for the derivation of a collection of requirements. Those will be part of the design of a bidirectional AR modeling.

5 Demonstration of an AR Modeling

The complex problem of designing an AR modeling is parametrized by a morphologic box. The morphologic box can be seen in Fig. 5 and shows dimensions considering design maxims identified previously as follows.

The *knowledge construction axiom* presented in Sect. 4.1 is considered implicitly by the meta-model for AR modeling (Sect. 4.2). This answers the question, which kind of modeling objects shall be considered in which world. *Use cases* identified in Sect. 4.3 are considered in order to answer the question, which kind of operations are required in which world. Here, operations are clustered to the categories *modeling*, *simulation* and *administration*. The *modeling modes* designed in Sect. 4.4 are considered in order to answer the question, which kind of background requirements are necessary in regard to modeling scenarios.

Parameters for each dimension are derived from basic control concepts presented in Sect. 2.3. In general, they are based on the use of a touch pad, eye

Origin	Dimension	Parameter					
Meta-Model for an AR Modeling	Modeling Object Focus	only time-based position related objects	perspective related objects	all objects			
Use Case Diagram for an AR Modeling (Modeling)	Choose Intended Shape	per cursor and touch pad activation from legend	per physical look on AR modeling belt	per voice recognition	per thoughts and unique shape denomination		
	Select Intended Shape	per cursor and touch pad activation	per visual contact and eye focus analysis	per voice recognition and unique shape denomination	per thoughts and unique shape denomination		
	Grab Intended Shape	per touch pad activation	per image analysis and grap movement	per eye blink pattern	per voice recognition	per thoughts about grab instruction	
	Position Intended Shape	per touch slide along axises	per movement within reality (sensory information, image analysis)	per voice recognition (axis and distance instructions)	per thoughts about position instruction		
	Drop Intended Shape	per touch pad activation	per image analysis and drop movement	per eye blink pattern	per voice recognition	per thoughts about drop instruction	
	Delete Intended Shape	per shape selection and delete button activation	per image analysis and delete movement	per eye blink pattern	per voice recognition	per thoughts about delete intended shape instruction	
	Connect Intended Shape	per cursor on connection visualization and touch pad activation	per relation object selection (see „choose intended shape“)	per image analysis and connect movement	per eye blink pattern	per voice recognition	per thoughts about connect shapes instruction
	Zoom On Intended Shape	per cursor and touch pad activation using two fingers	per image analysis and zoom movement	per eye blink pattern	per voice recognition	per thoughts about zoom instruction	
	Modify Size of Intended Shape	per shape and axis selection and touch pad activation using two fingers	per image analysis and size modification movement	per eye blink pattern	per voice recognition	per thoughts about increase size instruction	
	Rotate Intended Shape	per shape and axis selection and touch pad activation using three fingers	per image analysis and rotate movement	per eye blink pattern	per voice recognition	per thoughts about rotate instruction	
	Undo	per button selection and touch pad activation	per image analysis and undo movement	per eye blink pattern	per voice recognition	per thoughts about undo instruction	
	Redo	per button selection and touch pad activation	per image analysis and redo movement	per eye blink pattern	per voice recognition	per thoughts about redo instruction	
Use Case Diagram for an AR Modeling (Simulation)	Start Simulation	per cursor on start button and touch pad activation	per image analysis and start movement	per eye focus on start button and blink pattern	per voice recognition of start instruction	per thoughts about start instruction	
	Pause Simulation	per cursor on pause button and touch pad activation	per image analysis and pause movement	per eye focus on pause button and blink pattern	per voice recognition of pause instruction	per thoughts about pause instruction	
	Stop Simulation	per cursor on stop button and touch pad activation	per image analysis and stop movement	per eye focus on stop button and blink pattern	per voice recognition of stop instruction	per thoughts about stop instruction	
	Next Time Step of Simulation	per cursor on next button and touch pad activation	per image analysis and next movement	per eye focus on next button and blink pattern	per voice recognition of next instruction	per thoughts about next instruction	
	Previous Time Step of Simulation	per cursor on previous button and touch pad activation	per image analysis and previous movement	per eye focus on previous button and blink pattern	per voice recognition of previous instruction	per thoughts about previous instruction	
Use Case Diagram for an AR Modeling (Administration)	Register	per digital keyboard and key selection (see „choose intended shape“) in guided register process	per movement on key of digital keyboard and image analysis in guided register process	per eye focus on key of digital keyboard and blink pattern in guided register process	per voice recognition in guided register process	per thoughts in guided register process	
	Login	per digital keyboard and key selection (see „choose intended shape“) in guided login process	per movement on key of digital keyboard and image analysis in guided login process	per eye focus on key of digital keyboard and blink pattern in guided login process	per voice recognition in guided login process	per thoughts in guided login process	
	Create New Model	per digital keyboard and key selection (see „choose intended shape“) in guided create process	per movement on key of digital keyboard and image analysis in guided create process	per eye focus on key of digital keyboard and blink pattern in guided create process	per voice recognition in guided create process	per thoughts in guided create process	
	Load Model	per digital keyboard and key selection (see „choose intended shape“) in guided load process	per movement on key of digital keyboard and image analysis in guided load process	per eye focus on key of digital keyboard and blink pattern in guided load process	per voice recognition in guided load process	per thoughts in guided load process	
	Save Model	per save buton selection (see „choose intended shape“)	per movement on key of digital keyboard and image analysis in guided save process	per eye focus on save button and blink pattern in guided register process	per voice recognition and save instruction	per thoughts about save instruction	
Modeling Modes for an AR Modeling	Modeling Scenario Id	1	2	3	4	5	6
	Temporal Dissolution of Modeling Activities	Moments	Sequences				
	Number of Persons Modifying Models	Single Person Modeling	Multiple Person Modeling				
	Technical Time Capabilites	Record Capability	Real-Time Capability				
	Number of Devices Used for Modifying Models	Single Device Modeling	Multiple Device Modeling				
	Modeling World Interplay	No Interplay	Unidirectional Interplay	Bidirectional Interplay			
	AR Modeling Modes	Unidirectional-Interplay-Mode	One-Person-Mode	Collaboration Mode	Multiple-Device-Mode		

Fig. 5. Morphologic box for a bidirectional AR modeling.

focus analysis, voice recognition, image and video analysis, computer vision and thought detection. Parameters presented here do not attempt for completeness rather than provide a first approach for structuring and parametrize an AR modeling. Given a highly evolutionary technical environment and so far non-standardized hardware controllers, parameters of each dimension and dimensions of the morphologic box can be extended and modified easily. Hence, on base of a continual redesign of the morphologic box, an always uptodate and state-of-the-art, flexible framework can be constructed.

A selection of parameters for each dimension of the morphologic box serves as *configuration* of an AR modeling. An example configuration is given by yellow highlighted cells (Fig. 5). Using the morphologic box presented here, each configuration is suited for an application in an AR modeling tool, such as the *Augmentor* [3].

6 Evaluation

All in all, the morphologic box presented in Sect. 5 provides 132 different configurations. Each represents a working AR modeling characterization. Since this framework is designed to be flexible and changes in the contemporary environment, such as IT hardware developments and creative, new control concepts, will lead to further dimensions and parameters. This drastically increases the number of total configurations and with this the complexity to identify attractive configurations rises as well. The question remains, which of those configurations supports an AR modeling best and how best configurations can be identified easily.

Aiming to identify the configuration of the morphologic box, which realizes process modeling best in both worlds, the common 2D process modeling world and the 3D AR modeling world, an evaluation of available configurations can be carried out in two ways in accordance to Zwicky: First, a survey can be conducted in a workshop session with modeling experts, so that their majority acceptance can be identified for one configuration. Second, experiences can be collected based on the application of one morphologic configuration in a software and its use in modeling projects [22]. Here, an evaluation can either focus on this individual configuration and evaluate its practicability within realistic project settings, or it can focus on a comparison of a set of applied configurations. In a comparison of this set, the best configuration will show most attractive evaluation values.

Since an implementation of any configuration of the morphologic box here presented has not been realized yet, and the best configuration of this morphologic box will be implemented, the first evaluation approach is attractive and was carried out as described in the following:

Interviews have been held with a group of 63 individuals educated in Business Process Management (BPM) at the University of Potsdam for one semester. This includes students of all: economics, computer science and business informatics. The BPM module included a theoretical BPM education, the application of

Table 1. The fulfillment of design-science research guidelines

Guideline	Description
Guideline 1: Design as an artifact	The authors design a flexible framework for the definition of an AR Modeling within the common 2D process modeling world and the 3D AR modeling world. This is founded on the following sub artefacts: a knowledge construction axiom, a meta-model, a use-case collection and a modeling mode design. This framework is demonstrated in workshop sessions. A best configuration is identified that can serve for benchmarks with further AR modeling approaches
Guideline 2: Problem relevance	Considering the previously mentioned artifacts, the business problem of complex interplay of 2D modeling world and 3D AR modeling world is overcome by a simple configuration of a morphologic box. With this artefact, a common framework is presented that can serve for comparison with further AR modeling approaches, and be applied to different implementations. As the framework is parameter-based, the concrete framework around contemporary AR hardware and IT systems is reasonable, given a highly evolutionary technical environment and the continual application of the framework
Guideline 3: Design evaluation	The efficacy of the designed artifacts was demonstrated rigorously by means of surveys conducted in workshop sessions with modeling experts. The utility and quality of the morphologic box was demonstrated by the identification of a configuration with the most acceptance. The execution precisely followed the proceeding specified by [22]. Therefore, validation of the morphological box is valid within a first application, and will be implemented and used in larger projects and real-life settings as a next step
Guideline 4: Research contributions	The design-science contributions of this research are the proposed framework, its sub artefacts and evaluation results in the form of surveys conducted in workshops. These contributions advance our understanding of the manner in which to carry out AR modeling best
Guideline 5: Research rigor	Research on process modeling approaches has long been based on the 2D world. In this contribution, a multi dimensional, parameter-based framework provides the underlying integration strategy of the common 2D process modeling world and the 3D AR modeling world, which allows for efficient process modeling realizations (such as the modeling, sharing, cooperation and visualization) and enables the development of more context-specific modeling software, benchmarks, and applications in projects
Guideline 6: Design as a search process	As discussed previously, the implementation of AR modeling strategies, application in projects, and benchmarking in iterations is essential. The authors studied variations in realization strategies over a period of 7 months within the aforementioned workshops. Creativity and problem-solving capabilities were involved in the construction of this framework
Guideline 7: Communication of research	The presentation of this research is aimed at an audience familiar with process modeling theory, AR hardware and software implementation. Even so, the contribution provides useful information for managerial audiences. While the authors present a thorough discussion of sense-full configurations, the contribution provides evidence for both technical implementations and economic reasoning

five modeling languages in different modeling projects and the discussion about concepts presented here. In workshops, the acceptance per each dimension of the morphologic box was made subject to the discussion and a preference in regard to presented parameters and a practical implementation has been conducted.

Overall, that parameters were determined for each dimension, that showed the greatest acceptance. Considering Fig. 5, highlighted, yellow cells do not only represent an example configuration, as was mentioned in Sect. 5. They represent parameters with the greatest acceptance. Hence, highlighted cells can be interpreted as best configuration for an AR modeling so far.

7 Conclusion

In this paper, a flexible *framework for a bidirectional AR modeling* has been drawn. This supports the use of various modeling languages since it is derived from a *meta-model* for modeling languages. It considers state-of-the-art systems and various modeling domains since it follows the *knowledge construction axiom*. Further, basic operations and requirements are considered following typical *use cases* and *AR modeling modes*. Six design maxims were derived from those sub artefacts.

By the integration of all six design maxims into a morphologic box, a framework was presented that specifies an AR modeling systematically. With this, the first sub research question is answered (*"How can an AR modeling be specified systematically?"*). The conduction of the greatest acceptance per dimension of the morphologic box in a survey answers the second sub research question (*"How can process modeling be realized best in both worlds?"*). The configuration highlighted in Fig. 5 defines how a process modeling is realized best in both worlds. So, considering all artefacts and insights in regard to sub research questions, the main research question (*"How can processes be modeled within the augmented world?"*) can be answered effectively: In accordance with the design-science research guidelines of Hevner et al., this contribution satisfies the requirements for effective design-science research and is complete [4], as it is indicated in Table 1.

This table presents seven design-science research guidelines and describes how presented artefact and this contribution satisfies them. Presented insights and research contributions have to be limited in so far as artefacts only have been validated by surveys conducted in workshop sessions. This is satisfying for now, since an implementation of the best AR modeling configuration in modeling tools is still missing.

Therefore, next steps will focus on an implementation of the best AR modeling configuration identified here and its application in various modeling projects, such that the main research question can be answered in a statistical satisfying manner.

References

1. Booch, G.: Object-Oriented Analysis and Design with Applications, p. 2. The Benjamin/Cummings Publishing Company Inc., San Francisco (1995)
2. Gronau, N.: Modeling and Analyzing Knowledge Intensive Business Processes with KMDL: Comprehensive Insights into Theory and Practice, vol. 7. GITOmbh, Berlin (2012)
3. Grum, M., Gronau, N.: Integration of augmented reality technologies in process modeling - the augmentation of real world scenarios with the KMDL. In: Proceedings of the Seventh International Symposium on Business Modeling and Software Design, BMSD, vol. 1, pp. 206–215. SciTePress (2017). http://www.scitepress.org/PublicationsDetail.aspx?ID=iMEBA7DZ6aw=&t=1. ISBN 978-989-758-238-7
4. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. Manag. Inf. Syst. Q. **28**(1), 75–105 (2004)
5. Kethers, S.: Multi-Perspective Modeling and Analysis of Cooperation Processes. University of Aachen, Aachen (2000)
6. Koelstra, S., Mühl, C., Patras, I.: EEG analysis for implicit tagging of video data. In: 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Amsterdam, pp. 1–6 (2009). https://doi.org/10.1109/ACII.2009.5349482
7. Kosiol, E.: Modellanalyse als Grundlage unternehmerischer Entscheidungen. Westdeutscher, Cologne (1961)
8. Kueng, P., Bichler, P., Kawalek, P., Schrefl, M.: How to compose an object-oriented business process model? In: Brinkkemper, S., Lyytinen, K., Welke, R.J. (eds.) ME 1996. IFIPAICT, pp. 94–110. Springer, Boston (1996). https://doi.org/10.1007/978-0-387-35080-6_7
9. List, B., Korherr, B.: An evaluation of conceptual business process modelling languages. In: Proceedings of the 2006 ACM Symposium on Applied Computing, pp. 1532–1539 (2006). https://doi.org/10.1145/1141277.1141633
10. Michaely, A.H., Ghodsi, M., Wu, Z., Scheiner, J., Aleksic, P.: Unsupervised context learning for speech recognition. In: IEEE Spoken Language Technology Workshop (SLT), San Diego, CA, pp. 447–453 (2016). https://doi.org/10.1109/SLT.2016.7846302
11. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford University Press, Oxford (1995)
12. North, K.: Wissensorientierte Unternehmensführung: Wertschöpfung durch Wissen. Gabler, Wiesbaden (1998)
13. Peffers, K., Tuunanen, T., Gengler, C.E., Rossi, M., Hui, W., Virtanen, V., Bragge, J.: The design science research process: a model for producing and presenting information systems research. In: 1st International Conference on Design Science in Information Systems and Technology, DESRIST, vol. 24, no. 3, pp. 83–106 (2006)
14. Pérez, B.N., Sanfeliu, A., Vidal, E.: National Symposium in Pattern Recognition and Image Analysis: Selected papers from the IVth Spanish Symposium. World Scientific, Singapore (1992)
15. Reihlen, M.: Arbeitsberichte des Seminars für Allgemeine Betriebswirtschaftslehre, In: Delfmann, W. (ed.) Betriebswirtschaftliche Planung und Logistik der Universität zu Köln, Köln (1997)
16. Ritchey, T.: Problem structuring using computer-aided morphological analysis. J. Oper. Res. Soc. Spec. Issue Probl. Struct. Methods **57**(7), 792–801 (2006). https://doi.org/10.1057/palgrave.jors.2602177

17. Stachowiak, H.: Modelle - Konstruktion der Wirklichkeit. Fink, München (1983)
18. Sultanow, E., Zhou, X., Gronau, N., Cox, S.: Das Modellverständnis in der Wirtschaftsinformatik: Historie, Literaturanalyse und Begriffsexplikation. Int. Rev. Comput. Softw. IRECOS **6**, 3309–3319 (2012)
19. Tarski, A.: Der Wahrheitsbegriff in den formalisierten Sprachen. Studia Philosophica (1935)
20. Thomas, O.: Modeling of processes, systems and knowledge a multi-dimensional comparison of 13 chosen methods. Institut für Wirtschaftsinformatik (IWi) im DFKI **184**(1), 1–41 (2005)
21. Watzlawick, P.: Die erfundene Wirklichkeit: Wie wissen wir, was wir zu wissen glauben?. Piper, München (1969)
22. Zwicky, F.: Entdecken, Erfinden, Forschen im Morphologischen Weltbild. D. Knaur, California (1966)
23. Zwicky, F.: Discovery, Invention, Research - Through the Morphological Approach. The Macmillan Company, Toronto (1969)

Efficient Aggregation Methods for Probabilistic Data Streams

Maksim Goman(✉)

Department of Business Informatics – Information Engineering,
Johannes Kepler University, Linz, Austria
Maksim.Goman@jku.at
http://www.ie.jku.at

Abstract. In this paper, we consider aggregation algorithms for SUM operator for uncertain stream processing. Deterministic algorithms can not be used here because of uncertain data and high rates of data change, time and memory constraints. We compare the most promising available methods. Instead of full distribution functions of query result, we use a set of six parameters based on key moments and quantiles to describe the distributions. It enables us to perform fast recomputations of the aggregation with $O(1)$ complexity. Experimental results demonstrate good performance of uncertain aggregation in comparison to deterministic case. We also found that usage of central limit theorem may be restricted to problems where data satisfy certain conditions.

Keywords: Probabilistic databases · Uncertain query
Uncertain data · Stream processing · Uncertain aggregation

1 Introduction

Currently, complexity of business processes grows, streams of data from sensors and data interchange with third-party software accelerate and increase, big data analysis became a routine activity, in-memory databases became normal practice. The trend is clear that complex business analysis is going to be performed real-time on every minor data update and control dashboards are constantly updated. Additionally, risk management becomes extremely vital for business competitiveness. Risk is always connected with uncertainty. Uncertainty is usually present in many kinds of data, mainly in forecasts and estimations, imprecise measurements and ambiguous observations where uncertainty may not be excluded. In relation to massive data streams, risk may originate directly from uncertainty in the raw data. Beyond risk analysis, due to uncertainty in data, applications of uncertain data processing can be real-time scheduling of atomic operations (e.g. in manufacturing), quality control, monitoring of user satisfaction, worldwide fraud management in financial transactions, etc.

Thus, enterprise software systems are likely to perform uncertain data processing soon. Current deterministic algorithms will need replacement. Moreover,

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 116–132, 2018.
https://doi.org/10.1007/978-3-319-94214-8_8

great volumes of data for uncertain databases as well as shift of processing to mobile devices having lack of computational power appeal for fast algorithms.

We will illustrate importance of uncertain aggregation with a couple of examples. Both of them can be modeled as a data stream with sliding window of fixed time or size.

Firstly, let us imagine a logistic supply chain in which products lose their value according to known distribution (e.g., a cold supply chain). Certain parameters can influence the actual condition of each item of products of many types and this is accounted for in the distributions. Some of the parameters of all or some items are regularly updated, although this information may have random error too. Real condition of each unit is unknown to control system until it is sold or disposed. Nevertheless, possible loss due to natural deterioration and need for additional delivery to certain hubs or regions can be computed probabilistically. Value of uncertain aggregation with some probabilistic threshold can be used in a decision rule in inventory management system as well as in certain cost/benefit or sales trend prognoses in ERP constantly.

Secondly, a controller can monitor a large area collecting information from mobile sensors. Sensors send messages with relevant data attributes of known types. Technology, ambiance, and other factors introduce random errors into collected data. There is uncertainty about position of every sensor (i.e., does it belong to the area) and values of attributes in messages from sensors. We assume, that we know distributions of the errors. Control influence of the controller depends on aggregation result of the data from sensors in the area (e.g., an alarm signal). Use of stochastic algorithms in such problem makes it possible better adjustment of control parameters, fine-tuning the thresholds for control variables and reduction of excessive false positive and false negative events. Uncertain aggregation is able to assist here. Notice, that uncertainty about position of sensors is actually an existence uncertainty, i.e. distribution of attributes in messages are conditional distributions in this example.

This work introduces fast universal approach for uncertain aggregation operator SUM suitable for application in data stream processing. Our method does not depend on type of distributions of variables and obtains fast approximations of result characteristics of distributions with $O(1)$ complexity without using integration. Instead of full distribution functions of query result, we use a set of six parameters based on key moments and quantiles to describe the distributions. The advantage of such algorithms is applicability to any current problem where aggregation of uncertain data is needed.

Uncertain attribute values are usually represented as random variables. They can be a set of possible values with their respective probabilities that are mutually exclusive (probability mass function) for a discrete case or with probability distribution function (PDF) for continuous case. The attribute-based probabilistic model is used in our study. Comments regarding tuple-based uncertainty (e.g., tuple existence uncertainty TEP) will be given as well. We will work with continuous PDFs and concentrate on getting suitable result for our problem from uncertain query without consideration of possible worlds semantics (PWS) conception. This enables generalization of results to any type of distribution of uncertain attributes.

Original Contribution. To begin with, we suggested alignment of the uncertainty aggregation problems to enterprise domain applications. To the best knowledge of the authors, this is the first piece of work that analyses the problem of uncertain aggregation with $O(1)$ complexity for uncertain data streams. Our uncertain operation SUM is universal in its application to uncertain variables with arbitrary distributions. A six-value-set is used to store the aggregation result and we considered three methods that compute two bounding values for the six-value-set with $O(1)$ operation complexity. This enables efficient maintenance of the aggregation on sliding window change and suits data stream processing.

Next, attention to problems of conditional uncertainty (e.g., TEP) and cases of complex multivariate distributions are also innovative for uncertain stream processing. Consideration of uniform and multivariate distributions are new in this context, e.g., we derived Chernoff bounds for uniformly distributed random variables for our method. While criteria for CLT applicability are well known in probability theory, our conclusion about its restricted suitability to applications of uncertain data processing is a contribution. These constraints of CLT applicability are usually omitted in papers. Finally, we touched the problem of relevance of uncertain aggregation result to the needs of a certain application problem. Particular problems may require detailed information about resulting distribution and others may not need it. As a result, different uncertain aggregation methods can be required.

The rest of the paper is organized as follows. Problem formulation is given in Sect. 2. The review of today's state of the art in uncertain and probabilistic query processing is done in Sect. 3. Methods of efficient uncertain aggregation are introduced in Sect. 4. Section 5 shows results of experimental evaluation. Conclusion summarizes the work and outlines future research directions.

2 Problem Definition

We consider a continuous SQL-like query of the following type against a fixed length sliding window of a constant width n in data stream:

$$\text{SELECT SUM}(\mathbf{X}) \quad \text{FROM [window]},$$

where $\mathbf{X}$ is an uncertain (random) vector.

Tuples in the window have at least one uncertain attribute. Relational function SUM is applied to uncertain attribute $\mathbf{X}$ of tuples in the window. Queries in the sliding-window are continuous: Query should be updated whenever there are changes in sliding window because an attribute was updated or a new tuple came to the window or an old one left the window. A sliding window can be very large and the stream is supposed to be very fast. In order to maintain temporal validity, one should remove the expired tuples from aggregation. As tuples are continuously arriving and leaving the fixed-length window, recomputation of the sum should be very fast. To incorporate the aggregation method into queries with grouping and aggregation or nested queries, we need to maintain aggregations for many groups and do it fast. We seek for an opportunity to update our aggregation easily with complexity $O(1)$.

All uncertain attributes in our stream are described by known probability distributions. All uncertain attributes and tuples are considered independent. Only continuous query in sliding window is under consideration and a threshold will be an integral part of the aggregation operator.

Fast approximate estimations of the result of uncertain SUM operator is used instead of full distribution. Aggregation operator returns six-value-set that describes the distribution of the aggregation result similar to [1]: Expected value μ, variance σ, an upper bound U_b, the probability P_{U_b} that the value is above this upper bound, a lower bound L_b, and the probability P_{L_b} that the value is below this lower bound. Probabilities for upper and lower bounds are obtained from the given threshold parameter α. The answer is flexible for desired accuracy (measured by α) and easy to explain.

Applicability of the following techniques will be studied for estimation of the bounds in our aggregation:

- direct approach through use of known quantile function;
- approximation with central limit theorem (CLT);
- approximation with Chernoff bounds using moment generating functions (MGF) or characteristic functions (CF) or with Chebyshev bounds.

As literature shows, while normal distribution is widely used in research of queries, the uniform distribution is not usually considered, except the case of the use of random sampling methods. Therefore, normal distribution and uniform distribution families will be considered because they are naturally very useful for many real-life problems. An uncertain attribute is given by the structure of two values of its parameters in one of the following forms:

1. All random variables are normally distributed values $N(\mu_i, \sigma_i)$, where μ means the expected value and σ is the respective variance.
2. All random variables are distributed uniformly in intervals $U(a_i, b_i)$.

Other distributions of random variables can be considered by analogy. Attention to models with multiple random variables will be given. Conceptually, tuples with two or more dimensional random variables as well as tuples with joint random variables should be processed in the same way. The main ideas will be illustrated on the example of one random variable in a tuple. For simplicity, only normally distributed attributes were used in experimental evaluation.

3 Related Work

There are not many publications about aggregation for uncertain data streams. Because the problem of uncertain aggregation is important for uncertain databases, we paid attention to what was done in that domain too.

Compared with uncertain queries in probabilistic datasets, uncertain queries for uncertain data streams suffer from time constraints and limited memory resources. While approximation is a common approach to improve speed of queries, techniques must achieve small bounded error and meet stringent performance requirements [2]. To compute the resulting distributions of aggregates of attributes having conditional distributions or tuple existence uncertainty (TEP) is a hard problem as well.

To aggregate two or more random variables, we need to produce convolution of their PDFs. It is done by means of integration. This is a computationally expensive operation. In order to add n random variables, we need to perform integration $n-1$ times. Even if one uses an approximation method for the intermediate PDFs, the complexity is still exponential in n. Depending on implementation, use of approximation may not be easily scalable.

The result of aggregation depends on what an application expects as the outcome. Researchers are usually employing approximations instead of distributions of random variables. Some approaches focus on obtaining one exact aggregate value (e.g. mean, mode or expected value) and as such, transfer to deterministic expected values rather than consider distributions or realizations of PWS. This means the loss of uncertainty. In many applications, data uncertainty is important and vital for its goals. Although, the full distribution of the answer is not required in most applications, additional information seems appropriate. The result of the uncertain aggregation operator should include uncertainty to some extent. In the very general case, the resulting PDF of aggregation is a function of the input PDFs. There are two principal ways to evaluate uncertain data aggregation [1]:

- compute full or approximate distribution of aggregation;
- produce information about resulting distribution with a number of values, e.g., its main moments and some approximated bounds.

Many studies in probabilistic databases focus on discrete probabilistic attributes and employ PWS [1–5]. According to Tran [4], aggregates in existing probabilistic databases were restricted to a couple of the principal moments of resulting distributions like expectation and variance because it is computationally difficult to obtain resulting distribution. Accordingly, aggregates in probabilistic data streams are usually represented as expected values, variance and some higher moments [3]. Some researchers try to produce only one value – expected value of the aggregation of random variables [6,7]. Simplifications like that are very widespread [1,2,7–9]. One of the reasons is concern about the speed of the query. Many techniques for continuous random variables employ multivariate integration, discretization and Monte Carlo simulation, but according to Tran [4], they have low performance and sometimes even low accuracy. In contrast, Tran [4] considers, that knowing a few moments of a query (aggregate) is not enough to answer the queries accurately and aims at characterizing full distributions of the answer.

Continuously distributed random variables allow usage of probability theory techniques for them. This can be generalized for a discrete case. There are two main approaches for continuous attributes: integral-based approach [10] and sampling-based approach [1]. Integration is too slow and discretization may not offer accuracy guarantee. Monte Carlo simulations need quite a large number of samples to achieve higher accuracy which makes it slow for stream processing.

Cheng et al. [10] provided a general classification of probabilistic queries and evaluation algorithms for uncertain data sets including aggregates. The authors emphasize that answering uncertain aggregate queries is more challenging than range queries, because the answer is application dependent: What value to report, e.g., only the object with the highest probability or only those objects whose probability exceeds a minimum probability threshold. Several algorithms were given and quality of results returned by probabilistic queries was discussed.

Aggregation functions are also considered in [1]. The authors claim that statistical information about the result of a query is more important than the full distribution and the information can be computed not only faster, but also be more accurate. The answer of a query is a six-value-set describing the resulting distribution with bounds and most of the distribution is inside the bounds with a given reliability [1]. They conclude that discrete distributions suit fast calculations better. At the same time, the authors prove that computing a continuous distribution of the result is not effective, not scalable and slow. This conclusion coincides with the result of experiments by Tran [3,5].

SUM and COUNT operations are usually considered to be the simplest. Calculation of others is considered more difficult [2,11–13]. Wang [2] provides an extensive overview of probability representation models, many types of uncertain queries (including top-k, nearest-neighbor queries, aggregates, joins, range and threshold queries) and their features for uncertain databases and stream applications. The paper gives also their relational algebra prototypes of uncertain aggregation functions and uses simplification: The output of aggregation is represented as one expected value instead of a distribution.

Sampling techniques are used for aggregates. For both continuous and discrete random attributes, this enables to convert a probabilistic stream to a deterministic stream by means of picking many realizations of PWS with a randomized algorithm and later averaging the sum of the samples (i.e., sample average approximation or SAA). There are statistical metrics that allow to find the minimal number of required samples for a given reliability of the result (e.g., Vee in [7]). To further improve precision of the answer, the above sampling can be repeated k times and the median of the averages used. Sampling can be easily parallelized. While the approach is robust, it requires additional processing of repeated samples and extra time for that. Sampling heuristic approximation techniques for reduction of complexity are provided in [1]: Algorithms RAND, K-TOP and CONDENSE. The algorithms have quadratic complexity and accumulate the error with each intermediate step.

There are two main difficulties with sampling for data stream problems. First, we need too many samples for analysis with hard time constraints. Second, sampling typically works slowly for fast streams and many updates of data in the sliding window because repetition of the sampling algorithms is required on every change. Possibly, samples from the past iteration can be reused to update the model, but it is not always possible for the application problem or expensive in practice for data stream models.

Tran et al. developed several approaches to implement queries for uncertain data streams [3–5]. They implemented MIN, MAX, SUM, AVG and COUNT operators for continuously distributed random variables. These are the only authors who showed experimental results among the publications. The authors characterized the final query answers with full distribution for each aggregate. They consider that only a number of moments, such as expectation and variance, of the resulting distribution are not a sufficient answer for many queries [4]. Two algorithms were introduced that use Gaussian mixture models (GMM) and linear functions to approximate resulting PDF or CDF respectively of resulting distribution of the aggregate. Unfortunately, GMM method could not be combined with TEP < 1. When the fast computations are required, *use of CLT* was advised. According to their performance evaluations, GMM method turned out to be very slow for practical purposes. They produced error estimation of intermediate and final query results: They use Chebyshev's inequality to derive upper bounds of errors. The authors also compared their algorithms to classical randomized methods of CDF approximation in all three articles. They call such algorithms "Rand" and they are based on Monte Carlo simulations.

Ge et al. [1] built approximations of resulting CDF by mapping continuous intervals of PDFs to discrete set of points. They introduced a scalable algorithm SERP to obtain fast approximation of resulting distribution. They proposed to partition the original continuous PDF into k intervals of different size, such that CDF$=1/k$ for each interval (weighted sampling algorithm). Thus, only values indicating the boundaries of the intervals are required. Statistical weighted random sampling from each interval was used to compute the output distribution of the SUM operator. They also proposed statistical mode without reproduction of a distribution of the result. They used Azuma-Hoeffding's inequality (does not assume data independence) or Chebyshev's inequality (assumes data independence) to establish upper and lower bounds for the results. Information of their empirical study is not fully comprehensive. Nevertheless, it is clear that the quality of the resulting distribution approximation of SERP is much higher than of random sampling algorithms and statistical mode is faster than SERP from 15 to 50 times in real-life application.

We can conclude that uncertainty is not uniformly understood and can be modeled differently for diverse applications. Computation of complete distributions of uncertain aggregation results can require too much resources for data stream processing. This can become even more difficult when the problem has high precision or other quality objectives.

4 Aggregation of Random Attributes for Stream Processing

After realizing related work in the area of aggregation methods in uncertain databases and streams, we suggest one method of the operation SUM that satisfy our problem of data processing in uncertain data stream. The core of the method is representation of the aggregation result with the six-value-set that is easy to maintain on sliding window change and provide extended information about the sum of uncertain variables. Therefore, we discuss methods that compute lower and upper bounds for our uncertain aggregation result with $O(1)$ complexity in this section. These fast algorithms are suitable for fast uncertain data processing.

Let aggregation be performed on an uncertain attribute A after conditioning and grouping. A is a random vector $\boldsymbol{X} = (X_1, X_2, \ldots, X_k)^T$ that has the size k. Thus, our goal is to effectively compute and maintain aggregation result in $O(1)$ operations on each update of attribute A of tuples or shift of sliding window in data stream. In order to do that, we will store the result of uncertain aggregate sum of tuples in the window in the six-value structure. The structure is updated on changes of data or of the window itself. According to the properties of independent random variables:

$$E[\boldsymbol{X}] = \sum_{i=1}^{k} E[X_i]; \qquad Var[\boldsymbol{X}] = \sum_{i=1}^{k} Var[X_i]. \tag{1}$$

By supplying a threshold $\alpha \in [0.5, 1]$ in the query we get the bounds $P_{L_b} = (1-\alpha)/2 = \alpha', P_{U_b} = 1 - \alpha'$. Probability that the value gets into the interval $[L_b, U_b]$ is α. We need only to find the lower (upper) bound, such that the aggregation result can be below (above) the bound with probability P_{L_b} (P_{U_b}). Next, the remaining parameters L_b, U_b of the aggregation result have to be identified.

Method 1: Direct Quantile. If we know CDF Φ_R of the distribution of the sum of k random variables and its inverse CDF function exists, U_b and L_b can be computed easily as quantiles. For instance, the sum of k normally distributed random variables is a normally distributed random variable and inverse CDF $\Phi^{-1}(X)$ of normal distribution is known.

$$\begin{aligned} Pr(X \le L_b) = \alpha' &\Leftrightarrow L_b = \Phi_R^{-1}(\alpha') \\ Pr(X \ge U_b) = 1 - \alpha' &\Leftrightarrow U_b = \Phi_R^{-1}(1-\alpha'). \end{aligned} \tag{2}$$

We assume that a call to a function has constant time complexity and does not depend on n. We have exactly two subtractions for expectancy and variance in (1) on every tuple that leaves the window or two additions where a new tuple arrives; in either case, two calls to quantile function (2) are performed (for lower and upper bound respectively). Therefore, complexity of this approach is $O(1)$.

It is not as easy for uniform distributions, as their sum is not normally distributed. Formulas of PDF and CDF of a sum of k independent random variables $X_1, X_2, \ldots, X_n$ with $X_i \sim U(a_i, b_i)$ for $i = 1, 2, \ldots, k$ are given in Theorem 1 in [14] and Theorem 2.2 in [15]. Unfortunately, it is hard to derive the inverse CDF for them in practice and even then it seems to be too sophisticated to implement them due to problems with integration bounds. Consequently, it is more promising to obtain bounds with approximation methods for convolutions of uniform and other complex distributions.

Method 2: Chebyshev Bounds. Independently of whether the distribution if known, we can obtain L_b and U_b provided that we know the left part of the Chebyshev's inequality, i.e. probabilities P_{L_b} and P_{U_b} [16]. Probability bounds from both sides are symmetric. Double-sided formula gives the following solution:

$$L_b = \mu - \sqrt{\frac{\sigma^2}{\alpha'}}, \quad U_b = \mu + \sqrt{\frac{\sigma^2}{\alpha'}}. \tag{3}$$

And the single-sided formula gives:

$$L_b = \mu - \sqrt{\frac{\sigma^2}{\alpha'} - \sigma^2}, \quad U_b = \mu + \sqrt{\frac{\sigma^2}{\alpha'} - \sigma^2}. \tag{4}$$

Method 3. Use of MGF/CF and Chernoff Bounds. Chernoff bounds allow to obtain better estimation of U_b and L_b for distributions with known MGF or CF functions than Chebyshev bounds. MGF/CF produces the expected value of the convolution of n random variables as the value of its first derivative at the point $t = 0$.

Let $M_Z(s)$ be an MGF of the sum of random variables. Let us consider upper bounds for the common case $Pr(Z \geq t) \leq \min e^{-st} M_Z(s), \forall s > 0$ [16]. The MGF/CF of a sum of n independent random variables is a product of the MGFs/CFs of the components:

$$U_B : Pr(Z \geq t) \leq \min_{s>0} e^{-st} E\left[\prod_i e^{sX_i}\right] = \min_{s>0} e^{-st} E\left[\prod_i M_{X_i}(s)\right]. \tag{5}$$

We will derive U_B for this method. CF can be employed for cases where MGF does not exist by analogy. Derivation of the lower bound and use of CF are analogous.

One can choose a value of s that gives a convenient form. We can also find an optimal s^* through solution of minimization problem $\frac{d}{ds} e^{-st} M_Z(s) = 0$. Substituting s^* to the initial bound (5), we can find the general form of the optimal bound. We know given probability of the bound $Pr(Z \geq t)$, so we now need to solve the problem (5) against the bound value t^*, s.t. the inequality holds for the given probability $Pr(X \geq t^*) \leq \alpha'$. As with Chebyshev bounds, we use known probabilities P_{L_b} and P_{U_b} that the random value falls below or above the respective threshold. Now we will derive Chernoff bounds for normal distribution family. For the MGF $M_{Z_N}(s)$ of the sum of n random variables $X_i \sim N(\mu_i, \sigma_i^2)$

$$M_{Z_N}(s) = e^{\Sigma_{i=1}^n \mu_i s + \Sigma_{i=1}^n \frac{\sigma_i^2 s^2}{2}}, \text{ and the optimal } s^* = \frac{t - \Sigma_{i=1}^n \mu_i}{\Sigma_{i=1}^n \sigma_i^2}, \tag{6}$$

the upper bound using known reliability threshold α is:

$$U_b = \sum_{i=1}^n \mu_i + \sqrt{-2 \ln \alpha' \sum_{i=1}^n \sigma_i^2}. \tag{7}$$

Derivation of Chernoff bounds for the sum of uniform random variables is more problematic. Let the sum of random variables $X_i \sim U(a_i, b_i)$ have the MGF

$$M_{Z_U}(s) = \frac{\prod_{i=1}^{n} \left(e^{b_i s} - e^{a_i s}\right)}{s^n \prod_{i=1}^{n} (b_i - a_i)}. \quad (8)$$

There is no analytic solution for the minimization problem (8) in this case. Alternatively, we can derive non-optimal, yet more simple formula for s^* taking into account $e^{b_i s} - e^{a_i s} \leq e^{b_i s}$

$$M_{Z_U}(s) \leq M'_{Z_U}(s) = \frac{\prod_{i=1}^{n} e^{b_i s}}{s^n \prod_{i=1}^{n} (b_i - a_i)}. \quad (9)$$

And the optimal solution for the optimization problem (9) with function $M'_{Z_U}(s) \geq M_{Z_U}(s)$ is the following:

$$s^* = n(\sum_{i=1}^{n} b_i - t)^{-1}. \quad (10)$$

Thus, upper bound using known reliability threshold α is deduced similarly:

$$U_b = \mu - \frac{1}{s^*}\left(\ln (s^*)^n + \ln \alpha' + \sum_{i=1}^{n} \ln(b_i - a_i) - \sum_{i=1}^{n} \ln\left(e^{b_i s^*} - e^{a_i s^*}\right)\right). \quad (11)$$

Method 4: Use of CLT for Aggregation. Related works often apply CLT for uncertain aggregation of large number of tuples. The assumption that a set of random numbers satisfy conditions for CLT is commonly made in uncertain data processing. The theorem tells us, however, that it should be applied only if summands meet certain conditions. Then, the sum of uncorrelated random variables can converge to normal distribution and we can directly derive our bounds using the method of direct quantile with complexity $O(1)$:

$$\sum_{i=1}^{n} X_i \sim N\left(\sum_{i=1}^{n} \mu_i, \sum_{i=1}^{n} \sigma_i^2\right). \quad (12)$$

But unless we know a priori that our variables satisfy them, we need to check the conditions. We will show below that CLT appicability check turns out to be complex. It is not worth implementing the check because this is only the test of applicability of CLT, not aggregation itself. Violating CLT conditions does not guarantee convergence of the sum to normal distribution and quantiles may not be valid. This applicability check is usually not mentioned in papers. But we found that our experimental data violated the conditions, so we did not use CLT in our experiments.

There are a number of interconvertible criteria for CLT applicability verification and we consider $\sigma^2 = \sum_{i=1}^{n} \sigma_i^2$ in the conditions below.

1. Lindeberg-Feller condition [17] checks that not a single variance is greatly larger than others. From Lindeberg's condition

$$\text{let } \sigma^2 = \sum_{i=1}^{n} \sigma_i^2, \quad \lim_{n\to\infty} \frac{1}{\sigma^2} \sum_{k=1}^{n} \mathbb{E}\left[(X_k - \mu_k)^2 \cdot \mathbf{1}_{\{|X_k - \mu_k| > \varepsilon\sigma\}}\right] = 0 \tag{13}$$

follows the Feller's condition: $\forall i \in [1, n]$ and given $\delta > 0, \max_i \frac{\sigma_i^2}{\sigma^2} < \delta$. Parameter δ regulates applicability of CLT regarding the quality of convergence. One may imagine it as a ratio of known values, e.g., $\delta = \frac{\sigma^2}{10n}$ is easy to maintain with $O(1)$ complexity. Thus, the choice of δ or comparable parameter for CLT is the measure of applicability of the method. If Lindeberg-Feller condition does not hold, we can not use CLT. It turned out that we need to track the attribute with maximum variance in the window. Even if we employ a structure like priority queue, this tracking task changes our complexity from $O(1)$ to $O(log(n))$ because it dominates CLT aggregation complexity. Approximations like hashing and smart queues can be employed. Nevertheless, we believe it is an unacceptable solution.
2. Lyapunov condition [17]:

$$\forall\delta > 0, \ \frac{1}{\sigma^2} \sum_{i=1}^{n} E\left[|X_i - \mu_i|^{2+\delta}\right] \xrightarrow[n\to\infty]{} 0. \tag{14}$$

Unfortunately, higher central moments or their absolute values are not given in our problem. There is no point in SAA or Monte Carlo simulations, so this approach is unacceptable for our problem.

Proposition 1. *In order to use the criteria, we can maintain actual sum of central moments of the third or higher order (fourth order is more convenient) because moments of higher order can be computed as derivative of the respective order of known MGF/CF at point 0. But complexity of operations turns out to be polinomial in n because any central moment of a random variable will be represented as a nonlinear combination of products of its non-central moments, i.e., the number of summands grows very fast:*

$$\begin{gathered}
E[(X-\mu)^3] = E(X^3) + 2(E(X))^3 - 3E(X)E(X^2), \\
E[(X-\mu)^4] = E(X^4) - 4E(X^3)E(X) + 6E(X^2)(E(X))^2 - 3E(X^4) \\
\textit{and} \\
E[(X_1 + X_2)^3] = E[X_1^3 + 3X_1^2X_2 + 3X_1X_2^2 + X_2^3] \\
= E[X_1^3] + 3\left(E[X_1^2X_2 + x_1X_2^2]\right) + E[X_2^3], \\
E[(X_1 + X_2 + X_3)^3] = E[((X_1 + X_2) + X_3)^3] \\
= E[X_1^3] + E[X_2^3] + E[X_3^3] + 3(E[X_1]E[X_2^2] + E[X_1^2]E[X_2] + E[X_1^2]E[X_3] \\
+ E[X_2^2]E[X_3] + E[X_1]E[X_3^2] + E[X_2]E[X_3^2]) + 6E[X_1]E[X_2]E[X_3], \\
\textit{and so on...}\ .
\end{gathered} \tag{15}$$

3. Berry-Esseen theorem with $0.4097 \leq C \leq 0.7975$ [17]:

$$\max_{z \in \mathbb{R}} \left| Pr\left(\frac{1}{\sigma^2}\sum_{k=1}^{n}(X_k - \mu_k) \leq z\right) - \Phi(z)\right| \leq C\frac{\sum_{k=1}^{n} E(|X_k - \mu_k|^3)}{\sigma^3\sqrt{n}}. \quad (16)$$

   Again, central moments are not available in our problem. For the sum of arbitrary distributions, use of (15) is not effective for the same reasons as for Lyapunov's CLT condition. This approach is unacceptable for our problem as well.
4. Finally, we can check the convergence of the normalized sum to the standard normal distribution using the following four metrics for main parameters of the distribution because all four parameters are constant for any normal distribution:
   - expected value of the sum $\Sigma_{i=1}^{n}\mu_i \to 0$;
   - variance of the sum $\sigma^2 \to 1$;
   - skewness of the sum $E[(X-\mu)^3]/\sigma^3 \to 0$;
   - kurtosis of the sum $E[(X-\mu)^4]/\sigma^4 \to 3$.

   For the sum of arbitrary distributions, these parameters should be examined using (15) and the complexity checks of skewness and kurtosis are far from $O(1)$ for n aggregated random variables, as it was shown above.

Thus, the verification of applicability becomes much more complex than the potential gain from the aggregation with CLT. Nevertheless, we believe that further attention should be paid to CLT application to aggregation of complex joint or truncated random variables with or without TEP. Moreover, there is multivariate CLT for random vectors that can be applied to joint distributions including those with TEP. For a sequence of iid. random vectors $\boldsymbol{X}_i$ with $E(\boldsymbol{X}_i) = \boldsymbol{\mu}$ and $Cov(\boldsymbol{X}_i) = \boldsymbol{\Sigma}$ under the restriction that no random vector dominates, it converges in order α to the following multivariate normal distribution [17]:

$$\sqrt{n}\left(\frac{1}{n}\sum_{i=1}^{n} X_i - \boldsymbol{\mu}\right) \sim_\alpha N(\mathbf{0}, \boldsymbol{\Sigma}). \quad (17)$$

Additional Notes About Uncertain Aggregation. When tuples are updated in the window or window slides, we need only to recompute expectancy, variance and bounds using methods above. First, we update expected value and variance of the sum. Once new tuples enter the window, we need to add the respective attribute values to our existing aggregations. If a number of tuples leave the window, we need to perform subtraction. Afterwards, we recompute the bounds.

Aggregation of truncated random variables looks hard in general. The possibility of deriving the resulting distribution of the convolution is vital. Only truncated uniform distributions transform into uniform distributions and can be aggregated by means of the quantile functions for uniform distributions. For the general case, we can determine upper and lower bounds using Chernoff bounds which are in many cases appropriate for determination of parameters if distributions of independent random variables are not known as well as for discrete distributions.

Aggregation of values with TEP is challenging because they can be mixed random variables and additional work on their aggregation is needed. Moreover, they can be truncated as well. The same reservations as for truncated distributions above are valid. Chebyshev and Chernoff bounds are possible here, because according to properties of independent random variables, expectancy and variance of products of them are $E[X \cdot Y] = E[X] \cdot E[Y]$ and $Var[X \cdot Y] = Var[X] \cdot Var[Y] + (E[Y])^2 Var[X] + (E[X])^2 Var[Y]$ [1].

5 Implementation and Evaluation

The comparison of the methods 1–3 of uncertain aggregation on an example data set was performed. Our interests in evaluation were:

1. to compare performance of uncertain aggregation with deterministic one;
2. to compare quality of bounds of uncertain aggregation methods.

Data set was generated for the experiments in the following way. Expected values for random attributes were taken randomly from the attribute AverageTemperature from the dataset Global Temperature Records [18]. Respective variances have been generated as random numbers with Gaussian normal distribution with variance not greater than 5. Data set for performance test contained 2 million tuples. Comparison of computation time was done for four sliding window sizes and three aggregation methods. Conclusion about quality of the result of aggregation for different methods was made. All experiments were performed using Samsung np900x3d computer with Intel® Core™ i5-2537M Processor, 4 Gb of installed memory and 128 Gb SSD hard drive. Operation system was Debian Linux 9 (stretch) with Linux kernel 3.16.0-4-amd64. The program was built using GCC 6.3.0 compiler and maximum optimization key.

Implementation was based on a PipeFabric C++ framework (license GPL 3) for data stream processing [19,20]. Summary of average time of simple atomic operations is given in Table 1. Maintenance of stochastic aggregation SUM is

Table 1. Averaged time of atomic operations

Aggregated type	Window size	SUM Time, s^{-6}	COUNT Time, s^{-6}
Double	10	0.23	0.23
Double	100	0.23	0.26
Pair of doubles	10	0.22	0.22
Pair of doubles	100	0.23	0.23
Stochastic	10	1.53	0.27
Stochastic	100	1.64	0.27

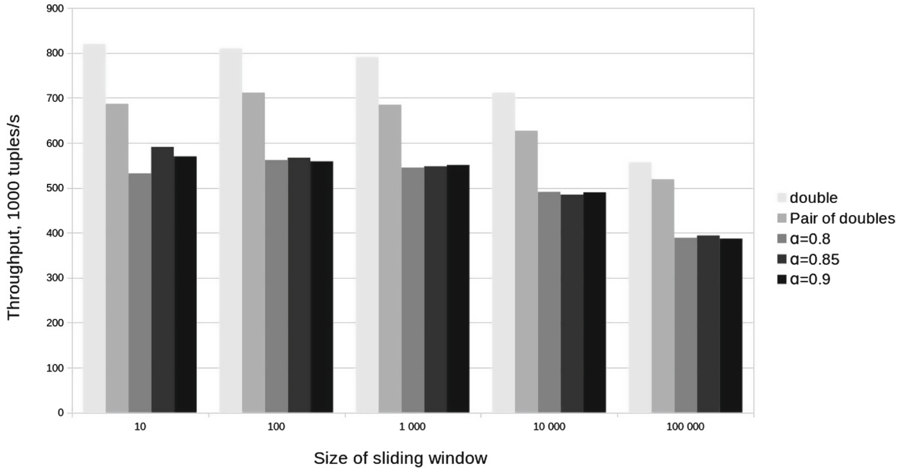


Fig. 1. Results of performance test

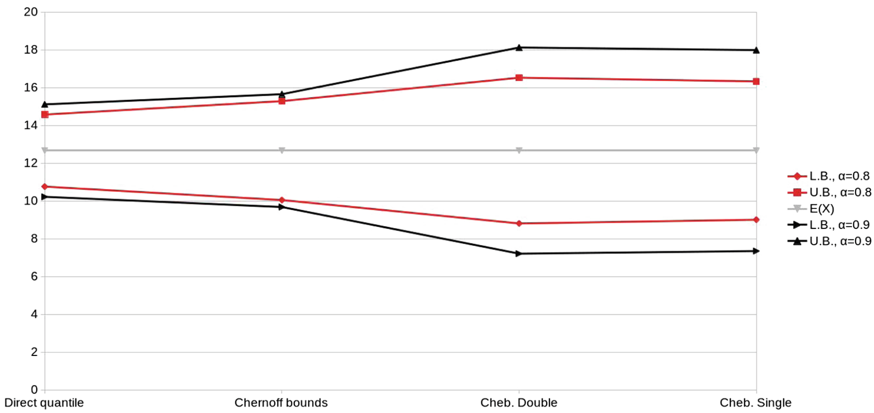


Fig. 2. Comparison of bounds of stochastic aggregation (SUM) methods, $\mu = 12.661$, $\sigma = 1.487844$

several times slower than the deterministic ones. Results of our performance test using our sample data set are shown in Fig. 1. It shows averaged time of three runs of query processing for deterministic and stochastic operator SUM with different window sizes. Uncertain aggregation needs longer time. Nevertheless, time of uncertain SUM is comparable to the time of respective deterministic operator: The speed decrease is about 30% for all window sizes.

Comparison of quality of aggregation with different methods is shown in Fig. 2. Bounds for a sum of values in the window (a normally distributed random variable) were found with several methods for two parameters α. Direct quantile method produced the best aggregation bounds, i.e., the bounds are the narrowest. The second best approximation method – Chernoff bounds – produced very close result. Chebyshev bounds produced clearly worse results. This conforms to the theory. On the opposite, computation of the direct quantile (ca. 1 µs) took more than two times longer on average than computation of Chebyshev bounds (ca. 0.38 µs). This is explained by the need of additional C++ class initialization and a call to the library function in our implementation. Chernoff and Chebyshev bounds need only simple algebraic calculus. Chernoff bound has middle time between Direct and Chebyshev methods (ca. 0.8 µs). All three methods can be used in practice for aggregation. They represent a trade-off between speed and accuracy wherever it is important.

6 Conclusion

Literature review showed, that processing of uncertain data is still an open research problem. At the same time, pace of digital business transformation pushes us to approach uncertain data processing in information systems. Progress in technology promises to realize the advantages.

We have introduced a method for uncertain operation SUM of random variables that suit fast uncertain stream processing. The methods are generic and independent from distribution assumption, so it can be applied to continuous, discrete or mixed types. They may be used in uncertain databases as well. Aggregation results are characterized by means of key moments (expectation and variance) and two additional exact or effectively approximated quantiles that mean lower- and upper bounds for a given threshold. We chose methods that compute bounds for aggregation result with complexity $O(1)$. They include direct quantile, Chebyshev and Chernoff bounds. We provided further information for cases of complex combinations of uncertain data in a query. Derivation of Chernoff bounds was performed for the case of sum of uniformly distributed random variables. In particular, if a tuple is added to the sliding window or deleted from it, the computational cost of updating aggregations is $O(1)$.

We proved in our experiments that aggregations can be efficiently maintained over time that suit real-time data processing in data streams. Experimental evaluation with SUM aggregation function and normally distributed random variables was done. The results show that the speed is comparable with aggregation of deterministic values (average overhead was always less than 1.5 times in our implementation).

Beyond applicability of the method to modern problems with uncertain data, existing algorithms in business software can be upgraded to use uncertain data without rewriting the whole software. Polymorphism principle of object-oriented programming can employ aggregation algorithms for uncertain data types to complement existing code. Applicability to all types of random variables enables holistic approach to uncertain data processing due to similarity to deterministic data algorithms: Deterministic operator SUM is equally applicable to any numerical variables (e.g. real or integer). In the same way, uncertain operator SUM is suitable to all types of random variables. Additionally, as there is no change in syntax and mathematical notations, database query languages can benefit from that.

We evaluated the use of CLT for uncertain aggregation. Caution should be taken because certain criteria must be met for data in the sliding window. It is not easy to check criteria fast unless the design of the stream supplies statistical moments of higher order. That means, that CLT should not be used unless the criteria are met or there is a priori certainty (based on prior experiments or theoretical foundations) that CLT is suitable for the data set. Otherwise, use of CLT seems inefficient because it requires a lot of computations for applicability verification of CLT. Further research may be required towards application of CLT, especially for multidimensional uncertain cases.

Development of the uncertain SUM operator is the first step in development of a family of uncertain aggregation operators. Other aggregate operations that return a single number in deterministic relational databases (like MAX, MIN), can return a set of tuples for uncertain aggregation depending on uncertainty model. We will develop techniques for other uncertain aggregation operators in future. Where the result of aggregation is a set, we suggest that methods should enable a balance between speed and accuracy, give a suitable result of the query in terms of problem formulation and be more computationally efficient than those based on both complex mathematical methods like approximating full distributions using GMMs and those that approximate intermediary distributions with sampling-based methods like Monte-Carlo simulations. Based on the research, the following directions of future work can be considered:

– design relational algebra model for other uncertain aggregation operators and create prototypes,
– further study of CLT utilization for fast aggregation of uncertain data, and
– further research of aggregation of multi-dimensional or joint random attributes and TEP.

References

1. Ge, T., Zdonik, S.: Handling uncertain data in array database systems. In: IEEE 24th International Conference on Data Engineering (ICDE 2008), pp. 1140–1149. IEEE, Cancun (2008). https://doi.org/10.1109/ICDE.2008.4497523
2. Wang, Y., Li, X., Li, X., Wang, Y.: A survey of queries over uncertain data. Knowl. Inf. Syst. **37**(3), 485–530 (2013). https://doi.org/10.1007/s10115-013-0638-6
3. Tran, T., McGregor, A., Diao, Y., Peng, L., Liu, A.: Conditioning and aggregating uncertain data streams: going beyond expectations. In: The 36th International Conference on Very Large Data Bases (VLDB 2010), vol. 3, no. 1, pp. 1302–1313. VLBD, Singapore (2010)
4. Tran, T., Peng, P., Diao, Y., McGregor, A., Liu, A.: CLARO: modeling and processing uncertain data streams. VLDB J. **21**(5), 651–676 (2012). https://doi.org/10.1007/s00778-011-0261-7
5. Tran, T., Peng, L., Li, B., Diao, Y., Liu, A.: PODS: a new model and processing algorithms for uncertain data streams. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, pp. 159–170. ACM, New York (2010). https://doi.org/10.1145/1807167.1807187

6. Murthy, R., Ikeda, R., Widom, J.: Making aggregation work in uncertain and probabilistic databases. IEEE Trans. Knowl. Data Eng. **23**(8), 1261–1273 (2011). https://doi.org/10.1109/TKDE.2010.166
7. Aggarwal, C. (ed.): Managing and Mining Uncertain Data. Springer, Heidelberg (2009). https://doi.org/10.1007/978-0-387-09690-2
8. Dallachiesa, M., Jacques-Silva, G., Gedik, B., Wu, K.-L., Palpanas, T.: Sliding windows over uncertain data streams. Knowl. Inf. Syst. **45**(1), 159–190 (2015). https://doi.org/10.1007/s10115-014-0804-5
9. Dezfuli, M.G., Haghjoo, M.S.: Xtream: a system for continuous querying over uncertain data streams. In: Hüllermeier, E., Link, S., Fober, T., Seeger, B. (eds.) SUM 2012. LNCS (LNAI), vol. 7520, pp. 1–15. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33362-0_1
10. Cheng, R., Kalashnikov, D., Prabhakar, S.: Evaluating probabilistic queries over imprecise data. In: Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pp. 551–562. ACM, New York (2003). https://doi.org/10.1145/872757.872823
11. Jayram, T.S., Kale, S., Vee, E.: Efficient aggregation algorithms for probabilistic data. In: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 346–355. Society for Industrial and Applied Mathematics Philadelphia (2007)
12. Jin, C., Yi, K., Chen, L., Yu, J., Lin, X.: Sliding-window top-k queries on uncertain streams. VLDB J. **19**(3), 411–435 (2010). https://doi.org/10.1007/s00778-009-0171-0
13. Hua, M., Pei, J., Zhang, W., Lin, X.: Ranking queries on uncertain data: a probabilistic threshold approach. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 673–686. ACM, New York (2008). https://doi.org/10.1145/1376616.1376685
14. Killmann, F., von Collani, E.: A note on the convolution of the uniform and related distributions and their use in quality control. Econ. Qual. Control **16**(1), 17–41 (2001)
15. Bradley, D., Gupta, R.: On the distribution of the sum of n non-identically distributed uniform random variables. Ann. Inst. Stat. Math. **54**(3), 689–700 (2002). https://doi.org/10.1023/A:102248371
16. Mitzenmacher, M., Upfal, E.: Probability and Computing - Randomized Algorithms and Probabilistic Analysis. Cambridge University Press, Cambridge (2005)
17. Spanos, A.: Probability Theory and Statistical Inference. Econometric Modeling with Observational Data. Cambridge University Press, Cambridge (1999)
18. Climate Change: Earth Surface Temperature Data. https://www.kaggle.com/berkeleyearth/climate-change-earth-surface-temperature-data
19. Pfabrik, C++ framework. https://github.com/dbis-ilm/pipefabric
20. Saleh, O., Sattler, K.-U.: The PipeFlow approach. In: 9th ACM International Conference on Distributed Event-Based Systems (DEBS 2015), pp. 326–327. ACM, New York (2015). https://doi.org/10.1145/2675743.2776763

A Method for Operationalizing Service-Dominant Business Models into Conceptual Process Models

Bambang Suratno(✉), Baris Ozkan, Oktay Turetken, and Paul Grefen

Eindhoven University of Technology, Eindhoven, The Netherlands
{b.suratno, b.ozkan, o.turetken, p.w.p.j.grefen}@tue.nl

Abstract. Service Dominant Logic (SDL) is a mindset that creates many opportunities for designing and innovating networked-business models. One general problem in business model design is the limited support that would guide the operationalization of business models into process-aware information systems (PAIS). This paper proposes a method (namely, SDBMOM) for the operationalization of service-dominant business models into conceptual business process models in BPMN as a first step to the business model implementation. SDBMOM is developed as part of BASE/X business engineering framework that aims to provide conceptual and methodological support for adopting SDL in the end-to-end business design and operationalization. In the development of the SDBMOM, we follow the design-science research methodology, where we defined the problem and set of design goals, developed and designed our artifact, and demonstrated its use. SDBMOM is conceptualized and characterized in the BASE/X framework and presented as a stepwise method that relies on the well-known process modeling approach - BPMN. In this paper, we use an illustrative scenario of travelling service (i.e. TraXP eXecutive) to demonstrate the validity of the method. A structured method which ensures the operationalization of business models as a whole and delineates the operational scope and their boundaries for each value co-creating organization will provide the basis for the specification of conceptual and executable process models, and eventually the implementation as a process-aware information system.

Keywords: Service-dominant business model · Operationalization
Process model · Service composition

1 Introduction

Service Dominant Logic (SDL) is a marketing-grounded mindset that has emerged and evolved for capturing and extending a convergence of the perspectives on the evolution of value and exchange [1]. Specifically, SDL shifts the focus of business away from the production and distribution of products in the form of goods or traditional services (goods-dominant logic) toward the concept of "service", the application of knowledge and skills, as the basis of all exchange in which service is exchanged for service [1].

While the emergence of SDL has led a movement in marketing research, providing a holistic view of value and exchange, SDL influenced many business domains and

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 133–148, 2018.
https://doi.org/10.1007/978-3-319-94214-8_9

also other research fields such as services science [2, 3]. It has been used as a logical framework for capturing and understanding the business and the shift in business thinking and innovation. It emphasizes integrated customer solutions over products, relations and experiences over transactions, network-centric co-created value that emerge in-context over provider-centric value offerings [4]. Online Shopping, Car Sharing, Online Food Delivery, On-demand Music Streaming business domains provide many everyday life examples of an information system supported businesses that represent this shift which is well captured by SDL.

Despite the diverse opportunities for business, the practical implications of SDL is limited to a set of generic normative guidelines [5]. Therefore, a conceptual and methodological support for adopting SDL in their business processes and mapping their business decisions to the SDL mindset is needed by enterprises. Additionally, a good business strategy is one that can be effectively implemented [6]. Thus, enterprises design business models as a template to implement their strategy [7]. A business model is defined as a way for an organization along with its providers and partners to create value for all its stakeholders [8]. Taking a broader perspective, a business model is viewed as a story that explains how an enterprise works [9]. From another functional point of view, the business model concept bridges the business strategy to information system (IS) support [10]. Therefore, an enterprise IS needs to be aware of the business processes and their organizational context (process-aware) [11].

To address these needs, we propose a structured method to operationalize service dominant business models, which we refer to as "SDBMOM". The concepts in this method is mainly based on a business engineering framework that is specifically designed for SDL business settings (i.e. BASE/X) [12, 13]. BASE/X is organized on a business pyramid that comprises four interdependent layers: Business Strategy, Business Model, Service Composition, and Business Services [12, 13]. BASE/X aims business agility as the result of rapid-design of business models by utilizing business services. Business services are the slowly evolving capabilities that enterprises build over time in accordance with their business strategy. BASE/X introduced SDBM/R as a service dominant business model template [6], which is a high-level business model template that is designed to address the SDL requirements. SD business model uses service compositions for it operationalization, which are specific arrangements of business services that realize the value offering in business models [6, 13]. We take a top-down view of operationalization as part of implementation, in which the focus is on the process of transforming a high-level business model into a set of business processes – eventually in an executable form - following a structured method.

BASE/X has addressed the business strategy [12] and business model layers [6]. However, while the concepts and relationships in BASE/X business pyramid has been established, there is no structured method to operationalize it. As the realization of these layers is in operational level, the service compositions need to be able to bridge two different domains (i.e. business and operation). It need to align the business models that represents the high-level business decisions and the business process models that explain how to conduct the business in everyday situation. Therefore, in continuation of this service-dominant business engineering line of research, we present a method for the operationalization of the service dominant business models into conceptual process models, which is guided by the following research question: *How can we develop a* *method that will operationalize service-dominant business models into conceptual business processes in the form of service compositions given a set of business services?*

Accordingly, to develop SDBMOM, we first elaborate on the concepts in the service-dominant business model operationalization process. Next, we explain the stepwise method that we developed for operationalizing service dominant business models at a conceptual level. We followed the design science research methodology [14] in developing our method. We demonstrate the application of the SDBMOM in an illustrative scenario [15] of high-class travelling service (i.e. TraXP eXecutive).

The remainder of the paper is structured as follows. Section 2 gives a background on the business engineering framework and an overview of the related work. In Sect. 3, we describe the method that we proposed by describing the concept and the transformation process. An illustrative scenario of our approach in a business case is presented in Sect. 4. Finally, we conclude in Sect. 5 with an outlook for future work.

2 Background and Related Work

In this section, the background for service-dominant business model operationalization (SDBMO) is given and related literature on business model operationalization is reviewed.

2.1 Service Dominant Business Engineering Framework (BASE/X)

Service Dominant Business Engineering Framework (BASE/X) follows a holistic and end-to-end view of a service dominant business system which is centered on four core components: business strategy, business models, service compositions (or "conceptual business processes") and business services (Fig. 1).

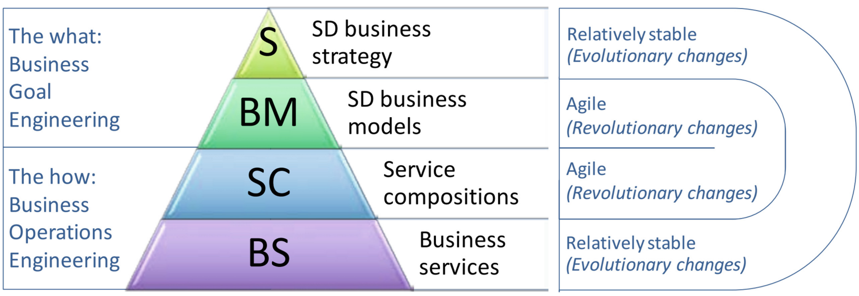


Fig. 1. Business pyramid- adapted from [6]

Accordingly, the business strategy and the business model together represent the driving idea and the goals (what?) of the business, whereas the service compositions and the business services represent the operational aspects, thus the activities and the competences (how?) for putting the business idea and goals into effect, following the SDL mindset. The main premise of this framework is that the extent of an organization agility is determined by its rapid switching capability between business model and service composition combinations, which is essential for the participation in value co-creation processes for the continuous service provision in concert with its relatively slow-evolving strategy and business services. Following this premise, service composition in this framework functions as conceptual business processes which operationalize service dominant business models by composing the business services [6, 12, 13, 16]. However, while BASE/X conceptualizes service compositions for the operationalization of SDBM/R models, it does not offer a structured method for this purpose.

2.2 Business Model Operationalization

Business model operationalization (BMO) involves the realization of business models through derivation of their business process models and information system (IS) components [8]. A process model is an organization of business roles and activities to achieve a particular goal. From the top-down business model operationalization perspective, the process goals and the arrangements of the related business operations should follow from a business model [8, 17, 18]. Subsequently, the IS components are developed and the processes models are executed. In service-dominant business modelling literature, service compositions refer to the process models [6].

Studies in the BMO literature propose diverse approaches employed to guide the business model transformation process [19]. Among the approaches applied, the Model Driven Architecture (MDA) and Service-Oriented (SO) based frameworks stand out in terms of the number of methods that take them as a basis. These approaches require different abstraction levels for the transformation process. For example, the MDA framework considers three models: computer independent model (CIM), platform independent model (PIM), and platform-specific model (PSM) [20, 21]. Based on these frameworks, several studies have proposed methods for the alignment of business models and information systems, which can be realized via process models. Below, we briefly mention about the notable ones.

Fayoumi and Laucopoulos [22] present a set of conceptual modeling steps, which transform a high-level business goal model into a process model in nine steps. The method suggests tools for a systematic analysis that leads to identifying the necessary IS services as a part of it. Di Valentin et al. [23] introduce a method for transforming business models into process models through the use of a component-based model. This BM is subsequently transformed using a four-view process template. Ulmer et al. [24] introduce a metamodel-centric methodology for business process management by integrating concepts from the MDA. This methodology allows business analysts to develop graphical conceptual model, in accordance with a formalized meta-model. De Castro et al. [25, 26] developed Service-Oriented Development Method (SOD-M) that includes the transformation of a value model by following the MDA approach. The authors represent the CIM level by using Business Process Model and Notation (BPMN) [27] for representing business processes, and by using value model for identifying services in the business perspective. Rhazali et al. [28–31] propose a CIM to PIM transformation method with a set of transformation rules using a graphical presentation. This approach provides a solution to the problem of transforming value models represented at CIM level to analysis and design models, modeled at PIM level.

The abovementioned approaches, however, do not take service-dominant business models as input to their transformation process, which requires a network of organizations including the customer to collaborate for the co-creation of the value.

3 Research Design

In this study, we have followed a design science research methodology [14], as our primary goal is to develop a new IS artefact, which we refer to as service-dominant business model operationalization method (SDBMOM). Specifically, we followed design science research methodology (DSRM) process model proposed in [32]. The model of this iterative process is given in Fig. 2.

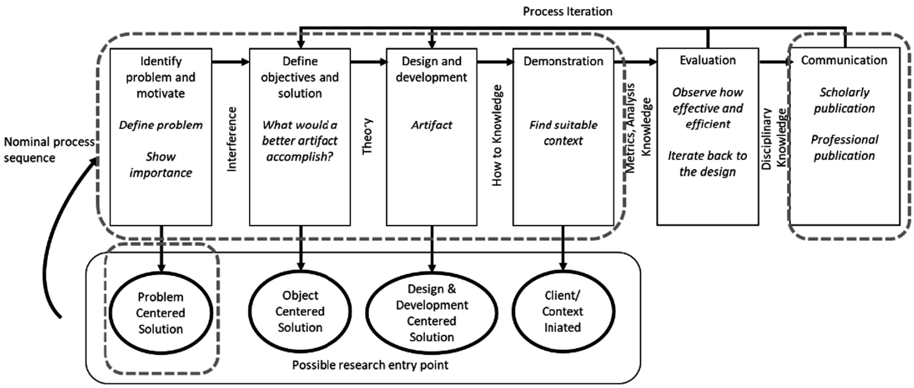


Fig. 2. DSRM process model adapted from [32]

Accordingly, the DSRM process we followed and presented in this study is a problem centered initiative that has completed the first four steps of its research in its initial iteration, and communicating the research on the initial version of SDBMOM. The process involved the following: identifying the problem, defining objectives, designing/developing a satisfactory model, and demonstrating the method in a suitable context to examine its validity and utility [32]. The descriptions for each step completed in the first iteration presented in this study are given in Table 1.

4 SDBMOM

In this section, we present the Service Dominant Business Model Operationalization Method (SDBMOM). Figure 3 presents the expected inputs and output of the method. SDBMOM takes as an input the *Business Model* represented in *SDBM/R,* which is realized by the customer as explained in the *Customer Experience* and uses the *Business Services* of all the actors participating in the business model. The operationalization is performed at a conceptual level; thus, the method produces conceptual

Table 1. DSRM steps in SDBMOM development

DSRM process step	Step description for SDBMOM design process iteration
Identify problem & motivate	(Problem) How can we develop a method that will operationalize service-dominant business models? (Motivation) so that enterprises can realize and leverage SDL mindset by operationalizing of their business models in a systematic and agile way Perform Systematic Literature Review
Define objectives of a solution	Objectives (Requirements) O1 Integrate SDBMOM into the BASE/X Framework O2. Follow SDL mindset and its premises O3. Focus on short term and evolutionary aspects (business agility), thus aim conceptual business processes in the form of service compositions given a set of business services. O4. Follow established BPM approaches and tools
Design & development	The artifact is SDBMOM which outputs conceptual process models in BPMN as collaboration diagrams given a business model in SDBM/R, customer experience and the business services of all actors. The model elements are activities, control flows, actors as roles and messaging between all actors including the customer
Demonstration	SDBMOM is instantiated and demonstrated in an illustrative scenario of travelling business model operationalization case
Communication	This paper

process models (service compositions) that contain and organize a set of business services. The method uses BPMN 2.0 notation for the representation of conceptual process models [27].

(Service-Dominant) Business Model (i.e. SDBM/R) is a tool for business modelling that has a network-centric design at its core, allowing the composition of service design in multi-party business networks [6]. The SDBM/R consists of four layers: Co-produced value-in-use, actor value proposition, actor coproduction activity, and actor cost/benefit. Each party in the network is called an actor, which integrates its competences and resources into complex services. The radar distinguishes between the types of actors: The first slice is for the customer, the second is for the focal

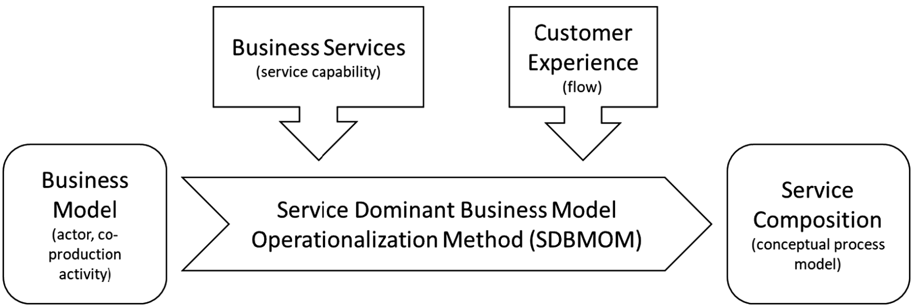


Fig. 3. SDBMO design process

organization (the "primary integrator" in SDL terminology), and then the rest are for the core and enriching parties involved in the creation of the value-in-use. Accordingly, the "bullseye" represent value-in-use which emerges as the actor value propositions are realized. The co-production activity defines the high-level activity statements that each actor performs in the business for achieving its value proposition. All actors has at least one co-production activity. Cost/benefits defines the financial and non-financial expenses/gains of the actors that relate to service provision. In Fig. 4, A1 represents customer, A2 represents focal organization, and A3–A4 represent other parties.

Customer Experience. An SDBM/R always comes with a statement of Customer Experience. Customer experience is an informal customer-oriented operational scenario that contains information on how value is perceived and realized by the customer (similar to "service experience" and "service encounter" terms used in the marketing domain). It contains the customer's activities and his/her interactions with the service composition in a time-frame. The customer's value-in-use is brought about by the customer-business actor and business actor-business actor interactions.

Business Service. Business services the micro-specialized competences (complex activity and resource bundles) of an enterprise that are reusable across the business models and tightly connected to the business strategy. They enable the actor's co-production activities in a SDBM/R model. BASE/X and SDBMOM method distinguishes the business services that interact with the customer and that interact with a partner termed as *Interacting* and *Non-interacting* business services.

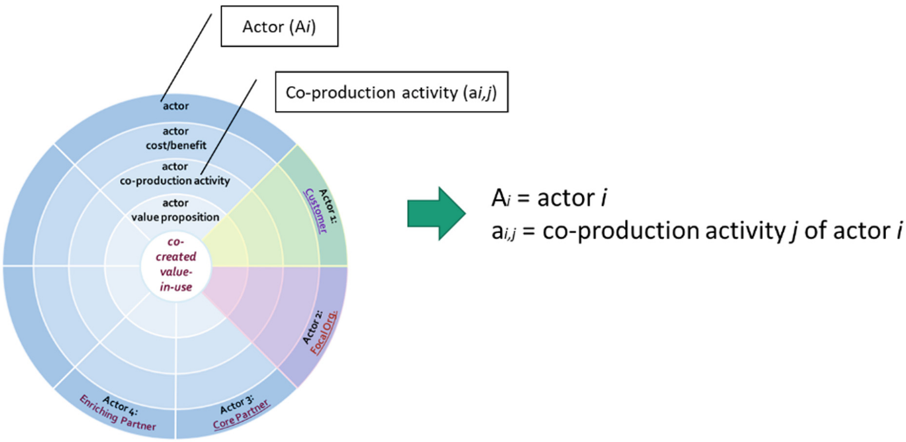


Fig. 4. Actor and co-production activity in SDBM/R

Service Composition (Conceptual Process Model). A service composition is a special type conceptual process model, a specific arrangement of business services to fulfill a business model. At service time, business services are micro specialized activities performed in the service process by combining competences and resources. Service compositions are represented in BPMN 2.0. Accordingly, actors (including the customer) and the activities conducted by actors in SDBM/R are represented by activities, roles and lanes of BPMN notation in a service composition. The workflow in a service composition is designed based on the customer experience using gateways and flows. All interactions between actors are represented as message flows. Together, the customer experience process and the service composition (as two pools) is a BPMN collaboration diagram [33].

The operationalization of a business model can aim for different purposes and business layers, resulting in varying details in its operational artifact. In-line with the literature and the BASE/X requirements, in our case the business model is operationalized through three consecutive transformations (see Fig. 5) with different purposes.

The first transformation (T1) is performed at the conceptual level and from a business viewpoint. The aim of T1 is the operationalization of a SDBM as a whole in which it is ensured that the value-in-use is co-created, business network actors perform their co-production activities using their business services and customer experiences the service as designed by interacting with the other actors. The output is a conceptual process model which serves as an operational contract between all actors delineating each actor's operational scope and showing information flows between all actors. The second level of transformation (T2) is the transformation of conceptual process models

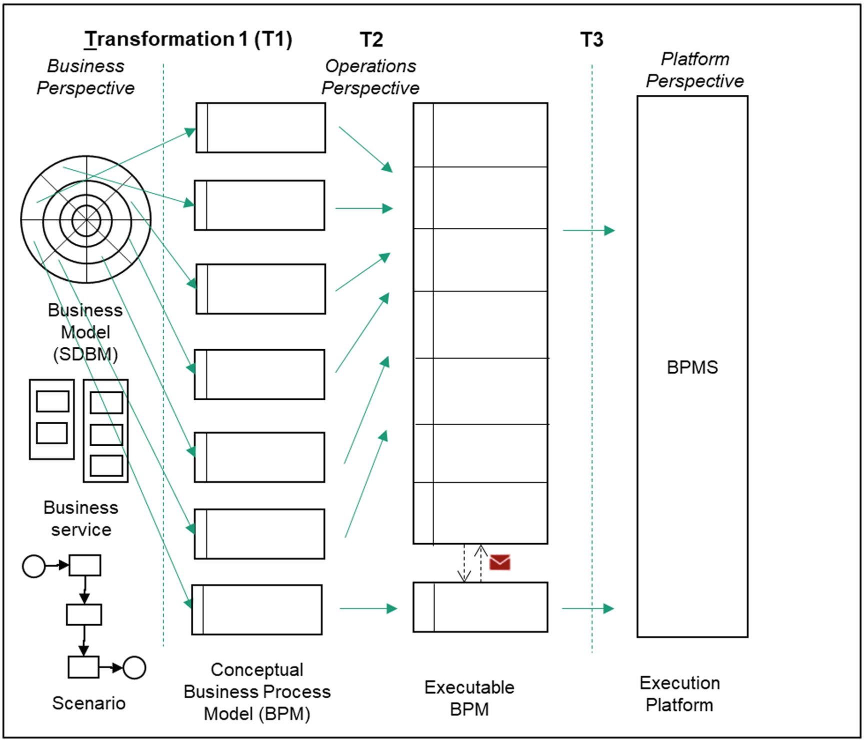


Fig. 5. SD business model operationalization process as transformations

into executable process models to be executed by a process-aware BPM Platform. In the last step (T3), the executable processes are automatically transformed into executables by the containers of a BPM system platform. SDBMOM applies a *T1 transformation* where a business model is transformed into business service compositions in the form of conceptual process models.

4.1 Method Steps

Our method follows a collaborative perspective where actors involved work together in this SDBMO design process. The steps in the SDBMO are as follows:

Step 1: Identify *customer activities* and *interactions* from the customer experience. Create a two-pool collaboration diagram that includes the customer experience and the service composition as separate interacting processes (Fig. 6).

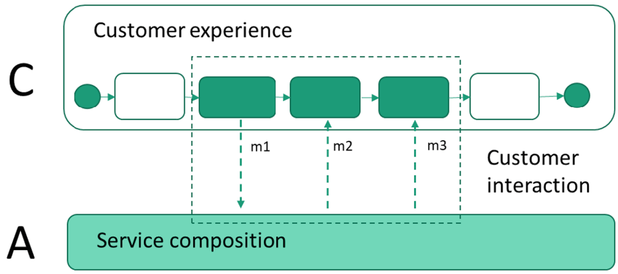


Fig. 6. Service composition after Step 1

Step 2: Identify the *actors* and their *co-production activities* from the business models. Identify *business services* that will enable the co-production activities of each actor from their business service catalogues. From the selected services, identify services that will interact with the customer (*interacting business services*). The design of the *sequence/flow* of interacting business services will match the sequence of the customer interaction. Add *non-interacting business services* into the composition. On the left side of Fig. 7, the organization of business services which support the business model and enables co-production activity is shown. The dark colored boxes represent *interacting* business services while the light colored boxes represent the *non-interacting* ones.

Step 3: Create lanes for each actor and reallocate the business services into corresponding actor lanes. The step results in a complete *service composition* (the model on the right side of Fig. 7).

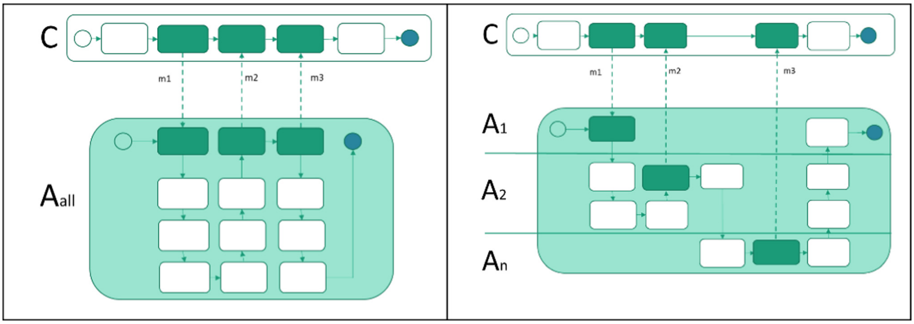


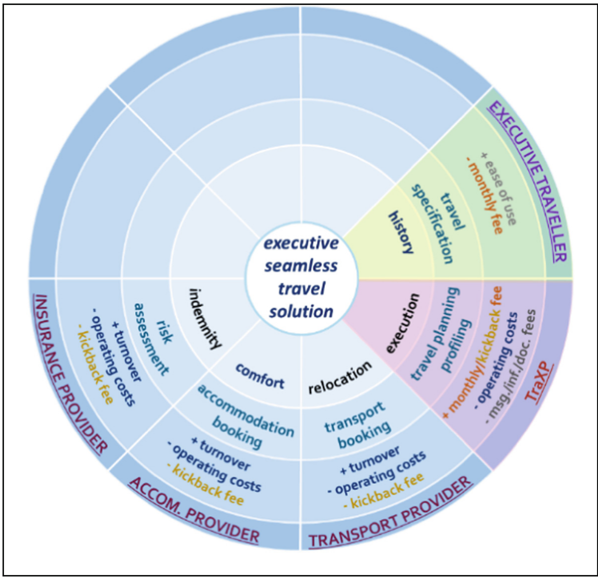
Fig. 7. Service composition after Step 2 and Step 3 (Color figure online)

5 Demonstration of SDBMOM

5.1 SD Business Model and Customer Experience

For demonstration purposes, we use a business case of a virtual travel agency company – TraXP [34] which is the focal organization for the business model *eXecutive*. This business model aims to provide company executives a “seamless travel experience” service. In this model (Fig. 8), the executive traveler wishes to travel without the burden of organizing the trip.

A total of five actors involved (including the executive traveler as a customer), each provide their co-production activity. In Fig. 8, to save space, only an excerpt from the



John Smith is the CFO of a multi-national industry organization. He travels frequently for his job – for which he is an executive customer of TraXP. In a Monday morning meeting in Amsterdam about a new stock emission, John decides that he needs to be on Wall Street in New York City at 10am on Thursday the same week. With his mobile phone, he sends a text message to a dedicated TraXP number saying,” Wall Street NYC USA 10am - 2pm this Thu”. Ten minutes later he receives a text message from TraXP saying,” For NYC: pick-up at home 10:30 this Wed, back at office at 9:30 this Fri”. The trip is completely settled now for John - TraXP knows his preferences…

Fig. 8. TraXP *eXecutive* SDBM/R and customer experience statement

eXecutive business model is provided. The reader is referred to [34] for the full description of the customer experience.

5.2 SDBMOM Application

By using the information provided in the eXecutive business model and the customer experience statement, modelers from all actor organization collaborate to apply SDBMOM:

Step 1. The customer activities and the customer's interaction with service composition are extracted from the customer experience (see Table 2).

During extraction of information, it is possible that business process modeler and domain expert are necessary to be involved in order to formally translate the activities according to the BPMN and to avoid missing activities. In Table 2, information extracted from the customer experience shows that there are six main customer activities and customer interactions. By using this information, a collaboration diagram between executive traveler and TraXP/X service composition can be generated (see Fig. 9).

Step 2. Business services that enable co-production activities of each actor in TraXP are selected from actors' service catalog.

We assume that each actor has a catalogue of business services that are ready to be deployed, and that they can be invited into this collaboration (except for customer, which doesn't necessarily own business services themselves but can still do a co-production activity by providing information for other actor business services). In practice, situations can also occur where actors discover the necessity to create new business services or re-design them to cope with the new business opportunities. The selected business services (see Table 3) are the input to the service composition according to TraXP/X customer activities.

Figure 10 shows a collaboration diagram, which several business services interact with customer, while the other do not. The interaction guides the flow of business process.

Step 3. Next, we reallocate the business services to the actors. Then, we get a complete service composition for TraXP eXecutive (see Fig. 11).

The TraXP/X service composition (SC) shown in Fig. 11 serves as a conceptual process model for TraXP eXecutive business model (BM) in a 'happy' scenario. In this step, instead of only a sequence of activities (or business services), some simple logic, such as the booking of hotel should happen in the same time with the booking of airline ticket and other transport services, can be represented using AND gateway (BPMN). Thus, it enriches the SC in order to serve as a feasible business process model.

Furthermore, it is shown that the traceability of the process model to the business model can also be achieved. The roles in the SC corresponds with the actors in the BM. All necessary activities of the actors are present in the SC. Moreover, aligned with the SDL mindset where different actors act as single virtual enterprises, can also be represented in these steps. Thus, by using this SC as a blueprint for integration, each actor can then design how they will deliver their business services by themselves. If it is done, then a 'full' scenario can be presented. Further elaboration to make the SC into a full scenario or an executable process models is included in the T2 transformation.

Table 2. Customer experience in TraXP *eXecutive*

No	Customer experience statement	Customer activity	Customer interaction
1	With his mobile phone, he sends a text message to a dedicated TraXP number saying, "Wall Street NYC USA 10am - 2 pm this Thu"	Customer request travel and give the travel specification	Customer request travel via mobile phone
2	Ten minutes later he receives a text message from TraXP saying, "For NYC: pick-up at home 10:30 this Wed, back at office at 9:30 this Fri"		Customer receive notification of departure from TraXP
3	On Wednesday, John kisses his wife goodbye at 10:30 and steps into a taxi that has just arrived. The taxi brings him to the airport - John doesn't care which airport	Customer is picked up by taxi at home and then go to the airport	
4	Having arrived at the airport, John receives a text message saying, "Check-in at desk 56"		Customer receive notification of check in from TraXP
5	He checks in - John doesn't really care which airline. The plane fly him to some airport near NYC	Customer checks in to plane and then fly to destination airport	
6	After disembarking at this airport, he gets a text message saying, "Taxi pick-up at Exit 4"		Customer receive notification of pick up at the destination airport
7	He is greeted by a driver holding a sign with his name at Exit 4. The taxi brings him to a hotel	Customer is picked up by taxi at the airport and then go to the hotel	
8	The last text message that day says, "Taxi pick-up 09:30 tomorrow morning"		Customer receive notification of pick up at the hotel
9	The next morning, the taxi brings him to Wall Street	Customer is picked up by taxi at the hotel and then go to the travel destination	
10	The way back home proceeds in a similar way	Return home	Customer receive notification of pick up to return home

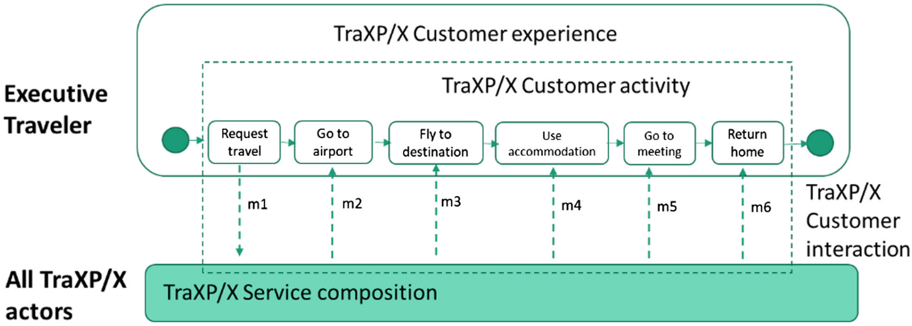


Fig. 9. Output from SDBMOM Step 1 applied to TraXP eXecutive

Table 3. Selected business services for TraXP/X

Actor	Co-production activity	Selected business service
Executive traveler	Travel specification	*Request travel*
TraXP	Profiling	Profile management
	Travel planning	Travel planning optimization
Insurance provider	Risk assessment	Risk assessment
		Insure
Transport provider	Transport booking	Airport transport
		Hotel transport
		City transport
		Home transport
Accommodation provider	Accommodation booking	Hotel accommodation

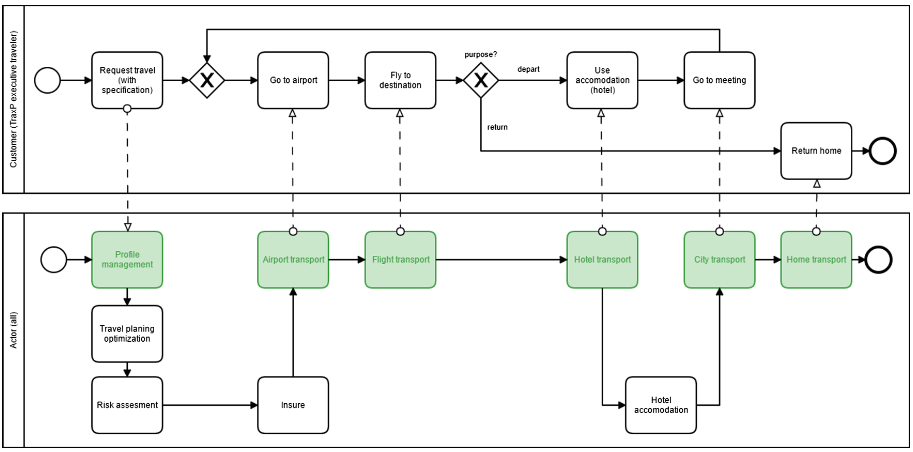


Fig. 10. Step 2 of applied SDBMO method on TraXP eXecutive business model

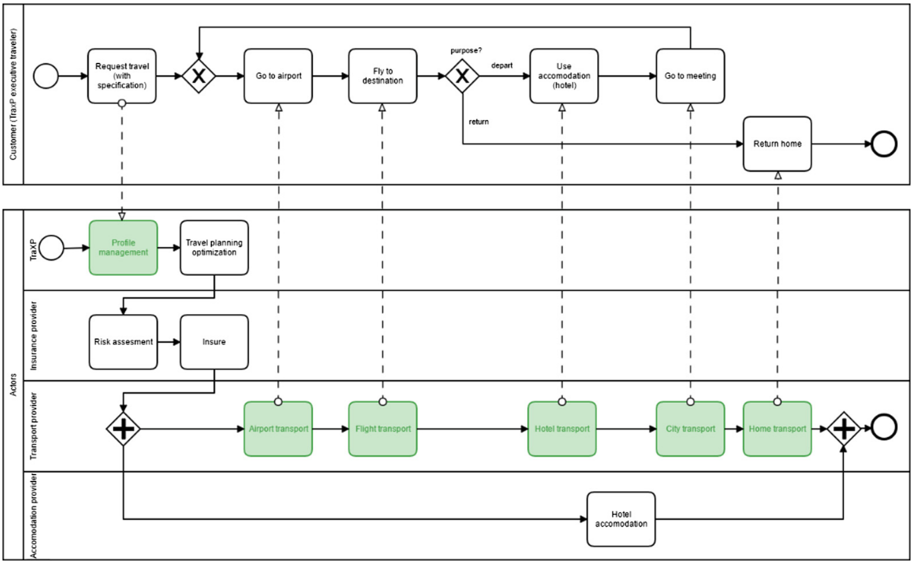


Fig. 11. TraXP/X service composition for TraXP eXecutive

6 Conclusion and Future Work

In this paper we have introduced Service Dominant Business Model Operationalization Method (SDBMOM). The contribution of this paper is twofold: first, the conceptualization of the service compositions layer and its characterization in the business engineering framework by elaborating its relation with the other framework concepts; second, a structured method for the operationalization of service-dominant business models as conceptual business process models.

An assumption of SDBMOM is that the method is applied to SD business models in a top-down and holistic way and the use of model is limited for bottom-up operationalization contexts. The benefit of this approach is that the entire business model is operationalized first and then each actor's scope is defined in swim lanes. Another approach would be a bottom-up approach which require that each the actor organizations collaborating process are identified first and then the interactions between the processes are identified. Another improvement opportunity is that the actor swim lanes of the service composition are converted into pools such that the interaction (message flow) between actor processes are defined in an additional step to our method. We consider these as design goals for the improved version of our method in the next design iteration. The method will also provide guidance for the trajectory of conceptual process models to executable process models as applications in process-aware information systems. Our future work will involve the application of our method in other real-life business cases to evaluate its validity and utility.

References

1. Vargo, S.L., Lusch, R.F., Archpru Akaka, M., He, Y.: Service-dominant logic. In: Malhotra, N.K. (ed.) Review of Marketing Research, pp. 125–167. Emerald Group Publishing Limited (2010)
2. Poels, G.: A conceptual model of service exchange in service-dominant logic. In: Morin, J.-H., Ralyté, J., Snene, M. (eds.) IESS 2010. LNBIP, vol. 53, pp. 224–238. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14319-9_18
3. Spohrer, J., Vargo, S.L., Caswell, N., Maglio, P.P.: The service system is the basic abstraction of service science. In: HICSS, p. 104. IEEE (2008)
4. Lusch, R.F., Nambisan, S.: Service innovation: a service-dominant logic perspective. MIS Q. **39**, 155–175 (2015)
5. Lusch, R.F., Vargo, S.L.: Service-dominant logic as a foundation for a general theory. In: The Service-dominant Logic of Marketing: Dialog, Debate, and Directions, p. 415. M.E. Sharpe (2006)
6. Turetken, O., Grefen, P.: Designing service-dominant business models. In: ECIS 2017 Proceedings, pp. 2218–2233. AISeL (2017)
7. Zott, C., Amit, R.: The business model: a theoretically anchored robust construct for strategic analysis. Strateg. Organ. **11**, 403–411 (2013)
8. Al-Debei, M.M., Avison, D.: Developing a unified framework of the business model concept. Eur. J. Inf. Syst. **19**, 359–376 (2010)
9. Magretta, J.: Why business models matter. Harv. Bus. Rev. **80**, 86–92 (2002)
10. Hedman, J., Kalling, T.: The business model concept: theoretical underpinnings and empirical illustrations. Eur. J. Inf. Syst. **12**, 49–59 (2003)
11. Aalst, W.M.P.: Process-aware information systems: lessons to be learned from process mining. In: Jensen, K., van der Aalst, W.M.P. (eds.) Transactions on Petri Nets and Other Models of Concurrency II. LNCS, vol. 5460, pp. 1–26. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00899-3_1
12. Luftenegger, E.: Service-dominant business design, Ph.D. thesis (2014)
13. Grefen, P., Turetken, O.: Achieving business process agility through service engineering in extended business networks. https://www.bptrends.com/
14. Hevner, A.R., March, S., Park, J., Ram, S.: Design science in information systems research. MIS Q. **28**, 75–105 (2004)
15. Peffers, K., Rothenberger, M., Tuunanen, T., Vaezi, R.: Design science research evaluation. In: Peffers, K., Rothenberger, M., Kuechler, B. (eds.) DESRIST 2012. LNCS, vol. 7286, pp. 398–410. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29863-9_29
16. Traganos, K., Grefen, P.: Hybrid service compositions: when BPM meets dynamic case management. In: Dustdar, S., Leymann, F., Villari, M. (eds.) ESOCC 2015. LNCS, vol. 9306, pp. 226–239. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24072-5_16
17. vom Brocke, J., Rosemann, M.: Handbook on Business Process Management 2: Strategic Alignment, Governance, People and Culture, 2nd edn. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-642-01982-1
18. Solaimani, S., Bouwman, H.: A framework for the alignment of business model and business processes: A generic model for trans-sector innovation. Bus. Process Manag. J. **18**, 655–679 (2012)
19. Suratno, B., Grefen, P., Turetken, O.: Towards operationalization of business models: designing service compositions for service-dominant business models. In: CEUR Workshop Proceedings (2017)
20. Kleppe, A., Warmer, J., Bast, W.: MDA Explained. Addison Wesley, Boston (2003)

21. Miller, J., Mukerji, J.: MDA Guide Version 1.0.1 (2003)
22. Fayoumi, A., Loucopoulos, P.: Conceptual modeling for the design of intelligent and emergent information systems. Expert Syst. Appl. **59**, 174–194 (2016)
23. Di Valentin, C., Burkhart, T., Vanderhaeghen, D., Werth, D., Loos, P.: Towards a framework for transforming business models into business processes. In: AMCIS 2012, p. 10 (2012)
24. Ulmer, J.-S., Belaud, J., Le Lann, J.-M.: Towards a pivotal-based approach for business process alignment. Int. J. Comput. Integr. Manuf. **24**, 1010–1021 (2011)
25. De Castro, V., Marcos, E., Vara, J.M.: Applying CIM-to-PIM model transformations for the service-oriented development of information systems. Inf. Softw. Technol. **53**, 87–105 (2011)
26. De Castro, V., Marcos, E., Wieringa, R.: Towards a service-oriented MDA-based approach to the alignment of business processes with it systems: from the business model to a web service composition model. Int. J. Coop. Inf. Syst. **18**, 225–260 (2009)
27. Object Management Group: Business Process Model and Notation (BPMN) Version 2.0 (2011). http://www.omg.org/spec/BPMN/2.0/PDF/
28. Rhazali, Y., Hadi, Y., Mouloudi, A.: A model transformation in MDA from CIM to PIM represented by web models through SoaML and IFML. In: 4th IEEE International Colloquium on Information Science and Technology (CiSt), pp. 116–121 (2016)
29. Rhazali, Y., Hadi, Y., Mouloudi, A.: A methodology for transforming CIM to PIM through UML: from business view to information system view. In: Proceedings of 2015 IEEE World Conference on Complex Systems, WCCS 2015, pp. 0–5 (2015)
30. Rhazali, Y., Hadi, Y., Mouloudi, A.: Model transformation with ATL into MDA from CIM to PIM structured through MVC. Proc. Comput. Sci. **83**, 1096–1101 (2016)
31. Rhazali, Y., Hadi, Y., Mouloudi, A.: CIM to PIM transformation in MDA: from service-oriented business models to web-based design models. Int. J. Softw. Eng. Its Appl. **10**, 125–142 (2016)
32. Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A design science research methodology for information systems research. J. Manag. Inf. Syst. **24**, 45–77 (2008)
33. von Rosing, M., White, S., Cummins, F., de Man, H.: Business Process Model and Notation —BPMN. Elsevier Inc., New York City (2015)
34. Grefen, P.: Service-dominant business engineering with BASE/X - business modeling handbook. Amazon Distribution, Leipzig (2015)

Interoperability of BPMN and MAML for Model-Driven Development of Business Apps

Christoph Rieger(✉)

ERCIS, University of Münster, Münster, Germany
christoph.rieger@uni-muenster.de

Abstract. With process models widely used as means for documentation and monitoring of business activities, the conversion into executable software often still remains a manual and time-consuming task. The MAML framework was developed to ease the creation of mobile business apps by jointly modeling process, data, and user interface perspectives in a graphical, process-oriented model for subsequent code generation. However, this domain-specific notation cannot benefit from existing process knowledge which is often encoded in BPMN models. The purpose of this paper is to analyze conceptual differences between both notations from a software development perspective and provide a solution for interoperability through a model-to-model transformation. Therefore, workflow patterns identified in previous research are used to compare both notations. A conceptual mapping of supported concepts is presented and technically implemented using a QVT-O transformation to demonstrate an automated mapping between BPMN and MAML. Consequently, it is possible to simplify the automatic generation of mobile apps by reusing processes specified in BPMN.

Keywords: BPMN · Business process modeling · Mobile app
Model transformation · Business app

1 Introduction

Business process models have been used for communicating, documenting, monitoring, and optimizing business processes for many years [19]. A wide variety of notations has emerged among which the Business Process Model and Notation 2.0 (BPMN) is best known [11]. However, process models need to be regularly synchronized with actual activities because most of them lack a sufficient level of detail to be directly executable using workflow engines. At the same time, companies are faced with global trends such as digitization and big data which require flexible business processes and new forms of organization in order to adapt to external influences.

At the intersection between process modeling and software specification, it is not possible to fully specify applications using solely BPMN models because of

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 149–166, 2018.
https://doi.org/10.1007/978-3-319-94214-8_10

their limited focus on the sequence flow of activities. Essential software development perspectives such as data models, user interfaces, and user interactions are missing. In addition, user tasks are vaguely specified and thus hard to transform into adequate representations on screen.

Especially with the increased use of model-driven approaches, domain-specific modeling notations compete against traditional general-purpose modeling notations. Ideally, changes within models propagate automatically to the derived software artifacts, thus enabling a fast reaction to changing requirements without time-consuming development cycles by IT departments. One example is the Münster App Modeling Language (MAML) which provides a graphical notation understandable both for programmers and end users to specify mobile business apps on a high level of abstraction using data-driven processes [13]. Without need for manual programming, the framework automatically generates functional cross-platform app source code for mobile devices.

MAML models integrate the full spectrum of app specification in contrast to the narrower scope of BPMN which only covers a process perspective. Nevertheless, with BPMN as the de facto standard for documenting business processes, the research question guiding this work arises: How can previous process documentation encoded in BPMN models be reused to support cross-platform app specification with MAML? An automated transformation between both notations would therefore be beneficial with regard to consistency and potential time savings: Existing process models could be enriched with app-specific information and app models representing the course of activities could be converted to BPMN for documentation and analysis.

The main contributions of this paper are three-fold and reflected by its structure: Firstly, the capabilities and shortcoming of the BPMN notation are analyzed with regard to the specification of mobile apps (Sect. 3, which also briefly introduces the MAML notation). Secondly, supported workflow patterns for BPMN and MAML are compared to highlight the conceptual differences (Sect. 4). Thirdly, transformations from BPMN to MAML and vice versa are proposed to demonstrate the practicability of the approach (Sect. 5).

2 Related Work

Several approaches which deal with the representation and transformation from process models to executable artifacts have been discussed in scientific literature, e.g., using Petri nets [21] or YAWL models [6]. Their focus lies on supporting the model interpretation by workflow engines. In the context of software development, applications are often conceptually described using layered architectures in order to distinguish the required constituents of data model, business logic, and presentation layer and, consequently, manage complexity. This can be achieved using a combination of generic notations such as UML [7] or domain-specific frameworks [1,3] but not necessarily depicting the application contents as process sequence.

For reasons detailed in the following section, the BPMN notation is not sufficient to represent all aspects of the target application. With regard to process-aware information system modeling, three categories of notations can be distinguished, each exhibiting advantages and shortcomings. Firstly, BPMN is often used to describe the process perspective together with additional notations. For example, Brambilla et al. [2] use BPMN and the WebML language focused on user interaction in order to create rich internet applications, and Traettenberg and Krogstie [19] use the refinement by Diamodl diagrams for dialog specification.

Secondly, the BPMN 2.0 notation itself can be extended to annotate further data using *ExtensionDefinitions* and *ExtensionAttributeDefinitions*. This approach is used by workflow engines such as Camunda [4] but does not allow for visualization of annotated data or new custom diagram elements.

Thirdly, custom notations have been proposed that provide integrated modeling for workflows in general or tailored to specific domains. For example, Kannengiesser et al. [9] propose an approach with a custom notation to combine UI representations with process flows through UI-related task types, and Kalnins et al. [8] combine a graphical notation for process flows with a textual expression syntax to manipulate data.

The MAML notation we use for modeling mobile business apps – i.e., form-based, data-driven apps interacting with back-end systems [10] – falls into the latter category of a domain-specific graphical notation that combines control flow logic, UI, and data perspectives such that all application concerns are specified (see Sect. 3.2). However, the transformation approach does not aim to modify the BPMN meta model but instead extracts the maximum amount of information such that the modeler only needs to enrich missing elements. In this regard, our approach is most similar to the WebRatio BPM tool [3] which transforms BPMN models to the more technical WebML notation.

3 Business Process Notations for Mobile App Development

In this section, the most commonly used notation for business process modeling, BPMN, is analyzed regarding the development of process-driven mobile applications, and the domain-specific language MAML is introduced.

3.1 Business Process Model and Notation (BPMN)

BPMN is a control flow-based notation to define a sequence of process steps. The main graphical elements can be subdivided into flow objects (activities, gateways, and events) and artifacts (data objects, groups, and annotations) which are interlinked by connecting objects (sequence flows, message flows, and associations) and organized in roles representing actors or organizations (pools and lanes) [11]. A simplified model depicting the process of writing a thesis is shown in Fig. 1 and contains several typical task types, control flow branches, and data transfer

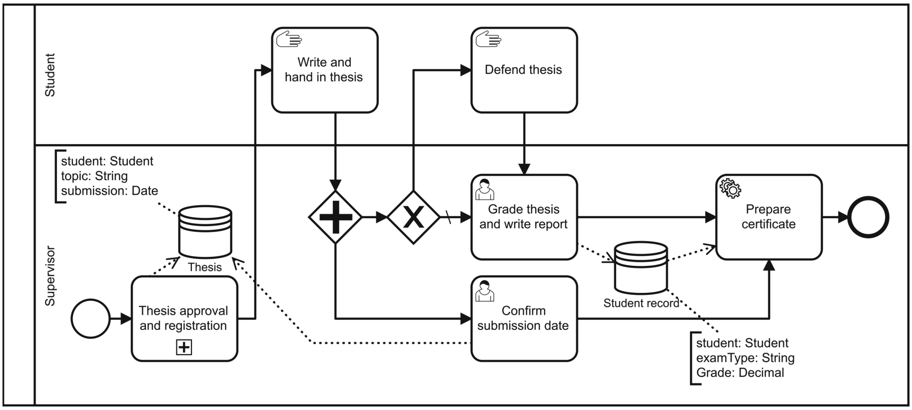


Fig. 1. Sample BPMN model representing a (simplified) process for thesis writing

mechanisms. It is further used to demonstrate how tool support for supervision-related tasks can be provided by transforming it into a mobile app.

Problems of the notation have already been studied in general [12] as well as for its applicability in software development projects [20]. The main criticism arises from deficiencies in business rule specification, symbol overload with partially superfluous or fuzzily delimited concepts, and the abundance of event types. The following paragraphs highlight the issues with regard to model-driven mobile development beyond the use of models for informally sketching process sequences.

Data Layer. BPMN is control-flow oriented and mostly neglects data modeling. Even with the enhancements of BPMN 2.0, the user can only specify the input and output of tasks using process-internal (*Data Input/Data Output/Data Object*) or persistent (*Data Store*) items and additionally distinguish between single items and collections as well as multiple states of the data. However, attributes, data types, and type interrelations – essential for the actual execution of data manipulation tasks – cannot be specified and can only be inferred on a generic level [5].

Business Logic Layer. BPMN models provide a wide range of elements to model the control-flow and conditional paths of execution. Whereas workflow engines support many such concepts, three characteristics of mobile devices complicate a direct utilization of generic BPMN elements.

Firstly, mobile apps focus on user-oriented tasks. In contrast to enterprise contexts in which many steps can be delegated to web services, apps focus on direct user interaction and require a specification of what actions to perform in a single step. Regarding data manipulation, this, e.g., includes whether an item is retrieved, updated, displayed, created, or deleted. Also, mobile-specific functionalities such as sensor/camera access and starting phone calls are not available.

Secondly, mobile devices may experience connectivity issues at any time and for unknown duration, causing unreliable response times if workflow instances are strictly allocated to one device. Similarly, multi-step transactions may lock the whole system.

Thirdly, mobile apps are usually small and inherently distributed systems. Parallel execution of tasks is hardly possible on small screens (and questionable as regards frequent context switches). Even considering a multi-role context in which activities are potentially performed by different users collaboratively, issues arise from the coordination of up-to-date distributed data and conflicting concurrent modifications.

Presentation Layer. Although default representations could be derived from the (unavailable) data attributes to display, a minimal specification of sensible user interfaces also requires the addition of informative texts and means of controlling the user interactions to navigate between views. Neither exists in BPMN.

3.2 Muenster App Modeling Language (MAML)

MAML is a graphical domain-specific language with the purpose of modeling cross-platform business apps. Following the model-driven paradigm, the framework allows for the fully automated generation of app source code for multiple smartphone platforms (currently Android and iOS) as well as a Java-based backend component for app coordination and as interface to other company systems.

The notation is built around five main design goals: (1) code-less cross-platform app creation, (2) a strong domain expert focus in contrast to complex technical notations, (3) a high level of abstraction by modeling data-driven processes, (4) task-oriented modularization using use cases, and (5) a declarative and platform-neutral description of logical steps in contrast to UI-centered configurators. Figure 2 depicts the same scenario as Fig. 1, however, it contains all the information to generate the app.

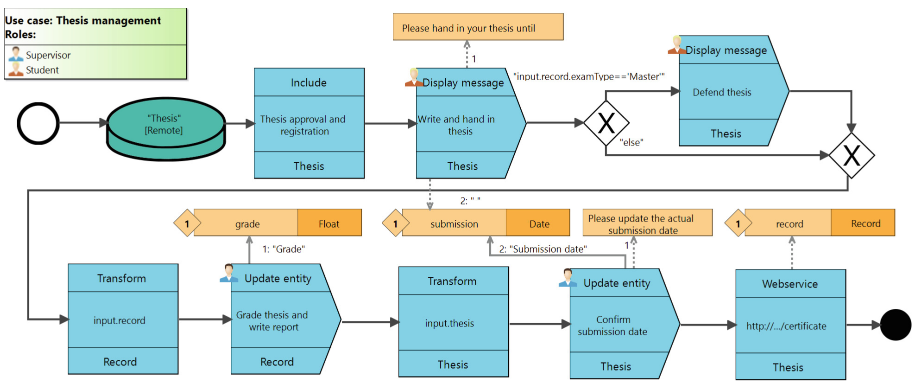


Fig. 2. Sample MAML model with thesis management process equivalent to Fig. 1

Although not necessarily limited to the domain of mobile apps, the notation considers interoperability with mobile device hardware (e.g., using elements for camera usage or sensor access), mobile interaction patterns (such as starting phone calls), and current generators target the Android and iOS platforms. Also, the integrated modeling approach to process flows and data structures may not be well suited for large-scale applications. For a more detailed introduction to MAML the reader is referred to [13,14].

4 Comparison of Workflow Patterns in MAML and BPMN

Russell et al. [17] have collected an extensive list of workflow patterns. As a comprehensive overview on workflow patterns also comprises a data access and resource perspective, respective patterns have been presented in [15,16]. To prepare a possible mapping of concepts, supported patterns need to be assessed first for both notations. An analysis of the BPMN notation is provided in [22]. Therefore, this section evaluates workflow concepts of MAML according to the pattern catalog and highlights the most important differences.

4.1 Control-Flow Patterns

Considering Workflow Control Patterns (WCP), basic patterns include the possibility to depict a *Sequence* (WCP1) of activities, a *Parallel Split* (WCP2) from a single thread of control into multiple threads executed in parallel, the *Synchronization* (WCP3) of multiple parallel activities into a single thread of control, an *Exclusive Choice* (WCP4) to execute one of multiple alternatives according to given conditions, and a *Simple Merge* (WCP5) to consolidate multiple alternative branches. In MAML, WCP1 is given by the corresponding "process connector" construct. However, as explicated in Subsect. 3.1, WCP2 and WCP3 as well as advanced branching and synchronization patterns including inter-workflow parallelism are not supported. Figure 3 depicts how branching the control flow (WCP4) in MAML can be based on a user decision (a) or evaluated based on the state of data objects (b), and can be merged accordingly (WCP5; (c)). Unlike BPMN, implicit branching and merging is disallowed to foster consistency and explicitly indicate control flow variations.

Regarding structural patterns, *Deferred Choice* (WCP16) represents a runtime choice between different branches of which the first executes. In MAML, this is possible to a certain extent: If multiple roles are assigned to one task, (only) the first user will execute the task (d). *Interleaved Parallel Routing* (WCP17) refers to a partial ordering of process step dependencies to be linearized and executed sequentially. *Critical Section* (WCP39) and *Interleaved Routing* (WCP40) are similar patterns that specify a set of subgraphs or individual activities to be executed once in arbitrary sequential order. To clarify the future app structure, MAML currently relies on an explicitly modeled, fixed sequence of activities.

Table 1. Workflow Control Patterns according to [17] in BPMN [22,23] and MAML

Workflow Control Pattern (WCP)		BPMN	MAML
1	Sequence	+	+
2	Parallel Split	+	−
3	Synchronization	+	−
4	Exclusive Choice	+	+
5	Simple Merge	+	+
6	Multi-Choice	+	+/−
7	Structured Synchronizing Merge	+	−
8	Multi-Merge	+	+
9	Structured Discriminator	+/−	−
10	Arbitrary Cycles	+	+
11	Implicit Termination	+	+
12	Multiple Instances without Synchronization	+	−
13	Multiple Instances with a Priori Design-Time Knowledge	+	−
14	Multiple Instances with a Priori Run-Time Knowledge	+	−
15	Multiple Instances without a Priori Run-Time Knowledge	−	−
16	Deferred Choice	+	+
17	Interleaved Parallel Routing	−	−
18	Milestone	−	+/−
19	Cancel Activity	+	+/−
20	Cancel Case	+	+
21	Structured Loop	+	+/−
22	Recursion	−	+
23	Transient Trigger	−	−
24	Persistent Trigger	+	+/−
25	Cancel Region	+/−	+/−
26	Cancel Multiple Instance Activity	+	−
27	Complete Multiple Instance Activity	−	−
28	Blocking Discriminator	+/−	−
29	Canceling Discriminator	+	−
30	Structured Partial Join	+/−	−
31	Blocking Partial Join	+/−	−
32	Canceling Partial Join	+/−	−
33	Generalized AND-Join	+	−
34	Static Partial Join for Multiple Instances	+/−	−
35	Canceling Partial Join for Multiple Instances	+/−	−
36	Dynamic Partial Join for Multiple Instances	−	−
37	Local Synchronizing Merge	−	−
38	General Synchronizing Merge	−	−
39	Critical Section	−	−
40	Interleaved Routing	+/−	−
41	Thread Merge	+	−
42	Thread Split	+	−
43	Explicit Termination	+	+

A *Milestone* (WCP18) represents the ability to execute an activity until a state-dependent execution point is reached. An exclusive choice with state-based condition (b) can be reused to represent this in MAML. BPMN also implements the previous patterns but does not support the concept of workflow state at all.

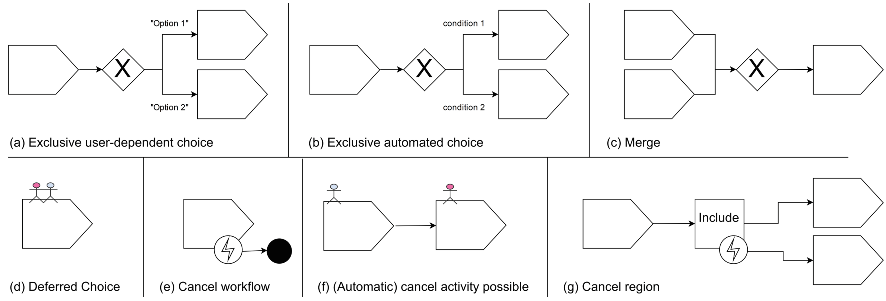


Fig. 3. Basic control-flow and cancellation patterns in MAML

Concerning cancellation patterns, MAML supports the *Cancel Case* (WCP20) pattern (e) to stop the current workflow instance. As tasks are allocated to roles – not individual users – the same process instance might also be started accidentally from multiple devices and automatically canceled for all but the first execution according to the *Cancel Activity* (WCP19) pattern (although not explicitly modeled that can occur in situations such as (f)). *Cancel Region* (WCP25) denotes a subgraph being canceled and can be represented in MAML through a subprocess terminating unsuccessfully (g).

Within a process model, *Arbitrary Cycles* (WCP10) with multiple end points can occur as depicted in Fig. 4(a). Regarding *Structured Loops* (WCP21) to execute activities repeatedly, two representations are possible: A loop is explicitly defined using an XOR element with an expression to test as condition before (for loop; (b)) or after (do-while loop; (c)) the activities are executed. In addition, users may choose multiple elements in a "select entity" step such that subsequent activities are performed for each of the selected elements. In contrast to BPMN, *Recursion* (WCP22) is possible because of passing data to subsequent process elements (d) instead of BPMN's token-based approach to sequence flows. *Persistent Triggers* (WCP24) can be used to start tasks based on a data condition (e). Finally, MAML supports both *Explicit Termination* (WCP43; (f)) and *Implicit Termination* (WCP11; (g)) of workflow instances, although the latter is discouraged to avoid incomplete process branches. Table 1 summarizes the full (+), partial (+/−), or lacking support of the workflow control patterns for BPMN and MAML.

4.2 Data Patterns

A variety of data patterns (WDP) complements the process view of a workflow (cf. Table 2). In MAML, a global perspective is adopted to define the flow of

Table 2. Workflow Data Patterns according to [16] in BPMN [22,23] and MAML

Workflow Data Pattern (WDP)		BPMN	MAML
1	Task Data	+	+
2	Block Data	+	+
3	Scope Data	+	+/−
4	Multiple Instance Data	+	+/−
5	Case Data	+	+
6	Folder Data	+	−
7	Workflow Data	+	+
8	Environment Data	+	+
9	Task to Task	+/−	+
10	Block Task to SubWorkflow Decomposition	+	+
11	SubWorkflow Decomposition to Block Task	+	+
12	To Multiple Instance Task	+	−
13	From Multiple Instance Task	+	−
14	Case to Case	+	−
15	Task to Environment - Push-Oriented	−	+
16	Environment to Task - Pull-Oriented	+	+
17	Environment to Task - Push-Oriented	−	−
18	Task to Environment - Pull-Oriented	−	−
19	Case to Environment - Push-Oriented	+	+
20	Environment to Case - Pull-Oriented	+	+
21	Environment to Case - Push-Oriented	+	−
22	Case to Environment - Pull-Oriented	−	−
23	Workflow to Environment - Push-Oriented	−	+
24	Environment to Workflow - Pull-Oriented	+	+
25	Environment to Workflow - Push-Oriented	+/−	−
26	Workflow to Environment - Pull-Oriented	+	−
27	Data Transfer by Value - Incoming	−	−
28	Data Transfer by Value - Outgoing	+/−	−
29	Data Transfer - Copy In/Copy Out	+	−
30	Data Transfer by Reference - Unlocked	+/−	+
31	Data Transfer by Reference - With Lock	+/−	−
32	Data Transformation - Input	+/−	−
33	Data Transformation - Output	+	−
34	Task Precondition - Data Existence	+/−	+/−
35	Task Precondition - Data Value	+/−	+/−
36	Task Postcondition - Data Existence	−	+/−
37	Task Postcondition - Data Value	−	+/−
38	Event-Based Task Trigger	−	−
39	Data-Based Task Trigger	−	+
40	Data-Based Routing	+/−	+

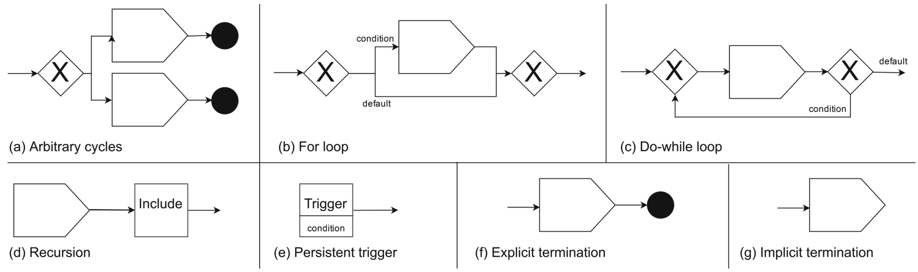


Fig. 4. Iteration, trigger, and termination patterns in MAML

data objects through the process. This represents the *Workflow Data* (WDP7) concept as depicted in Fig. 5(a). Only data required in a certain step is modeled and the state is passed to subsequent process steps, thus effectively also implementing the *Task Data* (WDP1) and *Block Data* (WDP2) patterns. *Case Data* (WDP5) representing objects specific to a process instance, and *Scope Data* (WDP3) denoting custom regions of data visibility can be achieved with local data storage which is transferred to the global data store at a later point of process execution (c). *Environment Data* (WDP 8) from the operating system can also be incorporated, for example by accessing the camera (d) or GPS location (e). Again, patterns related to concurrency are not supported, but the same task may be executed with different data objects in loops (*Multiple Instance Data*; WDP4; (f)).

This broad set of data visibility patterns reaches beyond the capabilities of BPMN which is limited to task, block, and case data through the use of parameters.

Regarding internal data interaction patterns, MAML and BPMN have similar capabilities. Data can be passed from *Task to Task* (WDP9; e.g. (a) in Fig. 5), from Block Task to *SubWorkflow Decomposition* (WDP10; e.g. (d) in Fig. 4), and *SubWorkflow Decomposition to Block Task* (WDP11) by simply connecting the respective process elements. *Data interaction To/From Multiple Instance Tasks* (WDP12–13) are unavailable for lack of concurrency. Also, *Case to Case* (WDP14) data interaction is not intended because of distributed execution on mobile devices with varying connectivity. External data interaction in MAML relies on the global data view and the user being in control of data interactions. Therefore, data can be transferred between task and environment in pull fashion using web services (WDP16; (h) in Fig. 5) as well as push-oriented (WDP15) using web services (i) or implicit/explicit remote data storage (j/k). As task data is available to the subsequent tasks of the case, this also applies to push- and pull-oriented data transfer between the environment and the case/workflow (WDP19–20, WDP23–24). All data is passed by reference (WDP30) without locking mechanisms to support low-connectivity scenarios. In contrast, BPMN supports value- and reference-based data transfer with locks as well as additional

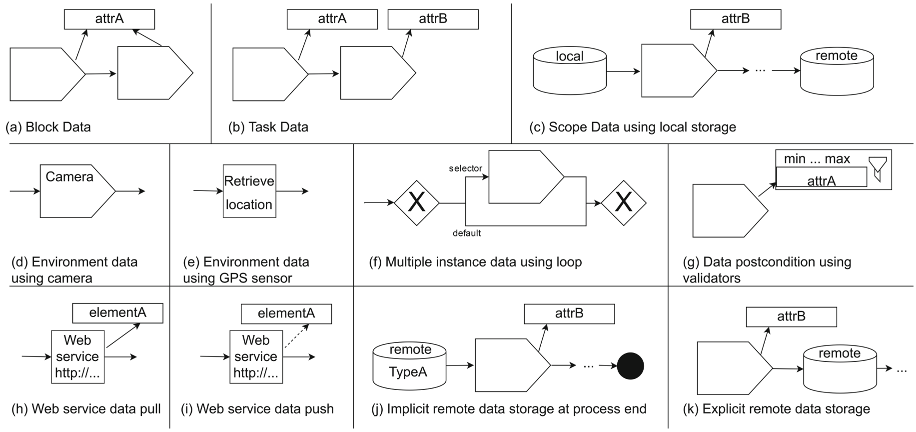


Fig. 5. Data visibility, interaction, and routing patterns in MAML

in-/output transformations, but only regarding task interactions with the environment (WDP15–18, WDP27–29, WDP32–33).

Finally, *Task Preconditions* (WDP34–35) are supported using the exclusive choice construct in MAML (cf. (b) in Fig. 3) but can flexibly validate the value of the data item in contrast to BPMN's more limited check for data existence. *Task Postconditions* (WDP36–37) can be applied equivalently or alternatively using data validators ((g) in Fig. 5). *Data-Based Task Triggers* (WDP39) (cf. (e) in Fig. 4) and *Data-Based Routing* (WDP40) possibilities (cf. (b) in Fig. 3) exist both in MAML and BPMN.

4.3 Resource Patterns

Concerning workflow resource patterns (WRP; cf. Table 3), both notations share the concept of roles to distribute tasks. A role can therefore represent a single actor (WRP1) or a group of resources (WRP2) as depicted in Fig. 6(a) and (b). The role concepts also serves as *Authorization* (WRP4) because distinct apps are created which allow only the execution of tasks related to that role.

Whereas work items in BPMN are allocated to resources (WRP14), the mobile context of MAML fosters an on-demand, pull-oriented execution of work items. Work items are *Distributed by Offer to Multiple Resources* (WRP13) and each worker can initiate the immediate execution of work items (WRP23) while having the *Selection Autonomy* (WRP26) to choose tasks from a list of open work items. To limit the amount of context switches in the distributed execution of workflows, MAML implements the *Retain Familiar* (WRP7) pattern which by default starts the next workflow step unless the current user does not comply with any of the assigned roles (c). If all tasks are assigned to the same role or only one role is defined for the use case without explicit assignment (d), this also allows for the *Case Handling* (WRP6) pattern in which all activities are performed by the same individual. Otherwise, subsequent roles are automatically

Table 3. Workflow Resource Patterns according to [15] in BPMN [22,23] and MAML

Workflow Resource Pattern (WRP)		BPMN	MAML
1	Direct Allocation	+	+
2	Role-Based Allocation	+	+
3	Deferred Allocation	–	–
4	Authorization	–	+/–
5	Separation of Duties	–	–
6	Case Handling	–	+/–
7	Retain Familiar	–	+
8	Capability Based Allocation	–	–
9	History Based Allocation	–	–
10	Organizational Allocation	–	–
11	Automatic Execution	+	+
12	Distribution by Offer - Single Resource	–	–
13	Distribution by Offer - Multiple Resources	–	+
14	Distribution by Allocation - Single Resource	+	–
15	Random Allocation	–	–
16	Round Robin Allocation	–	–
17	Shortest Queue	–	–
18	Early Distribution	–	–
19	Distribution on Enablement	+	+
20	Late Distribution	–	–
21	Resource-Initiated Allocation	–	–
22	Resource-Initiated Execution - Allocated Work Item	–	–
23	Resource-Initiated Execution - Offered Work Item	–	+
24	System Determined Work Queue Content	–	–
25	Resource-Determined Work Queue Content	–	–
26	Selection Autonomy	–	+
27	Delegation	–	–
28	Escalation	–	–
29	Deallocation	–	–
30	Stateful Reallocation	–	–
31	Stateless Reallocation	–	–
32	Suspension/Resumption	–	–
33	Skip	–	–
34	Redo	–	–
35	Pre-Do	–	–
36	Commencement on Creation	+	+
37	Commencement on Allocation	–	–
38	Piled Execution	–	–
39	Chained Execution	+	+
40	Configurable Unallocated Work Item Visibility	–	–
41	Configurable Allocated Work Item Visibility	–	–
42	Simultaneous Execution	+	+
43	Additional Resources	–	–

notified about the process instance for *Chained Execution* (WRP39). Also, complying with the *Commencement on Creation* (WRP36) pattern, new use case instances can be initiated for any process whose first task matches a user's role.

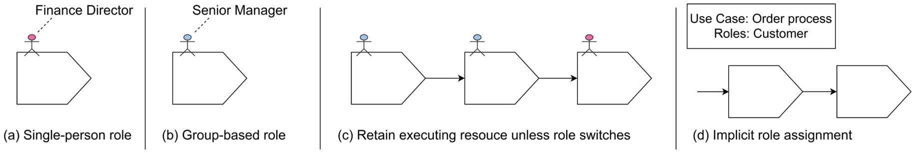


Fig. 6. Resource patterns in MAML

Automatic Execution (WRP11), i.e. trigger-based workflow execution without explicit resource allocation, is possible both in BPMN and in MAML (see (e) in Fig. 4). Also, there are no theoretical constraints on how many instances of a task are executed simultaneously (WRP42) by different users with the same role.

5 Model-to-Model Transformation

MAML models provide integrated representations for process-driven apps but need to be created from scratch. On the other hand, business processes in many organization are documented in BPMN models, although they lack the ability to fully describe mobile applications. To exploit the benefits of interoperability, a model-to-model transformation is presented based on the QVT-O language [18].

5.1 Mapping of Equivalent Language Constructs

Many core elements of the notations have direct correspondences, for instance in the thesis organization system depicted in Figs. 1 and 2. Firstly, both notations have the concepts of roles that can represent individuals or groups of resources. A MAML *use case* is represented as *pool* with a *lane* for each role in BPMN. Conversely, a process element is annotated with the respective *participant*.

Secondly, *user tasks* in BPMN correspond to *process elements* in MAML. A keyword-based matching strategy tries to classify the more specific MAML task type (e.g., *Create entity*) according to its description. Further mappings include:

- *Sub-process tasks* and *service tasks* have equivalent MAML elements named *Include* and *Webservice*
- *Manual tasks*, i.e. application-external actions, can be regarded as *Display message* steps which only output information and allow to proceed
- *Script tasks* indicate automated actions by the process engine. They need to be converted to web services as no arbitrary code can be executed.

– *Call activities* are created only for MAML's location retrieval, camera, and phone calls steps to reflect these global tasks. Other BPMN call tasks are transformed like regular sub-process tasks.

Thirdly, *start*, *end*, *timer*, and *error events* have the same semantics in both notations. Regarding data flows, this also holds true for inputs and outputs using *data associations* in BPMN and *parameter connectors* in MAML.

5.2 Mapping of Related Language Constructs

MAML applications are data-driven and therefore consider data flows together with the process flow such that each process element operates on one or multiple data items which are persisted and globally available after the process completes. Within the transformation, the respective MAML *data sources* need to be retrieved for each activity and annotated as BPMN *data store*. In addition, corresponding *InputSets* and *OutputSets* for an activity's *InputOutputSpecification* need to be created. However, nested attributes cannot be represented in BPMN and are lost during the transformation.

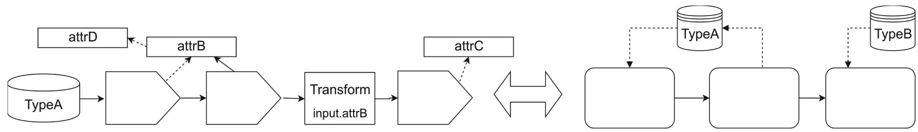


Fig. 7. Data transformation between MAML (left) and BPMN (right)

Conversely, data stores associated with BPMN tasks need to be integrated in the MAML process flow. In case the data item to process differs between tasks, additional *Transform* elements are needed to establish the link (see Fig. 7).

Without direct support for concurrency, several workflow patterns can be rewritten to concepts available in MAML. *Parallel gateways* and (equivalent) *implicit parallel splits* are for example transformed to a sequence of activities as depicted in Fig. 8(a). With exclusive gateways to branch process flows, inclusive gateways are mapped to a series of optional steps (b). Similarly, loops (WRP21; (b/c) in Fig. 4) and milestones can be transformed using automatically evaluated, state-based decisions.

5.3 Unmapped Language Constructs

BPMN contains several elements without equivalent representation in MAML. Because unstable mobile connectivity precludes concurrency, BPMN's *conversations* and *choreographies* are unsupported in MAML. In addition, *multiple instance tasks*, *(parallel) multiple events*, and *message flows* are not available in MAML and cannot be mapped to equivalent concepts because of the inherently sequential design of smartphone apps. Similarly, protocols for BPMN's

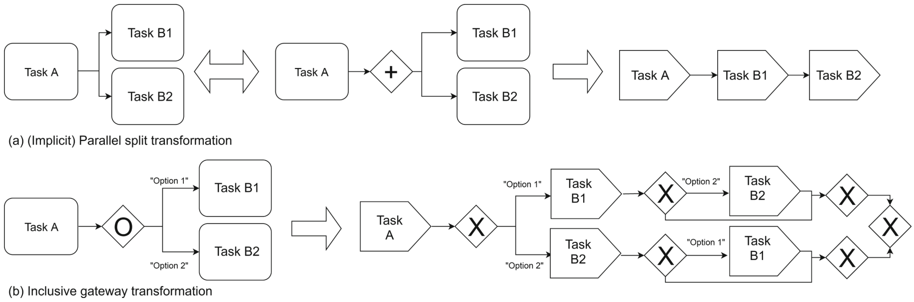


Fig. 8. Gateway transformation from BPMN to MAML

transactional tasks, *compensation*, and *cancel events* contradict the distributed characteristics of mobile systems.

At the current state of development, MAML also does not contain complex control flows for two reasons. Firstly, the model-driven approach aims to actually create executable source code. Features equivalent to BPMN's *signal*, *escalation*, or *termination events* and respective *event-based gateways* need to have intuitive user interactions in the app. Secondly, some elements in BPMN such as *complex gateways*, *business rule tasks*, and *compensation* are less structured or delegate processing to a workflow engine. However, each element requires an adequate in-app representation and a sufficient level of detail in the model to allow for an automated transformation.

On the other hand, some elements in MAML have no adequate representation in BPMN. Most important, custom data types and data objects with nested attribute structures reach far beyond BPMN's capability of specifying primitive key-value pairs as properties of activities. For the same reason, data-driven features such as *computed attributes* and data-dependent variations unknown at design time (*SingleResultEvent*/*MultiResultEvent*) have no equivalent representation in BPMN. Also, interface-related information such as labels, buttons, and field captions is not in the scope of BPMN.

6 Discussion

Section 4 explained the capabilities of MAML and contrasts them to the established BPMN notation. The transformation result of the BPMN model in Fig. 1 is displayed in Fig. 2 (enriched with label texts). As can be seen, many concepts are shared by both notations, although MAML does not strive for being a feature-complete workflow engine. Especially the data layer is currently not well represented in BPMN and has no direct interrelation with data modeling languages such as UML. Workflow engines such as Camunda bypass some of the limitations with custom extension attributes, but data-centric actions still need to be extracted into programmed web services. In addition, the granularity of modeled processes varies strongly and even with a detailed representation, the modeler might often apply mobile-specific adaptations to the representation and enrich the model with UI elements. We therefore refrain from extending BPMN for our needs. Instead, our transformation eases the import of potentially already existing process documentation and mobile app design. Conversely, small enterprises which have only process documentation on a higher level of abstraction can reuse models used for app generation and export a detailed BPMN representation. As the model-driven paradigm focuses on this primary artifact, the process documentation always remains synchronized with the actual implementation, both regarding process flows and data models which can be derived together with access restrictions directly from the MAML representation (cf. [14] for more details on the data model inference). Particularly for small and medium enterprises, MAML can therefore be seen as a tool supporting digitized business models and increasing quality of operations through custom business apps – eventually even designed by end users within the company. A transformation from and to BPMN thus unburdens departments by simplifying the specification of software and documentation of activities.

Limitations of our work include both improvements to the transformation between BPMN snd MAML as well as its practical application. On the one hand, the evolution of the domain-specific MAML notation may introduce workflow patterns already found in BPMN. Also, previously unmapped language constructs with no adequate representation might be substituted by app-related concepts in future (e.g., establishing conventions for utilizing event-based gateways in the context of notifications). On the other hand, a real-world use in companies with different engagement in process documentation is required to further validate the applicability of the approach and constitutes future work. By analyzing existing models of different granularities, recurring patterns may be uncovered that can also be used for improving the transformation.

7 Conclusion and Outlook

In this paper, the interoperability between the established process modeling notation BPMN and the domain-specific graphical modeling language MAML for mobile-app creation have been assessed. Therefore, MAML was analyzed based on 43 control flow, 40 data, and 43 resource patterns identified for workflows by Russell et al. [15–17]. Although many correspondences exist with regard to control flow, data, and resource handling, differences relating to concurrency, complex control flows, and transactional behavior remain due to different application domains of BPMN using synchronous workflow engines and MAML in distributed mobile environments. Nonetheless, a mapping between many concepts can be found and an implemented model-to-model transformation using QVT-O demonstrates the transferability of concepts. From MAML to BPMN, fully specified models are created. In the opposite direction, additional data- and UI-related elements need to be added due to their absence in the specification of BPMN. Thus, the development process of mobile business apps using MAML can indeed be supported by reusing process knowledge documented in BPMN models. Regarding future work, a real-world introduction of the approach is pending in order to validate the transformation rules, especially considering the detailedness of input models encountered in practice.

References

1. Barnett, S., Avazpour, I., Vasa, R., Grundy, J.: A multi-view framework for generating mobile apps. In: IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), pp. 305–306 (2015). https://doi.org/10.1109/VLHCC.2015.7357239
2. Brambilla, M., Preciado, J.C., Linaje, M., Sanchez-Figueroa, F.: Business process-based conceptual design of rich internet applications. In: ICWE, pp. 155–161 (2008). https://doi.org/10.1109/ICWE.2008.22
3. Brambilla, M., Butti, S., Fraternali, P.: WebRatio BPM: a tool for designing and deploying business processes on the web. In: Benatallah, B., Casati, F., Kappel, G., Rossi, G. (eds.) ICWE 2010. LNCS, vol. 6189, pp. 415–429. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13911-6_28
4. Camunda Services GmbH (2017). BPMN workflow engine. https://camunda.org/
5. Cruz, E.F., Machado, R.J., Santos, M.Y.: From business process modeling to data model: a systematic approach. In: QUATIC, pp. 205–210 (2012). https://doi.org/10.1109/QUATIC.2012.31
6. Decker, G., Dijkman, R., Dumas, M., García-Bañuelos, L.: Transforming BPMN diagrams into YAWL nets. In: Dumas, M., Reichert, M., Shan, M.-C. (eds.) BPM 2008. LNCS, vol. 5240, pp. 386–389. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-85758-7_30
7. de Giacomo, G., Oriol, X., Estañol, M., Teniente, E.: Linking data and BPMN processes to achieve executable models. In: Dubois, E., Pohl, K. (eds.) CAiSE 2017. LNCS, vol. 10253, pp. 612–628. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59536-8_38
8. Kalnins, A., Lace, L., Kalnina, E., Sostaks, A.: DSL based platform for business process management. In: Geffert, V., Preneel, B., Rovan, B., Štuller, J., Tjoa, A.M. (eds.) SOFSEM 2014. LNCS, vol. 8327, pp. 351–362. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04298-5_31
9. Kannengiesser, U., Heininger, R., Gründer, T., Schedl, S.: Modelling the process of process execution: a process model-driven approach to customising user interfaces for business process support systems. In: Schmidt, R., Guédria, W., Bider, I., Guerreiro, S. (eds.) BPMDS/EMMSAD -2016. LNBIP, vol. 248, pp. 34–48. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-39429-9_3
10. Majchrzak, T.A., Ernsting, J., Kuchen, H.: Achieving business practicability of model-driven cross-platform apps. OJIS **2**(2), 3–14 (2015). https://doi.org/10.19210/OJIS_2015v2i2n02_Majchrzak
11. Object Management Group (2011). BPMN: http://www.omg.org/spec/BPMN/2.0
12. Recker, J.: Opportunities and constraints: the current struggle with BPMN. Bus. Process Manag. J. **16**(1), 181–201 (2010). https://doi.org/10.1108/14637151011018001
13. Rieger, C.: Evaluating a graphical model-driven approach to codeless business app development. In: HICSS, pp. 5725–5734 (2018)

14. Rieger, C., Kuchen, H.: A process-oriented modeling approach for graphical development of mobile business apps. COMLAN **53**, 43–58 (2018). https://doi.org/10.1016/j.cl.2018.01.001
15. Russell, N., ter Hofstede, A., van der Aalst, W.: Workflow resource patterns. BETA Working Paper Series, WP 127 (2004)
16. Russell, N., ter Hofstede, A.H.M., Edmond, D., van der Aalst, W.M.P.: Workflow data patterns: identification, representation and tool support. In: Delcambre, L., Kop, C., Mayr, H.C., Mylopoulos, J., Pastor, O. (eds.) ER 2005. LNCS, vol. 3716, pp. 353–368. Springer, Heidelberg (2005). https://doi.org/10.1007/11568322_23
17. Russell, N., ter Hofstede, A., van der Aalst, W., Mulyar, N.: Workflow control-flow patterns: a revised view. BPM Center Report BPM-06-22 (2006)
18. The Eclipse Foundation (2016). QVT Operational: http://www.eclipse.org/mmt/qvto
19. Trætteberg, H., Krogstie, J.: Enhancing the usability of BPM-solutions by combining process and user-interface modelling. In: Stirna, J., Persson, A. (eds.) PoEM 2008. LNBIP, vol. 15, pp. 86–97. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89218-2_7
20. Tuomainen, M., Mykkänen, J., Luostarinen, H., Pöyhölä, A., Paakkanen, E.: Model-centric approaches for the development of health information systems. In: Studies in Health Technology and Informatics, vol. 129 (2007)
21. van der Aalst, W.M.P.: Putting high-level Petri nets to work in industry. Comput. Ind. **25**(1), 45–54 (1994). https://doi.org/10.1016/0166-3615(94)90031-0
22. Wohed, P., van der Aalst, W.M.P., Dumas, M., ter Hofstede, A.H.M., Russell, N.: On the suitability of BPMN for business process modelling. In: Dustdar, S., Fiadeiro, J.L., Sheth, A.P. (eds.) BPM 2006. LNCS, vol. 4102, pp. 161–176. Springer, Heidelberg (2006). https://doi.org/10.1007/11841760_12
23. Workflow Patterns Initiative (2017). http://www.workflowpatterns.com/

An Information Security Architecture for Smart Cities

A. R. R. Berkel[1], P. M. Singh[2], and M. J. van Sinderen[1(✉)]

[1] Faculty of EEMCS, University of Twente, Enschede, The Netherlands
a.r.r.berkel@student.utwente.nl,
m.j.vansinderen@utwente.nl
[2] OPAC Research Group, TU/e, Eindhoven, The Netherlands
p.m.singh@tue.nl

Abstract. The growing use of ICT in public life has coerced the concept of smart cities. In a smart city, numerous physical devices coupled with latest ICT technologies are used by city authorities to provide better services and infrastructure to its citizens. Smart cities have unique security challenges. Owing to the involvement of numerous stakeholders, information security breaches can have wide ranging, long-lasting consequences. The challenges of smart city information security have not received the attention they deserve. This paper combines threat analysis and enterprise architecture modelling to address and mitigate these challenges from a holistic perspective. An information security architecture is presented, which can help stakeholders of the smart city projects to build more secure smart cities.

Keywords: Smart cities · Information security · Enterprise architecture Archimate

1 Introduction

As cities around the world (e.g., Barcelona, Amsterdam, Santa Cruz etc.) increasingly use ICT for better administration, the concept of smart city has emerged. It refers to the co-ordinated use of physical devices, ICT, software systems and analytics by city authorities to provide better services and infrastructure to citizens. Smart cities have unique security challenges. Primary among those are safeguarding the security and privacy of data. Physical devices, software systems and analytics in a smart city use enormous amounts of data collected from the public domain. They also generate huge amounts of (potentially sensitive) information useful to government agencies and city authorities. Smart city security has raised much research interest, but so far results mostly focus on separate levels of the data chain or specific components of the smart city infrastructure.

This paper combines threat analysis and enterprise architecture modelling to address and mitigate smart city information security challenges from a holistic perspective. We performed a systematic literature review [22, 28] to acquire background knowledge on (1) smart city architectures and projects, (2) security issues and threats in smart city infrastructures, and (3) security controls that have been implemented to treat

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 167–184, 2018.
https://doi.org/10.1007/978-3-319-94214-8_11

such issues and threats. The need for a holistic perception of a smart city project inspired us to choose enterprise architecture modelling. Although considered in a broad (including physical and social) context, we aim at security controls at the technology and software application levels. To come up with a smart city information security architecture, we first developed a base-line enterprise architecture that incorporates the knowledge from the first part of our literature review. We then analysed the second and third part of our literature review and developed a target enterprise architecture using a meta-model for security from the literature [3]. To investigate and illustrate the completeness of the target architecture, we modelled an existing smart city project [29]. The author believe that this approach will aid in elevating the discussion of smart city security beyond the level of system developers and ICT experts to the level of administrators, managers and policy makers.

The rest of the paper is structured as follows: Sect. 2 provides the background on smart cities concept and existing smart city architectures. Section 3 discusses the baseline architecture for smart cities. Section 4 analyses relevant security threats and presents an information security architecture for smart cities. Section 5 illustrates our approach with a smart city case study. Finally, Sect. 6 concludes the paper with possible future works.

2 Background

2.1 Smart City Architectures

Many definitions for a smart city were proposed [2, 5, 10, 42]. Following one of the most popular papers, the following definition by Caragliu et al. [9] is used: *a city is smart when investments in human and social capital and traditional (transport) and modern (ICT) communication infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory governance.*

The fact that there are many definitions allow for many frameworks constituting what a city is. Many frameworks attempt to define a smart city by describing its critical factors or key elements [5, 10, 19]. Frameworks overlap even though they sometimes have completely different focuses. One framework that tries to make sense of all the different perspectives on smart cities is the framework of Nam and Pardo [32]. They state that a smart city is more of an umbrella term for many different kinds of cities that are all trying to overcome the urbanization challenges and improve the quality of life by rapidly improving some specific aspect of the city. They conceptualise a smart city along three dimensions: technology, human and institution. Within all of these dimensions, they name a number of conceptual cities that will fall in that dimension. The technology dimension for instance is the main focus of digital cities, intelligent cities, ubiquitous cities, wired cities, hybrid cities and information cities. They state that when all three dimensions are represented, a city can be called smart. It could thus be that many of the frameworks stated in the literature talk about the smart city while actually looking at a very closely related concept. Despite this promising framework, the lack of formal definition is still a problem and this results in many different architectures being proposed. In order to build a security architecture for sensitive data in smart city technology, first an analysis of these proposed architectures in the literature has to be made. This analysis will lead to a base-line architecture upon which the security extension can be build. Six architectures were analysed on their suitability for inspiration for a base-architecture. This suitability is mostly determined on the basis of two characteristics:

- Is the architecture developed using any language with a meta-model? More specifically, was a particular language used to create the architecture by making use of concepts and relationships. This will make it more meaningful and readable.
- Is the architecture focused on data/technology?

Using these criteria, the following architectures were found.

Jin et al. [23] describe an information framework that would enable the development of a smart city through the use of the Internet of Things (IoT). While providing a solid overview of the implementation possibilities of IoT for the application of smart cities, they also provide an IoT infrastructure that from the perspective of two domains: the communication stack and the data flow. Filipponi et al. [18] provide a high-level event driven architecture that discusses the use of Semantic Information Brokers (SIBs) that provide service interfaces known as Knowledge Processors (KP) that can be used by numerous agents. The architecture uses a more structured language, but still not an accurately defined one as the concepts and relationships are not defined. Gaur et al. [19] again provide an architecture from the perspective of IoT. Given the increasing amounts of data, there is a need for new methods and techniques for effective data management and analysis so that the resources within a city can be managed in an intelligent and dynamic way. Wenge et al. [42] attempted at synthesising a number of architectures and propose a new architecture from the perspective of the data that underpins all the functionality of the smart city. They propose the architecture and then analyse whether the proposed concepts were also used in other architectures (including some in this analysis). Anthopoulos and Fitsilis [2] propose a general architecture for urban development aimed at helping a city transform into a smart city (or as they call it themselves, a ubiquitous city). The architecture consists of five layers: the stakeholder layer (consisting of end users of the services), the service layer (where the services are provided), the business layer (where rules and policies are defined that allow the smart city to understand how to operate), the infrastructure layer (includes the basic network) and the information layer (where data is generated and stored). The architecture provides an overview of all different aspects involved with the smart city. Jalali et al. [21] propose an architecture for community level services through IoT. They show an architecture which enables citizens to make use of real time data that has been gathered using various sensory mechanisms in order to analyse and make decisions for future planning.

These findings will be used for a commonality analysis in Sect. 4.

2.2 Archimate

Owing to our research goal of a holistic smart city architecture model, we have chosen enterprise architecture modeling. An enterprise architecture is able to combine multiple viewpoints e.g. data, application, business, value, governance, security etc. in one architectural diagram. We argue that a smart city can be treated as an enterprise. It is an interconnected system of organisations which collaborate among themselves, the same way as departments in an enterprise [27]. Numerous enterprise architecture modelling languages can be found in literature. In this research, we have chosen Archimate 3.0 enterprise architecture modeling language [39]. The reasons for our choice are as follows. Firstly, Archimate 3.0 is an Object Management Group (O.M.G) standard and is extensively used in industry. Secondly, Archimate 3.0 has a security and risk extension which aligns well with our research goal in this paper. The security and risk metamodel used in this research is a subset of the original metamodel as presented in literature and is depicted in Fig. 1. ArchiMate 3.0 consists of four layers [39]: The business layer consists of products and services which are realised by business processes and functions. The application layer supports the business layer with application services, realised by software application components. The technology layer provides the required infrastructural services & devices needed to run software applications. The physical layer models physical assets of an enterprise.

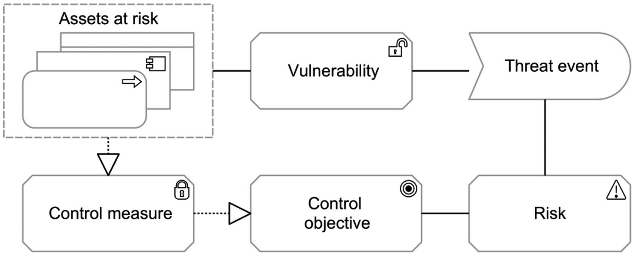


Fig. 1. Security and risk metamodel based on the Archimate 3.0 specification [3]

To design a smart city baseline enterprise architecture the following 4 design patterns will be used. These design patterns are derived from Archimate 3.0 concepts mentioned in [40].

a. *Service connection between layers.* The most basic pattern in the architecture consists of the main concepts within each layer as well as the connections between these layers. The main concept within each layer is the ‘function’ concept. A function realizes a ‘service’. These services represent the value of the function and connects to the function in the next layer or to another service.
b. *Data representation in different layers.* The next pattern concerns the flow of data. In the technology layer, data is stored in a database or application server and represented as an ‘artefact’. In the application layer, data is represented as a ‘data object’. This data object can either be realized by the artefact alone (e.g. raw data) or realized by the artefact plus an application function (with an access relationship).
c. *Service commitments with third parties.* Modelling service commitments in the way as depicted in Fig. 2 ensures better visibility of security threats. To model this connection, the ‘object’ concept is used and resembles a contract [33]. It is linked to the customer and service provider using an ‘association’ relationship. Furthermore, the contract is linked with the product using an ‘aggregation’ relationship.

d. *Application component aggregation.* The application layer can consist of multiple application components that together deliver an application function. However, components that together form a function are less relevant as the security aspects are not specific for a component. Therefore, the different components that comprise a function are not explicitly modelled but aggregated into one general component.

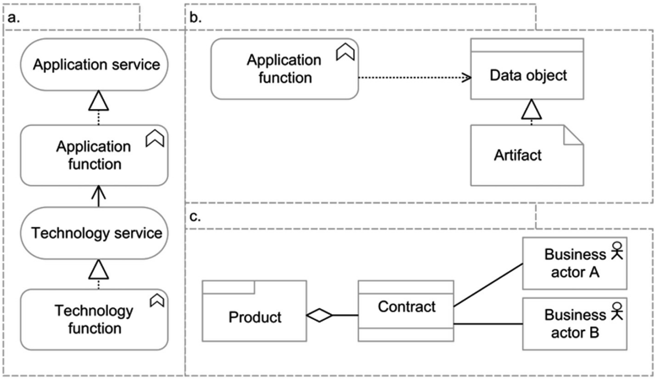


Fig. 2. The Archimate patterns of a, b, and c used in the architecture

Using these design patterns, we created a smart city base-line architecture, presented in the next Section. Using the security and risk metamodel, the baseline architecture will be extended. The advantage of building the security and risk architecture as an extension is that it can be easily added to an already existing architecture.

3 A Base-Line Architecture for Smart Cities

This section presents a base-line architecture for smart cities. It was created after studying the existing smart city architectures (Sect. 2.1) and performing a commonality analysis. Using a adapted version of the scope, commonality and variability (CSV) analysis, some commonalities among the architectures were identified [11], which are as follows. Many of the architectures model a smart city in layers that starts with data generation, which is being used by some application and ends with a service or processed data that serves an end-user (commonality a) [2, 19, 21, 23, 42]. Also, many architectures model a data flow where sensors and devices acquire data and a distributed network is used for transmission (commonality b) [2, 18, 19, 21, 23, 42]. This allows the creation of a layered, generic and holistic base-line architecture for smart cities (see Fig. 3). The layers of the baseline architecture are as follows:

Physical Layer. This layer consists of one component: sensors. It models every sensing device available within a smart city (e.g. RFID tags, weather sensors, smart card readers etc.). This connects to commonality b [18, 19, 21, 23].

Technology Layer. The technology layer consists of concepts needed to store, handle data and host applications [3, 21, 42]. Data and application servers can either be in the cloud or locally (for scalability reasons the cloud is recommended) [18, 21, 23]. If the cloud environment is chosen, the service will belong to a product and thus requires a contract between the service provider and the government. By studying the different types of data in existing smart city architectures, the database in a smart city, consists of four types of data: raw data (directly from the sensors), processed data, analysed data [19, 21, 23] and third-party data (for enrichment of analytics or smart services) [2] (commonality b). Both server types are modelled into one concept as security will not depend on the function of the server, rather on its technology.

Application Layer. The application layer consists of two types of applications. *(a)* Applications that process data [19, 21, 23]. This ensures that the database has data that can later be used by other applications. Data processing might include filtering, cleaning and transforming data. *(b)* Data analysis applications [19, 21, 23]. These applications use the processed data and transform it into useful information. Both application types are modelled into one concept as security will not depend on the function of the application, rather on its technology.

Business Layer. The business layer consists of high-level processes. It is entirely up to the implementation of the smart city to specify all processes necessary. Still, at a higher level of abstraction the generic service of delivering a smart service can be identified [18, 19, 21]. The way that it is delivered depends on (among others) the function of the service. Also, the underlying applications that are needed to generate the data for the service, will differ in complexity depending on the service.

4 An Information Security Architecture for Smart Cities

This section presents a detailed threat analysis of smart cities. Following a structured approach discussed in Sect. 4.1, requirements are listed in Sect. 4.2. Then, potential risks are listed in Sect. 4.3. For every risk, corresponding defence mechanisms and responses are specified in Sect. 4.4. The requirements, threats and responses are linked to architectural concepts so that they can be further used for modeling information security architecture for smart cities.

4.1 Approach

Data in a smart city includes individual, organisational and public data. Providing the right security for sensitive information when people use smart services is a major requirement [42]. In order to provide this security, firstly, possible threats need to be analysed.

Information security describes *the process and methodologies which are designed and implemented to protect print, electronic, or any other form of confidential, private, and sensitive information or data from unauthorized access, use, misuse, disclosure, destruction, modification or disruption* [37]. Band et al. [3] conclude that this type of security is part of the overall security function within Enterprise Risk Management.

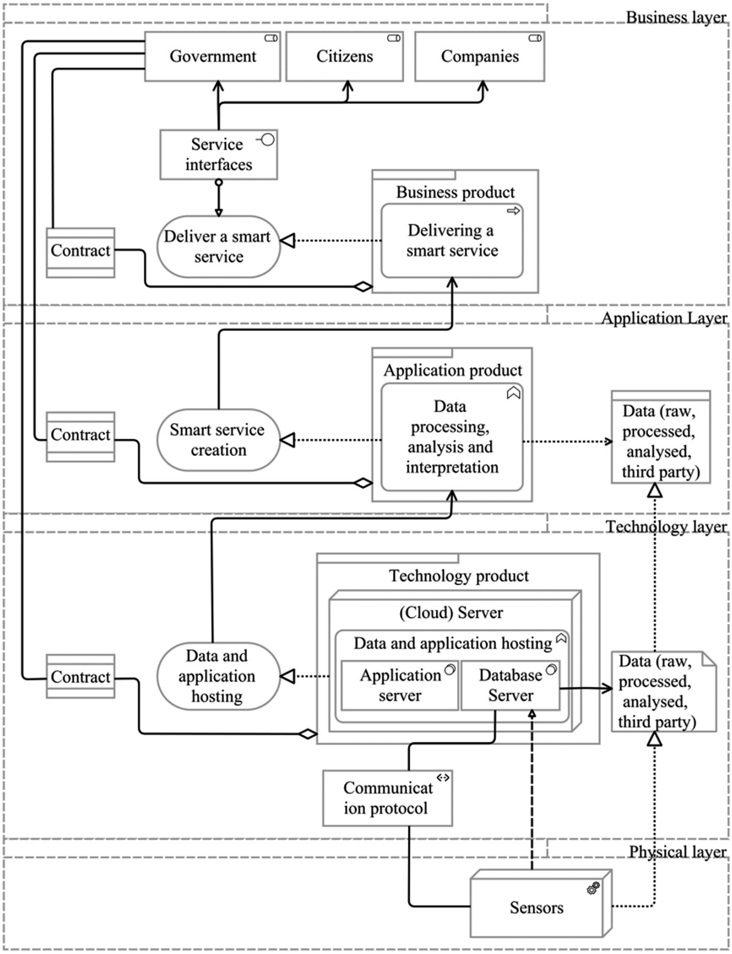


Fig. 3. A base-line architecture for smart cities

This definition is closely related to the classic confidentiality, integrity, and availability (CIA) model that is often used for developing security policies. We retrieved security issues from three research domains: information security, IoT security and Smart City security. In order to develop a successful security policy, six types of activities are proposed that will be used in this research [43]:

- Requirements setting. Managers in an enterprise must have a process in place to set objectives consistent with both the mission of the enterprise and its appetite for risk.
- Event and risk identification. Enterprises must identify both internal and external events and risks that could lead to security breaches.
- Risk Assessment. Every identified risk has to assessed separately, along with any successive effects. In this research, risk assessment will not be performed due to the fact that every city needs to assess their risks individually. Impact and likelihood of risks are case dependent and highly subjective.

- Risk Response. Enterprises must determine how to respond to risks.
- Controls, Information and Communication. The communication and documentation of risks within enterprises and the determination of procedures and policies for effective responses is beyond the scope of this research and is thus not discussed any further.

4.2 Requirements Setting

In order to specify requirements for security policy, the relevant areas for security have to be analyzed. Numerous security mechanisms might be required within smart cities [42]. In this subsection we first enumerate relevant areas of security which are then linked to the Archimate layers that can be used to model them.

- Sensor network security. Since most of the data in a smart city is generated by sensors, the security of data acquisition implies protection for sensor content. Many different disciplines come into play here: signal processing, hardware design, privacy rights etc. Techniques from a combination of these disciplines might be needed to ensure the reliability and security of data according to the CIA-triad. Since the most common threat for smart cities is from wireless sensor devices [35], therefore special attention needs to be paid to sensor network security. Sensing techniques use physical devices to acquire data, thus they represent the physical layer in Archimate.
- Transmission security. After data is acquired, sending it over a network proves to be another area which requires attention, as the network in itself has to be secure. The security here has to prevent network attacks and intrusion. Since transmission of data happens between layers of Archimate, we model this security at the junction of these layers: i.e. the service concept.
- Support layer security. This layer should prevent stored data from being revealed, damaged or tampered by anyone without the right authorization. This aligns with the technology layer of Archimate.
- Application security. This area is mainly focused on securing specific applications, platforms and interfaces. This aligns with the application and business layer of Archimate as this is where the actual applications, services, platform and interfaces are modelled.

There are a number of security requirements that need to be monitored in all the layers [14, 15, 20], which are as follows:

- Data confidentiality, integrity and availability according to the well-known CIA-triad.
- Access control. Information providers must be able to implement access control on the data provided.
- Client privacy. Measures need to be put in place so that only the information provider is able make inferences from observing the use of the lookup system, related to a specific customer. Also, when using sensors to acquire data, measures need to be put in place so that people are aware of the fact that it is likely that their data is being monitored and analyzed.

- User identification/identity management. The process of validating users before allowing them to use the system.
- Secure data communication. Authenticating communicating peers, ensuring confidentiality and integrity of communicated data, preventing repudiation of a communication transaction, and protecting the identity of communicating entities.
- Authenticity. This is a way to ensure that data, systems, communication processes or information are genuine. Authenticity is also responsible for validating that both parties involved in a communication process are who they claim to be.
- Resilience to attacks. Avoiding single points of failure and ensuring that physical or logical probing of the system is not possible even if it falls in the wrong hands.

4.3 Event and Risk Identification

Some generic threat events [4] relevant to our research context are listed as follows:

- Deliberate attacks from disgruntled employees, industrial espionage and terrorists that leads to attackers being able to penetrate a network, gain access to control software, and alter load conditions to destabilize the system in unpredictable ways.
- Inadvertent compromises of the information infrastructure due to user errors.
- Equipment failure.
- Natural disasters.

Furthermore, based on our requirement analysis, we present five most important risks in smart cities:

- Interception of messages. Smart cities should prevent, at all costs, the interception of messages. If not, smart cities run the risk of data and application misuse and lack of trust in the service. The risk of interception of messages emerges from the requirements of secure data communication, client privacy, and authenticity. It applies to transmission layer security.
  Sensing messages are a special kind of messages as they emerge directly from sensing devices in the environment of the city and thus need different responses to secure them. The risk of interception of sensing messages emerges from the requirement of resilience to attacks, authenticity, client privacy and secure data communication. It applies to sensor network security.
- Authentication. Smart cities must prevent unauthorized access or communication of data within their cities. If not, smart cities run the risk of wrongful access to data, application misuse and lack of trust in the service. Authentication related risks emerge from the requirements of access control, client privacy, user identification/identity management and authenticity. It applies to transmission layer security and support layer security.
- Hacking. Smart cities should prevent physical and logical hacking attempts on their systems. The risks related to hacking emerge from the requirements of resilience to attacks and client privacy. It applies to support layer and application layer security.
- Viruses. Smart cities should keep their systems virus free. Virus related risks emerge from the requirements of secure, data communication and authenticity. It applies to support layer and application layer security.

- Availability. Smart cities should implement responses to ensure high availability of services. Availability related risks emerge from the requirement of data availability. It applies to support layer and application layer security.

4.4 Risk Responses

With the risks listed in Sect. 4.2 above, this subsection proposes risk responses that can help in mitigating them. Many different ways of implementing security are available and different smart city projects might chose different implementations [32]. Nonetheless, our following analysis mentions various state-of-the-art mechanisms that we think are most relevant for smart cities. In this section, for each of the risks, appropriate responses are discussed. Then, the risks and responses, together with their linkage to smart city components are translated into an architectural pattern and presented individually. Together with the base-line architecture (Fig. 3), they represent the full information security architecture for smart cities.

Interception of Messages. There are multiple points within the system where this might happen as data transmission happens in the entire system. To mitigate the risk of messages being intercepted, the following response techniques can be implemented.

Encryption. In the traditional transmission layer where data is transported over a network, a by-hop encryption mechanism is used. This is because in this transmission process, the data needs to remain in plain text in each of the nodes of the system through decryption and encryption. However, in the traditional application layer, this is not the case. Here, only the sender and receiver can decrypt the data and each passing node only handles encrypted data. This is called end-to-end encryption [38]. This type of encryption provides the highest level of security.

Key Agreement and Authentication. These techniques ensure that no third party can interfere with the transmission of data over a distributed network [7]. Many algorithms are available to achieve this. E.g. Diffie-Hellman protocol [16], Rivers Shamir Adleman protocol [36].

Communication Protocols. They provide integrity, authenticity and confidentiality. In a transmission layer, one might look at the TLS/SLL and IPSec protocol that is designed for this purpose [38]. However, since many applications in a smart city could be implemented through an IoT platform, internet protocols are most appropriate.

Network Intrusion Detection. It can help in identifying (preferably in real time), unauthorized use, misuse and abuse of computer systems by both internal as well as external actors [30]. This response technique ensures data confidentiality and integrity.

Figure 4 shows the responses related to message interception risks as well as the concepts of the base-line architecture to which they relate, in an architectural model. The risk and responses are linked to the services that each layer presents to the layer above it. It is located here because these services form the bridge between the different layers. Therefore, they represent the transmission of data between those layers.

Interception of Sensing Messages. As mentioned in Sect. 4.3, messages from sensing devices might require different or extra responses to secure them. The communication

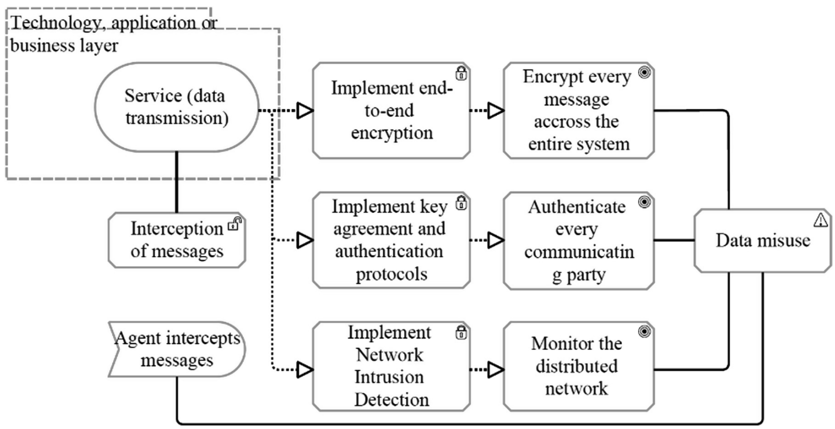


Fig. 4. Security architecture for the vulnerability of message interception

protocol between the sensors and the distributed network is the most likely point of attack. To solve the privacy and authentication issue here, symmetric-key cryptographic primitives can be used that ensure that no unauthorized reader will have access to an RFID node [25]. Unfortunately, this method is developed for only RFID and thus other techniques are also needed. Other relevant security mechanisms to use are Identity-Based Encryption from the Weil Pairing [8], Quantum Cryptography [6] or the Key-Management scheme [17].

Figure 5 shows the responses for risk related to interception of sensing messages.

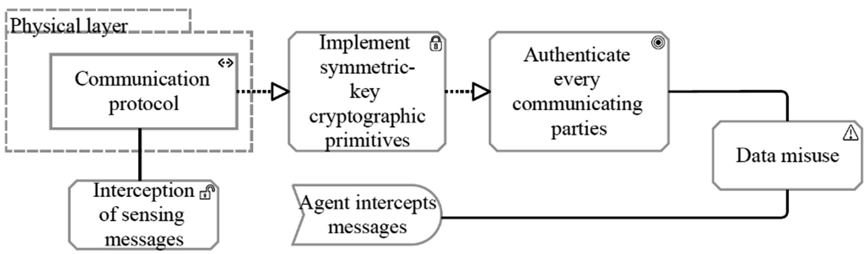


Fig. 5. Security architecture for the vulnerability of sensing message interception

Authentication. The most likely point of attack for this risk is the interface of the system as this is where authentication is done. To ensure application layer authentication, users may be asked to implement two-step verification. In this technique, users are asked to first provide a key that only the user would know (a password or pin) plus something that they can carry (such as a token) [31]. Threshold and Identity-based Key Management can be used to improve authentication specifically in an environment where networks are wireless and ad hoc [13]. Also, passwords and log in details of admin/users can be lost which can lead to unwarranted usage of the system. Therefore, passwords/keys of users should be regularly updated and reissued.

Figure 6 shows the responses for this risk related to authentication.

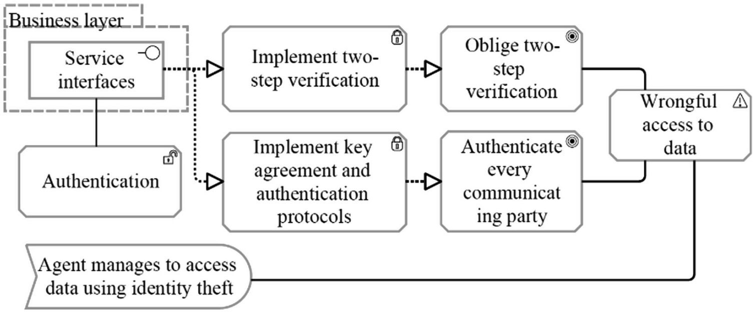


Fig. 6. Security architecture for the vulnerability of authentication

Hacking. The Trusted Platform Module (TPM) is a security specification that allows hardware devices (e.g. sensors) to be resistant to attacks. Its implementation is available as a chip that is physically attached to a platform's motherboard and controlled by software running on the system using well-defined commands. It provides cryptographic operations such as asymmetric key generation, decryption, encryption, signing and migration of keys between TPMs, as well as random number generation and hashing. It also provides secure storage for small amounts of information such as cryptographic keys. Since the TPM is implemented in hardware and presents a carefully designed interface, it is resistant to software attacks [34]. However, mobile devices need greater attention if this were to be implemented on these devices as well. Figure 7 shows the responses for risk related to hacking. Again, data transmission is represented in the service concept of every layer.

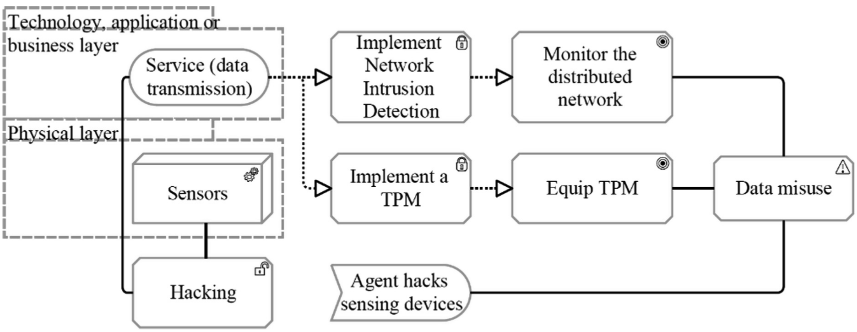


Fig. 7. Security architecture for the vulnerability of hacking

Viruses. Firewalls and anti-virus systems should be installed on systems being used in a smart city to localize potential attacks. These firewalls should increase the cost and reduce the success of these attacks thereby making these attacks less attractive.

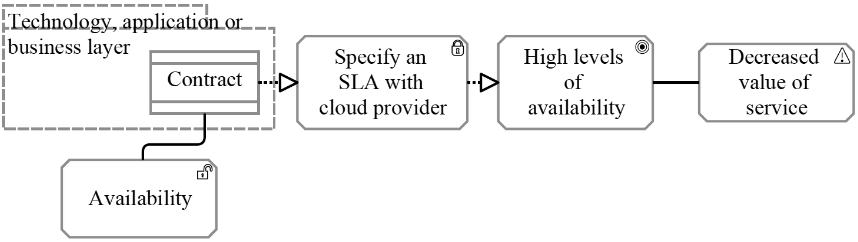


Fig. 8. Security architecture for the vulnerability of viruses

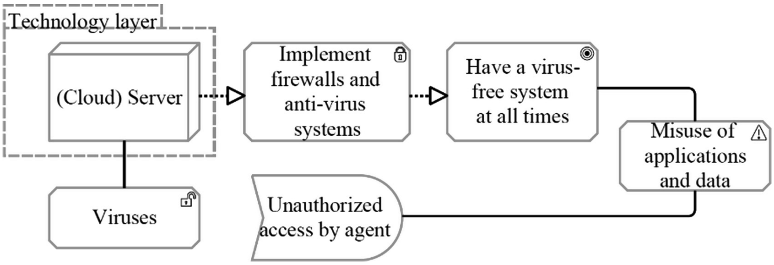


Fig. 9. Security architecture for the vulnerability of availability

Firewalls should permit only that traffic which is necessary for the intended applications and should hide all information about one network from the other [31]. Figure 8 shows the responses for this risk.

Availability. When using a cloud computing solution to expose smart services, we propose to implement a service level agreement (SLA) to ensure compliance by third party to uphold standards to a certain minimum [26]. This includes service availability, but also certain security responsibilities. Establishing a proper SLA is thus necessary to ensure that the third parties involved with the smart city initiative also implement the right safety measures. If the implemented cloud computing solution is hosted by the government itself, then the security measures needed to ensure secure data storage and access control need to be put in place by them. Frameworks by [12] and [41] can help in providing this. Figure 9 shows the responses for this specific risk.

4.5 Aggregating the Full Information Security Architecture

Section 4.4 shows how risks, requirements and responses are linked to each other and to concepts of the base-line architecture. Figures 3, 4, 5, 6, 7, 8 and 9 in totality show a complete information security architecture for smart cities. It shows the most likely points of attack as well as security measures that help strengthen these weak points. Stakeholders can monitor the security requirements and treat this activity as an integral part of the project using the architecture [24]. Every smart city project in itself can develop the architecture in more detail to encompass the exact implementation of that smart city. Drawing upon security issues that all smart cities face, this information security architecture for smart cities thus forms a blue print for smart city initiatives around the world.

5 Illustration

To illustrate the use of our approach in practice, a relevant smart city case study is used. The architecture is adjusted to this practical implementation of a smart city case to demonstrate how it can help in securing the technology in this specific case. The case study that is used concerns Smart City Lighting [29]. The monitoring of shared spaces, like public parks, provide a significant number of services valuable to a smart city. They include collecting information on functional performance, public security and safety-related data, infrastructure and maintenance requirements etc. Lampposts in outdoor shared spaces may be considered as suitable pieces of equipment to enable these services. In the case study, the above scenario is considered. Enhancements to the lamppost infrastructure through sensing, actuation and communication in order to facilitate the previously mentioned services are encouraged [1]. A number of objectives may be targeted here including energy saving, dangerous event localization and capturing information about a scene. Lampposts can be equipped with, or connected to, various sensors, like sound, light, and image sensors, in addition to LED lamps. By providing these services, there will be no need for extra personnel costs. At the same time lamppost can generate its own energy to cut down costs even further. To model this, the architecture is built from the bottom up. For each layer, we adapt the concepts of the base-line to the case to build an instance of the base-line architecture. Then, we analyze the base-line architecture to identify points of possible attacks according to our analysis in Sect. 4. Finally, we link these weak points to vulnerabilities that we have identified.

- The physical layer consists of sensors. Sensors in this case study represent sound sensors, light sensors and image sensors. Lampposts might be modelled individually. According to our analysis, when sensors are present, vulnerabilities are hacking and the interception of sensing messages.
- The technology layer shows the servers and the types of data being collected from these sensors. In this case light data, sound data and image data. Careful monitoring of possible viruses attacks should keep the servers up and running and the data safe.
- The application layer consists of software components that convert data to useful information required to provide smart services. In this case, special softwares like complex event processing (CEP), big data analytics, pattern prediction etc. are necessary to process data and convert it to useful information providing insights.
- The business layer consists of interfaces and business processes which deliver services to the stakeholders. In this case, some sort of interface should be available to some support staff who will be alerted when danger is present. Authentication is an issue here and correct responses should be implemented.

Furthermore, for every component within the base-line architecture, availability needs to be monitored. Also, the vulnerability of interception of messages between every layer needs to be addressed. In Fig. 10, the implementation of this case can be seen. Now, with the base-line architecture created and the vulnerabilities identified, practitioners can use the responses that are described in Sect. 4 to mitigate them. This case study shows us that smart city implementations can be modeled using our approach and exploiting the similarities in their structure.

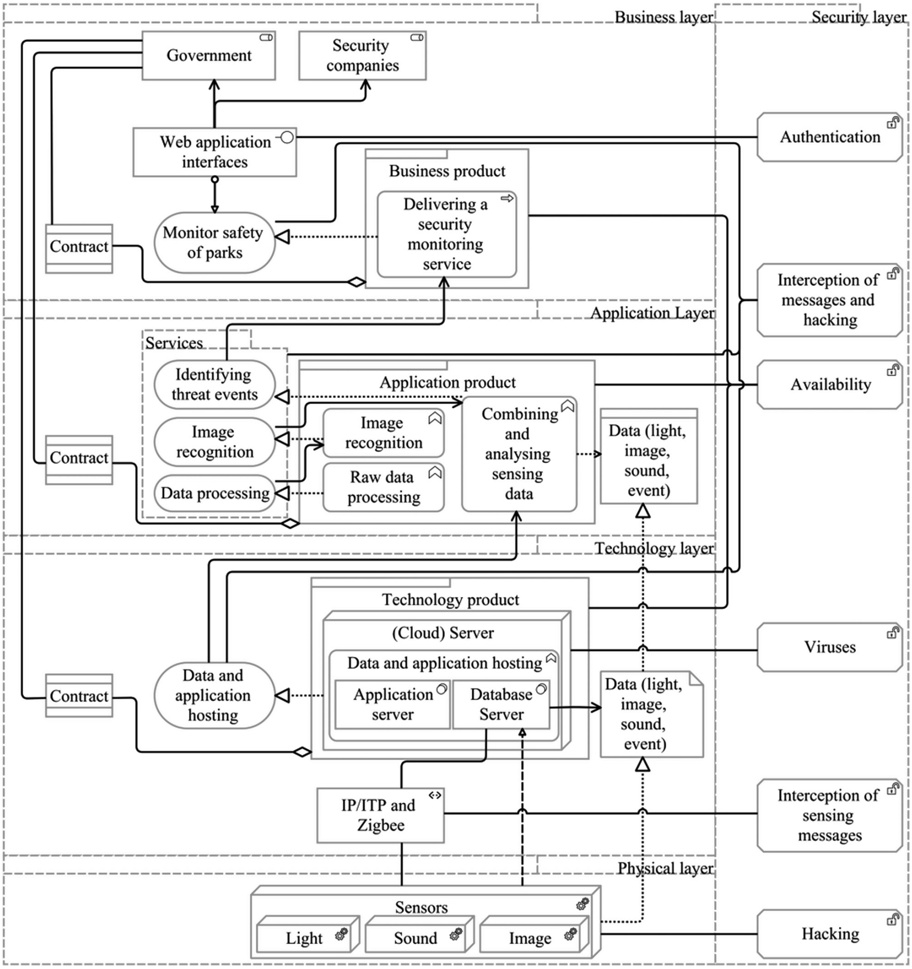


Fig. 10. The information security architecture applied to a smart lighting case study

6 Future Work and Conclusion

Future work should focus on a number of key aspects to further increase the value of this architecture:

- The architecture currently is static or a snapshot model. Future research can aim to make the architecture cover the lifecycle of a smart city project.
- The architecture at this moment shows high level Information Security for smart cities in its entirety. The architecture can however also be adopted to specific kinds of smart cities or even specific maturity levels of smart cities.

In this paper, three contributions to the current literature were made: (a) a generic base-line architecture for smart cities based on current smart city proposals, (b) an extensive security threat analysis of smart cities and (c) ways to mitigate these threats by presenting them in an information security architecture using Archimate. The proposed architecture can be used by smart city initiatives to accurately monitor their security and view this security as an integral part of each project. Without it, the benefits that the smart city could potentially provide will diminish as an unsafe integrated system could have disastrous consequences for the well-being of the citizens, companies and governments within such a smart city. The case study has shown that it is easy to adapt this architecture to a smart city implementation. However, further validation of the architecture is critical. It is advised to use this architecture when undertaking a smart city initiative and tailor it to the specific implementations.

References

1. Al-Anbuky, A. Sensor-actuator smart lighting system: system organizational concept and challenges. In: ICT4S 2014. Atlantis Press (2014)
2. Anthopoulos, L., Fitsilis, P.: From digital to ubiquitous cities: defining a common architecture for urban development. In: 2010 Sixth International Conference on Intelligent Environments (IE), pp. 301–306. IEEE (2010)
3. Band, I., et al.: Modeling Enterprise Risk Management and Security with the ArchiMate (2015)
4. Bartoli, A., et al.: Security and privacy in your smart city. In: Proceeding of the Barcelona Smart Cities Congress, pp. 1–6 (2011)
5. Batty, M., et al.: Smart cities of the future. Eur. Phys. J. Spec. Top. **214**(1), 481–518 (2012)
6. Bennett, C.H., Brassard, G.: Quantum Cryptography: Public Key Distribution and Con Tos5 (2014)
7. Blake-Wilson, S., Johnson, D., Menezes, A.: Key agreement protocols and their security analysis. In: Darnell, M. (ed.) Cryptography and Coding 1997. LNCS, vol. 1355, pp. 30–45. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0024447
8. Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44647-8_13
9. Caragliu, A., Del Bo, C., Nijkamp, P.: Smart cities in Europe. J. Urban Technol. **18**(2), 65–82 (2011)
10. Chourabi, H., et al.: Understanding smart cities: an integrative framework. In: 2012 45th Hawaii International Conference on System Science (HICSS), pp. 2289–2297. IEEE (2012)
11. Coplien, J., Hoffman, D., Weiss, D.: Commonality and variability in software engineering. IEEE Softw. **15**(6), 37–45 (1998)
12. Danwei, C., Xiuli, H., Xunyi, R.: Access control of cloud service based on UCON. In: Jaatun, M.G., Zhao, G., Rong, C. (eds.) CloudCom 2009. LNCS, vol. 5931, pp. 559–564. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10665-1_52
13. Deng, H., Mukherjee, A., Agrawal, D.P.: Threshold and identity-based key management and authentication for wireless ad hoc networks. In: Proceedings International Conference on ITCC 2004, vol. 1, pp. 107–111. IEEE (2004)
14. de Oliveira Albuquerque, R., et al.: A layered trust information security architecture. Sensors **14**(12), 22754–22772 (2014)
15. Dhillon, G., Backhouse, J.: Technical opinion: information system security management in the new millennium. Commun. ACM **43**(7), 125–128 (2000)

16. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Trans. Inf. Theory **22**(6), 644–654 (1976)
17. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proceeding of the 9th ACM Conference on Computer and Communications Security, pp. 41–47. ACM (2002)
18. Filipponi, L., et al.: Smart city: an event driven architecture for monitoring public spaces with heterogeneous sensors. In: 2010 Fourth International Conference on SENSORCOMM, pp. 281–286. IEEE (2010)
19. Gaur, A., Scotney, B., Parr, G., McClean, S.: Smart city architecture and its applications based on IoT. Procedia Comput. Sci. **52**, 1089–1094 (2015)
20. Jain, M.K.: Wireless sensor networks: security issues and challenges. Int. J. Comput. Inf. Technol. **2**(1), 62–67 (2011)
21. Jalali, R., El-Khatib, K., McGregor, C.: Smart city architecture for community level services through the internet of things. In 2015 18th International Conference on ICIN, pp. 108–113. IEEE (2015)
22. Jalali, S., Wohlin, C.: Systematic literature studies: database searches vs. backward snowballing. In: Proceedings of the ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, pp. 29–38. ACM (2012)
23. Jin, J., Gubbi, J., Marusic, S., Palaniswami, M.: An information framework for creating a smart city through internet of things. IEEE Internet Things J. **1**(2), 112–121 (2014)
24. Jonkers, H., Quartel, D.A.C.: Enterprise architecture-based risk and security modelling and analysis. In: Kordy, B., Ekstedt, M., Kim, D.S. (eds.) GraMSec 2016. LNCS, vol. 9987, pp. 94–101. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46263-9_6
25. Juels, A.: RFID security and privacy: a research survey. IEEE J. Sel. Areas Commun. **24**(2), 381–394 (2006)
26. Kandukuri, B.R., Rakshit, A.: Cloud security issues. In: IEEE International Conference on Services Computing, SCC 2009, pp. 517–520. IEEE (2009)
27. Lankhorst, M.: Enterprise Architecture at Work, 352 p. Springer, Berlin (2009). https://doi.org/10.1007/978-3-642-01310-2
28. Lavallee, M., Robillard, P.N., Mirsalari, R.: Performing systematic literature reviews with novices: an iterative approach. IEEE Trans. Educ. **57**(3), 175–181 (2014)
29. Merlino, G., et al.: A smart city lighting case study on an openstack-powered infrastructure. Sensors **15**(7), 16314–16335 (2015)
30. Mukherjee, B., Heberlein, L.T., Levitt, K.N.: Network intrusion detection. IEEE Netw. **8**(3), 26–41 (1994)
31. Murray, W.H.: Enterprise security architecture. Inf. Syst. Secur. **6**(4), 43–54 (1998)
32. Nam, T., Pardo, T.A.: Conceptualizing smart city with dimensions of technology, people, and institutions. In: Proceeding of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging times, pp. 282–291. ACM (2011)
33. Nardi, J.C., Almeida, J.P.A., et al.: Service commitments and capabilities across the ArchiMate architectural layers. In: 2016 IEEE 20th International EDOCW, pp. 1–10. IEEE (2016)
34. Perez, R., Sailer, R., van Doorn, L.: vTPM: virtualizing the trusted platform module. In: Proceedings of 15th Conference on USENIX Security Symposium, pp. 305–320 (2006)
35. Ralko, S., Kumar, S.: Smart City Security (2016)
36. Rivest, R.L., Shamir, A., Adleman, L.M.: U.S. Patent No. 4,405,829. U.S. Patent and Trademark Office, Washington, D.C. (1983)
37. SANS Institute (2013). www.sans.org.information-security.php

38. Suo, H., Wan, J., Zou, C., Liu, J.: Security in the internet of things: a review. In: 2012 international Conference on ICCSEE, vol. 3, pp. 648–651. IEEE (2012)
39. The Open Group: ArchiMate 3.0 Specification. Van Haren Publishing, Zaltbommel (2016)
40. van Drunen, H., Willemsz, E. (eds.): ArchiMate Foundation, Netherlands (2007)
41. Wang, C., Wang, Q., Ren, K., Lou, W.: Privacy-preserving public auditing for data storage security in cloud computing. In: 2010 Proceedings of IEEE INFOCOM, pp. 1–9. IEEE (2010)
42. Wenge, R., et al.: Smart city architecture: a technology guide for implementation and design challenges. China Commun. **11**(3), 56–69 (2014)
43. Muehlen, M.Z., Ho, D.T.-Y.: Risk management in the BPM lifecycle. In: Bussler, C.J., Haller, A. (eds.) BPM 2005. LNCS, vol. 3812, pp. 454–466. Springer, Heidelberg (2006). https://doi.org/10.1007/11678564_42

Three Categories of Context-Aware Systems

Boris Shishkov[1,4(✉)], John Bruntse Larsen[2], Martijn Warnier[3], and Marijn Janssen[3]

[1] Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Sofia, Bulgaria
b.b.shishkov@iicrest.org

[2] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark
jobla@dtu.dk

[3] Faculty of Technology, Policy, and Management, Delft University of Technology, Delft, The Netherlands
{M.E.Warnier,M.F.W.H.A.Janssen}@tudelft.nl

[4] Institute IICREST, Sofia, Bulgaria

Abstract. With regard to context-aware systems: some optimize system-internal processes, based on the context state at hand; others maximize the user-perceived effectiveness of delivered services, by providing different service variants depending on the situation of the user; still others are about offering value-sensitivity when the society demands so. Even though those three perspectives cover a broad range of currently relevant applications there are no widely accepted and commonly used corresponding concepts and terms. This is an obstacle to broadly understand, effectively integrate, and adequately assess such systems. We address this problem, by considering a (component-based) methodological derivation of technical (software) specifications based on underlying enterprise models. That is because context states are about the enterprise environment of a (software) system while the delivery of context-aware services is about technical (software) functionalities; hence, we need a perspective on both. We consider the SDBC (Software Derived from Business Components) approach that brings together enterprise modeling and software specification. On that basis: (a) We deliver a base context-awareness conceptualization; (b) We partially align it to agent technology because adapting behaviors to environments assumes some kind of pro-activity that is only fully covered by agent systems, in our view. We partially illustrate our proposed conceptualization and particularly - the agent technology implications, by means of a case example featuring land border security.

Keywords: Modeling · System design · Context-awareness · SDBC AORTA

1 Introduction

Whenever a group of *entities* collectively realize a *goal*, we consider them to belong to a **system** [17]. An **adaptable system** has the ability to adjust to new conditions [8]. We argue that an essential feature of adaptable systems is **context-awareness**

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 185–202, 2018.
https://doi.org/10.1007/978-3-319-94214-8_12

[22] – this is *adjusting the system behavior depending on the situation at hand* (**context state**). Our observation is that **context-aware systems** are all about adjusting "something" to the *context state*; however, what is adjusted differs: (i) Some context-aware systems optimize system-internal processes based on the context state at hand [3, 12], for example: *regulating the electro-consumption of home appliances for the sake of keeping the overall building consumption within some boundaries*. (ii) Other context-aware systems maximize the user-perceived effectiveness of delivered services, by providing different *service variants* depending on the situation of the user [1], for example: *treating a distantly monitored patient in one way when his/her condition is normal and in another way in case of emergency*. (iii) Still other context-aware systems are about offering value-sensitivity when the society demands so [7], for example: *in the case of supporting judiciary processes, different levels of transparency are to be provided to different categories of stakeholders*. Not claiming exhaustiveness, we argue that those three context-awareness perspectives cover a broad range of currently relevant applications, while corresponding system categorizations are observed to be missing in literature, especially as it concerns real-life (business) processes. Hence, we propose and elaborate (in the following section) the above **categorization**.

Further, there are *no widely accepted and commonly used concepts and terms* with respect to (i), (ii), and (iii). This is considered to be an obstacle with regard to *broadly understanding*, *effectively integrating*, and *adequately assessing* such systems. We address this problem, by taking a (component-based-) technical/software design perspective, assuming a desired *methodological derivation* of the **technical** (**software**) **specifications** on the basis of underlying **enterprise models**. That is because *context states* are about the *enterprise environment* of a (software) system while the delivery of *context-aware services* is about *technical* (*software*) *functionalities*; hence, we need a perspective on both. Stepping on *previous work*, we consider the **SDBC** (**Software Derived from Business Components**) approach that brings together *enterprise modeling* and *software specification* [17, 18, 20]. On that basis:

- We deliver a base context-awareness conceptualization (inspired by the current discussion and the analysis carried out in Sect. 2) that is claimed to hold for all (i), (ii), and (iii).
- We partially align that conceptualization to agent technology [11, 27] because adapting behaviors to environments is considered to assume some kind of pro-activity that is only fully covered by agent systems, in our view.

We partially *illustrate* our proposed conceptualization and particularly - the agent technology implications, by means of a *case example* featuring **land border security**.

The remaining of the current paper is organized as follows: Sect. 2 elaborates and discusses the above-mentioned context-awareness perspectives (*optimizing internal processes*, *maximizing the user-perceived effectiveness*, *being value-sensitive*). In Sect. 3, we: (a) briefly *outline the SDBC approach* that gives the general methodological guidelines we follow; (b) elicit (in line with SDBC) the *base context-awareness system concepts*, referring to the analysis in Sect. 2; (c) enrich those concepts from the *AORTA* (*agent-technology*) *perspective*; (d) distill on that basis a *meta-model* that is considered a key contribution of this paper and propose guidelines refinements. Assuming the SDBC methodological guidelines and referring to a real-life case study that is featuring the usage of drones in support of land border security, we partially illustrate in Sect. 4 the meta-model and the AORTA framework influence, emphasizing the applicability with regard to all three categories of context-aware systems, as discussed above. We justify the adequacy of the delivered contribution in Sect. 5, by analyzing *related work*. Finally, in Sect. 6 we *conclude* the paper.

2 System Behavior Perspectives

Referring to the notions considered in the Introduction, we will firstly elaborate on the context-driven optimization of system-internal processes (Subsect. 2.1), secondly – on the context-driven maximization of the user-perceived effectiveness (Subsect. 2.2), and finally, on the context-driven value-sensitivity (Subsect. 2.3). It is often that the context-awareness is enabled by sensor technology [21] allowing to "know" what is happening around; alternatively, there should be other ways of "sensing" the environment [1]. As it concerns (i), (ii), and (iii) – see the previous section – this counts for all of them. Further, considering the essence of their underlying system behaviors, we use the following labels: SELF-MANAGING CONTEXT-AWARE SYSTEM for (i); USER-DRIVEN CONTEXT-AWARE SYSTEM for (ii), and VALUE-SENSITIVE CONTEXT-AWARE SYSTEM for (iii). Finally, even though most often context-aware systems are "sensitive" to changes in the system environment, it is also possible that the "sensitivity" is towards internal issues (things happening inside the system (not in the environment) may trigger either internal optimizations, or changes in the services delivered to the user, or a reconsideration of the "covered" values). In this paper, we are not restrictive with regard to the "sensitivity" (whether it concerns the system or the environment).

2.1 Self-Managing Context-Aware Systems (SMCAS)

SMCAS' context-awareness is directed towards **internal** (**system**) **optimization purposes** [16]. Such **autonomic** solutions [12] are proposed as a way to reduce the cost of maintaining complex systems, and to increase the human ability to manage these systems properly, by automating (part of) their working. In essence, **self-managing** system can be characterized by a **feedback loop mechanism** that allows them to optimize their working based on input from the environment. For the basic *feedback loop*, the system receives *input from the environment* (monitor) and can change its behavior which in turn has an *effect on the environment* (effector). In **Autonomic Computing** [10] this basic loop is extended into *four components*, resulting in the **MAPE Cycle**, see Fig. 1 (left). Next to the "monitor" and "effector" phases, the system internally has an analyze phase that *processes environmental input* and a plan phase that *changes the internal and external working of the system*.

Taking this one step further, an **autonomic system** can manage another system (the **managed system**) by placing it in the *MAPE Cycle* as shown in Fig. 1 (right). Placed in such a configuration, the *autonomic* and *managed* systems together form a **SELF-MANAGING SYSTEM** or *self-adaptive system* [14].

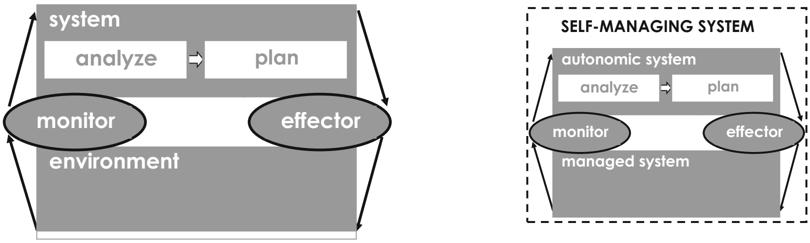


Fig. 1. Monitor-Analyze-Plan-Effect (MAPE) cycle (left); A *managed system* and an *autonomic system* that together form a *self-managing system* (right)

Internally *self-managing systems optimize their behavior based on inputs to and outputs from the managed system*. Such state updates can range from simple **if-then rules** (for example: *if the temperature is below zero degrees, then pre-heat the car*) to more sophisticated approaches such as those based on **machine learning techniques** (e.g., *neural networks* or *inference engines that determine the best action based on a large internal knowledge base*). Thus, the main **objective of self-managing systems** is to optimize their internal working based on inputs from the environments.

2.2 User-Driven Context-Aware Systems (UDCAS)

The UDCAS' context-awareness is directed towards the **maximization of the external (user) satisfaction** [1]. Hence, such systems should be able to: (i) identify the situation of the user (possibly through *sensors*); (ii) deliver a service to the user, that is suited for the particular situation, as illustrated in Fig. 2 (left).

As it is seen from the figure, a **service** is delivered to the **user** and *the user is considered within his or her context*, such that *the service is adapted on the basis of the context state* (or situation) the user finds himself/ herself in. That state is to be somehow *sensed* and often technical devices, such as sensors, are used for this purpose. UDCAS actually deliver services to the user by means of ICT (Information and Communication Technology) applications [17] (**applications**, for short). Hence, unlike "traditional" applications assuming that users would have common requirements independent of their context, *user-driven context-aware* applications are **capable of adapting their behavior to the situation of the user** (this is especially relevant to services delivered via mobile devices). Hence, such applications are, to a greater or lesser extent, **aware of the user context situation** (for example, *user is at home*, *user is traveling*) and **provide the desirable services corresponding to the situation at hand**. This quality points also to another related characteristic, namely that *user-driven context-aware* applications must be able to capture or be informed about information on the context of users, **preferably without effort and conscious acts from the user part** [18]. Hence, a basic *assumption* underlying the development of *user-driven context-aware* applications is that **user needs are not static**, however partially dependent on the particular situation the user finds himself/herself in, as already mentioned. For example, depending on his/her current *location*, *time*, *activity*, *social environment*, *environmental properties*, or *physiological properties*, the user may have different *interests*, *preferences*, or *needs* with respect to the *services* that can be provided by *applications*.

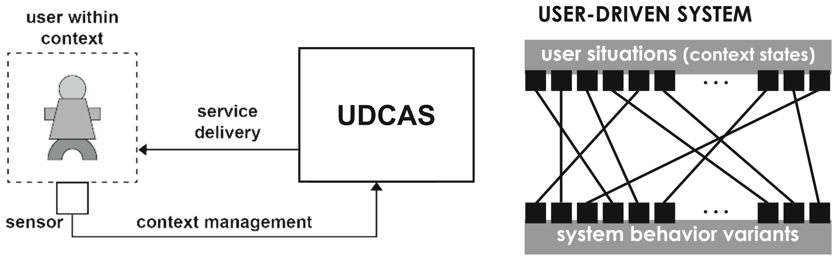


Fig. 2. The UDCAS vision (inspired by [18, 22]): a schematic representation (left); Context-states-driven system behavior variants (right)

User-driven context-aware applications are thus primarily motivated by their **potential to increase the user-perceived effectiveness**, i.e. to *provide services that better suit the needs of the user, by taking account of the user situation*. We refer to the collection of parameters that determine the situation of a user, and which are relevant for the application in pursue of user-perceived effectiveness, as *user context*, or **context** for short, in accordance to definitions found in literature [5].

Finally, UDCAS are hence about delivering **behavior variants** to corresponding user situations (**context states**), as illustrated by Fig. 2 (right). The idea is that for each *context state*, the system has a *behavior variant*. Nevertheless, this would not be always realistic because if the possible *context states* are too many (for example: tens or hundreds), the effort for "preparing" (at design time) *behavior variants* would be huge. For this reason, *statistical data analysis* and *probability studies* are needed, as studied in [17], for establishing the **context states of high occurrence probability** (for example, during a normal working day, an employee would most probably be either *at work* or *at home*, or *traveling*, and it is not very likely that the employee is somewhere else, especially during day hours). Hence all other (possibly hundreds) low probability *context states* would not need to be addressed at design time; instead, a "collecting" *context state* (for example, labelled: OTHER) may be considered at design time, assuming a more *generic behavior algorithm*, such that the behavior is "tuned" at real time (possibly in a *rule-based way*). This would of course lead to lower quality-of-service which is nevertheless justified since it would be very rare that the OTHER behavior variant is triggered. Thus, this is a matter of "trade-off" between *quality-of-service* and *resources*.

2.3 Value-Sensitive Context-Aware Systems (VSCAS)

VSCAS' context-awareness is directed towards a **sensitivity to public values** [7]. *Public values* (**values** for short) like *privacy* and *data protection*, *security*, *accountability*, *integrity* and *provenance*, and *sustainable data storage*, often need to be incorporated in the functionalities of *information systems*. Hence, the first question to be answered is: How do values relate to requirements (we mean particularly *non-functional requirements* because values are essentially *non-functional*)? In answering this question, we refer to [19] who argue that values are desires of the general public (or public institutions/organizations that claim to represent the general public), that are about properties considered societally valuable, such as *respecting the privacy of citizens* or *prohibiting polluting activities*. Even though values are to be broadly accepted (that is why they are **public**), they may concern individuals (for example: considering privacy). Hence, put broadly, values concern the *societal expectations with regard to the way services should be delivered* and with regard to the above question: *values* are *desires* or *goals*, not *requirements*. Values are abstract and not directly related to an enterprise or software system, as opposed to requirements. Moreover, values are construct by and for society and not by and for the enterprise domain in which a specific system will be used. Those domains may overlap but are not the same. Values that are adopted as goals by an enterprise would therefore impact the requirements on a system that the enterprise wants to introduce in order to realize its goals. For this reason, *the impact of values cannot be limited to non-functional requirements*. It is therefore considered important to clearly distinguish *values* from *requirements* and acknowledge the limitations of requirements engineering with regard to the development of value-sensitive (software) systems.

Hence, the development of systems should take into account the objective, the user needs, but also the operating and societal context. In this way *values* can be used as a *guidance for making choices when developing systems*; looking at possible *tensions* among values is an issue as well. Thus, we have identified challenges in several directions:

Firstly, values are normative by nature and different stakeholders might prefer different values; also, values might differ among countries and cultures.

Secondly, even though different societies may agree on a value at high-level, their cultural and other differences may "push" for different value "realizations". This may result in different operationalizations and implementations of the same value.

Thirdly, values may be conflicting to each other (for example: fulfilling two values at the same time may be impossible). Searching for criminals might lead to violating the privacy of innocent people, for example.

Thus, in considering VSCAS, it is important to be aware that **different context states may assume the consideration of different values**, which in turn would mean **different system functionality variants**. Nevertheless, this goes beyond a mapping just between *values* and *non-functional requirements* and would assume a *broader consideration of the software functionalities specification*.

Overall: *computing power becomes larger*, *wireless telecommunications are advancing*, and *sensor technology is developing fast* [17]; this allows for *ubiquitous network connectivity* and numerous *capabilities of smart devices*, as a basis for developments in several directions: (a) **Systems that are traditionally designed for one specific situation and task can be augmented to become "smarter", being able to operate in complex environments**. (b) **Systems are empowered to "sense" what is going on inside them and also what is going on with the end user while (s)he is utilizing corresponding services**. This concerns the system-internal processes, the way services are delivered to users, and the way values are considered, as discussed in the current section. In the following section, we will present our *proposed way of modeling context-aware systems*, taking into account those *three system behavior perspectives*.

3 Proposed Modeling and Design

Furthering previous work, we consider SDBC (see Sect. 1) as the approach of choice (generally) because of several reasons: its *strengths in aligning enterprise modeling and software specifications*, its *component-orientation* and support for *re-use*, and its *previous use for specifying context-aware and privacy-sensitive systems* [17, 18, 20]. We will thus briefly introduce SDBC in this section. We will then present *the main concepts* we consider, making sure that they are consistent with SDBC and relevant to the three categories of context-aware systems, as considered in the previous section; we will also provide a *conceptual enrichment from an agent-technology perspective* (see Sect. 1). We will then reflect this in a *meta-model* derived accordingly. And in the end, we will *narrow the design scope* (in the above context) to only touch upon issues that concern the key features of the considered three categories of context-aware systems.

3.1 SDBC

SDBC is an *approach* (consistent with *MDA* [15]) that is focused on the *derivation of software specification models on the basis of corresponding (re-usable) enterprise models*. SDBC is based on three key ideas: (i) The software system-to-be is considered in its enterprise context, which means that the *software specification models* are to stem from corresponding *enterprise models*; this means in turn that a deep understanding is needed on real-life (enterprise-level) *processes*, corresponding *roles*, *behavior* patterns, and so on. (ii) By bringing together two disciplines (*enterprise engineering* and *software engineering*), SDBC pushes for applying *social theories* in addressing enterprise-engineering-related tasks and for applying *computing paradigms* in addressing software-engineering-related tasks, and also for integrating the two, by means of sound methodological guidelines. (iii) Acknowledging the value of *re-use* in current software development, SDBC pushes for the identification of re-usable (generic) *enterprise engineering building blocks* whose *models* could be reflected accordingly in corresponding *software specification models*. We refer to [17] for information on SDBC and we are reflecting the SDBC outline in Fig. 3.

As the figure suggests, there are two SDBC modeling milestones, namely **enterprise modeling** (*first milestone*) and **software specification** (*second milestone*). The first milestone has as input a case briefing (the initial (textual) information based on which the software development is to start) and the so called domain-imposed requirements (those are the domain regulations to which the software system-to-be should conform). Based on such an input, an analysis should follow, aiming at structuring the information, identifying missing information, and so on. This is to be followed by the identification (supported by corresponding social theories) of enterprise modeling entities and their inter-relations. Then, the causalities concerning those inter-relations need to be modeled, such that we know what is required in order for something else to happen [23]. On that basis, the dynamics (the entities' behavior) is to be considered, featured by *transactions* [17]. This all leads to the creation of *enterprise models* that are elaborated in terms of *composition*, *structure*, and *dynamics* (all this pointing also to corresponding *data* aspects) – they could either "feed" further software specifications and/or be "stored" for further use by enterprise engineers. Such enterprise models could possibly be reflected in corresponding **business**

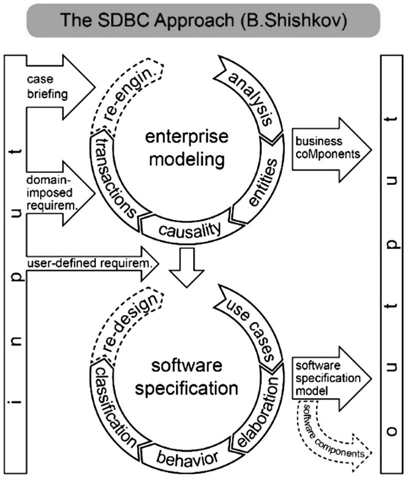


Fig. 3. SDBC - outline (Source: [20], p. 48)

coMponents (models of *business components* [17]). Next to that, re-visiting such models could possibly inspire enterprise re-design activities, as shown in Fig. 3.

Furthermore, the second milestone uses as input the enterprise model (see above) and the so called user-defined requirements (those requirements reflect the demands of the (future) users of the software system-to-be towards its functioning).

That input "feeds" the derivation of a use case model featuring the software system-to-be. Such a software specification starting point is not only consistent with the **Rational Unified Process** - **RUP** [13] and the **Unified Modeling Language** – **UML** [25] but is also considered to be broadly accepted beyond RUP-UML [17]. The *use cases* are then elaborated, inspired by studies of Cockburn [4] and Shishkov [17], such that software behavior models and classification can be derived accordingly. The output is a *software specification model* adequately elaborated in terms of *statics* and *dynamics*. Applying *de-composition*, such a model can be reflected in corresponding *software components*, as shown in the figure. Such an output could be inspiration for proposing in the future software re-designs, possibly addressing new requirements.

Further, in bringing together the first milestone of SDBC and the second one, we need to be aware of possible *granularity mismatches*. The enterprise modeling is featuring business processes and corresponding business coMponents but this is not necessarily the level of granularity concerning the software components of the system-to-be. With this in mind, an ICT **application** is considered as matching the granularity level of a *business component* – an ICT application is an implemented software product realizing a particular functionality for the benefit of entities that are part of the composition of an enterprise system and/or a (corresponding) enterprise information system. Thus the label **software specification model**, as presented in Fig. 3, corresponds to a particular ICT application being specified. **Software components** in turn are viewed as *implemented pieces of software*, which represent parts of an ICT application, and which collaborate among each other driven by the goal of realizing the functionality of the application (functionally, a software component is a part of an ICT application, which is self-contained, customizable, and composable, possessing a clearly defined function and interfaces to the other parts of the application, and which can also be deployed independently). Hence: a **`software coMponent`** is a conceptual specification model of a software component; the second SDBC milestone is about the identification of software coMponents and corresponding software components.

3.2 Conceptualization

In this sub-section, we will firstly consider the basic concepts aligned with SDBC and the system behavior perspectives, as considered in the previous section.

BASIC CONCEPTS (GENERAL)

SYSTEM: A collection of elements possibly interacting with each other, driven by the purpose of delivering a service to another entity or group of entities.

SUB-SYSTEM: A system part identified based on a functional decomposition with regard to the system (hence, a system can optimize itself, by optimizing corresponding sub-systems).

ENVIRONMENT: Anything not belonging to a system belongs to the system environment
Part of the environment concerns those environmental entities that are assumed to have some interaction with the system.

ENTITIES: Composition elements with regard to a system / environment
- Human vs Artificial entities;
- Passive (a passive entity is an entity that only performs actions when another entity interacts with it) vs Autonomous (an autonomous entity is an entity that performs actions on its own initiative) entities;
- Sensing (capturing context data in support of the system's service delivery) vs Actuator (causing changes in the environment on behalf of the system) entities.

ROLE: The state of carrying out certain objectives
- The entity-role combination is labelled "actor-role".

ACTOR: An autonomous entity that can enact a role.

USER: An actor that the system services.

ACTION: Something done by an entity
- A sequence of actions, occurring between two entities that are collaborating in support of the service delivery, is labelled "inter-action".

OBJECTIVE: The motive behind the service delivery.

REGULATIONS: Reflection of the existing norms that have impact on the service delivery, prescribing what is allowed in some situations, forbidden in others, and so on.

VALUES: Reflection of the public perception towards what is important regarding the service delivered by the systems*
* We address values that are shared among environmental human entities.

BASIC CONCEPTS (FEATURING SYSTEM ADAPTATION DIMENSIONS)

SELF-MANAGING BEHAVIOR: Context-driven enforcement of syst.-internal optimizations.

USER-DRIVEN BEHAVIOR: Context-driven service adaptation based on the user situation.

VALUE-SENSITIVITY: Service adaptation inspired by values, delivered through the operationalization of values.

Wishing to enrich those concepts from an agent-technology perspective (see the Introduction) **AORTA** is partially considered as a meta-model for **enabling agents to perform organizational reasoning** based on the *OperA* model [6] for agent organizations [11]. *Organizational reasoning* enables an agent to identify the objectives it is expected to solve, the other agents that it depends on for solving those objective, and whether an action is permitted, obliged, allowed or forbidden. In doing so the social expectations are made explicit through the AORTA concepts, which is considered an advantage when creating and reiterating the design of agents. Those *concepts* are shown below in the form of logical predicates:

CONCEPT	MEANING
role(Role, Objs)	Role is the name of a role and Objs is a set of main objectives of that role.
obj(Obj, SubObjs)	Obj is an objective that has SubObjs as a set of sub-objectives.
dep(Role_1, Role_2, Obj)	Role_1 depends on Role_2 in order to complete Obj.
rea(Ag, Role)	Agent Ag enacts Role.
cond(Role,Obj,Deadline,Cond)	When the condition Cond holds, Role is obliged to complete Obj before the objective Deadline.
obl(Ag, Role, Obj, Deadline)	Agent Ag is obliged to enact Role to complete Obj before Deadline.
viol(Ag, Role, Obj)	Agent Ag enacting Role has violated the obligation to complete Obj.

A *role* defines a *role* in the *organization*, and the *primary objectives* associated with that *role*. By *enacting a role*, an *agent* announces to other *agents* in the *organization* that it takes upon itself that *role* and thus that other *agents* can expect it to work towards solving the *objectives* associated with that *role*. It is also possible to make an explicit notion for how an *objective* can be *decomposed* into *sub-objectives*, and how a *role* may depend on other *roles* to solve an *objective*. Finally, *normative statements* are supported that define *conditions* on how *agents* should solve an *objective* in a certain *context*.

AORTA divides organizational reasoning into **three phases**: obligation check (OC), option generation (OPG) and action execution (AE). In the OC-phase the agent uses its beliefs to *check if obligations are activated*, *satisfied* or *violated*, and updates its beliefs accordingly. Following that it *generates possible organizational options* in the OPG-phase, which it then considers in the AE-phase when *deliberating an action*. Figure 4 visualizes such a reasoning "component".

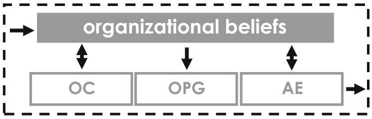


Fig. 4. Reasoning component

Further, in specializing some of the basic concepts (see above) from an agent-technology perspective (consistent with AORTA), we consider the following:

- Autonomy: agents operate without the direct intervention of humans or others, and have some kind of control of their actions and internal state;
- Social "ability": agents interact with other agents (and possible humans) via some kind of agent communication language;
- Reactivity: agents perceive their environment (which may be the physical world, a user via a graphical user interface, a collection of other agents, the Internet, or perhaps all of those combined), and respond in a timely fashion to changes that occur in it;
- Pro-activeness: agents do not simply act in response to their environment, they are also capable of exhibiting goal-driven behavior, by taking the initiative.

Hence, we define an **agent** as an **entity that is capable of operating autonomously, interacting with other agents, perceiving and reacting to changes in the environment, and taking actions toward solving an objective**.

Next to that, it is considered common to have **agent systems** where *an agent represents a larger group of agents*; such an agent is often tasked with *coordinating* the other agents in the group (such tasks can be captured with the notions of *role* and *objective*, with *agents enacting roles*). Thus, by applying the *agent* concept, we could conveniently capture the notions of *autonomy*, *social ability*, *reactivity* and *pro-activeness* that we find useful with regard to the challenge of modeling complex (context-aware) systems where it is not straightforward understanding how every system part works in detail.

Finally, we present below the enrichment (from an **agent-technology perspective**) with regard to the basic concepts that were presented above:

SYSTEM: A *system* is represented through the notion of an *agent* that enacts a designated system *role*. The *role* specifies the *objectives* (**obj**) of the system, reflecting its *purpose*. These *objectives* can be decomposed into *sub-objectives* by *obj*-statements. We decompose the *objective* of the *system* into *sub-objectives*, matching the *objectives* of the *sub-systems*. We show the *dependencies* (**dep**) between the *system* and its *sub-systems* by *dep*-statements.

SUB-SYSTEM: We represent individual *sub-systems* by *agents* enacting *roles*, specifying the *sub-objectives* of the system that they address. We also make explicit the dependency between the *system* and its *sub-systems* by *dep*-statements.

ENVIRONMENT: We represent the parts of the *environment* that we assume interact with the *system* by *agents*. An **environmental agent** may enact a *role*, meaning that the *system* can expect it to carry out a certain *objective*, but this is not required.

ENTITIES: Inspired by the agent notion as considered in [Wolldr95], we represent entities in general by **agents** and **roles**

- **human** vs **artificial** entities – no distinction; an *agent* can represent a *human* and implement a *human behavior model* but from a model perspective, *human* and *artificial agents* are indistinguishable;
- **passive** vs **autonomous** entities; *agents* that implement *proactive behavior* models are inherently capable of *autonomous behavior*, whereas *agents* that implement *reactive behavior* models are suited for representing *passive* entities;
- **sensing** vs **actuator** entities; we have *roles* dedicated to *sensing* and *actuator* such that we represent these entities by *agents* enacting those *roles*.

ROLE and ACTOR-ROLE: We represent *roles* and their designated *objectives* by *role-statements*, and the *entity-role combination* by *rea-statements*, stating that an *agent* enacts a *role*.

ACTOR: We represent an *actor* by its designated *agent*.

USER: The *user* is a specific *agent* that plays the *role* of *user*. For *context-aware systems*, the system *agent* depends on the *user* to deliver its *service* and for gaining *feedback*. The *user role* may have certain *objectives* but this is not required.

ACTION and INTERACTION: An *agent* is capable of performing *actions*, and in the **OPG**-phase, it *generates options* based on its *beliefs*. It then *deliberates the action* to take in the **AE**-phase. We represent the concept *interaction* by *dep*-statements, meaning that multiple *agents* are collaborating in solving an *objective*.

OBJECTIVE: We represent *objectives* primarily by *role*- and *obj*- statements.

REGULATIONS: We represent *regulations* by *condition* statements, stating *situations* where *obligations* to carry out *objectives* get activated. In the **OC**-phase, the *agent* updates itself for current active *obligations* and violated *obligations* with *obl*- and *viol*- statements.

VALUES: *Values* are reflected in the *regulations* expressed by *cond*-predicates.

3.3 Proposed Meta-Model

After having introduced our *concepts* (in synch with SDBC), we have derived a **meta-model** accordingly, considered a key contribution of the current paper. The meta-model is presented in Fig. 5, using the notations of UML – Class Diagram [25].

As it is seen in the figure, we consider a **system** and its **environment**. Both are composed of numerous **entities** which in turn can be **components** (*non pro-active*) or **agents** (*pro-active* and *intelligent*). One *entity* (an *agent*, for example) can enact many different **roles** (and in this research, we limit ourselves to *four role categories*, namely: **user**, **sensor**, **actuator**, and **processor**) that are restricted by corresponding **rules** and are subject of **regulations**. A *regulation* in turn is composed of many *rules* and is affecting not only the *roles* but the *system* as a whole.

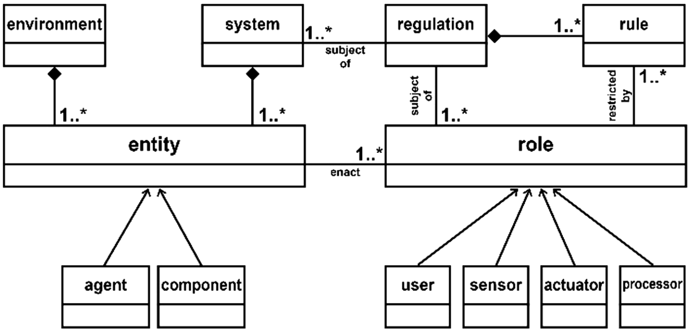


Fig. 5. Proposed meta-model

Since we are taking particularly an agent perspective, we consider it important **matching roles to their corresponding executing agents** because it is not for sure that *anybody* would have the right capabilities of fulfilling a *role*.

With regard to what an agent is capable of doing, *AORTA* takes a discrete approach to capabilities: **the capabilities of an agent are defined as the set of states that an agent can achieve**. More concretely, an agent is defined as the tuple (**α**, **MS**, **AR**, **F**, **C**, **μ**) where **α** is the *name of the agent*, **MS** is its *"mental" state*, **AR** are its *rules of reasoning*, **F** is a s*et of transitioning functions* (Action × MS → MS), **C** are its *capabilities* and **μ** is a *"mailbox"* for incoming messages from and outgoing messages to other agents.

When considering a role enactment, an *agent checks whether its capabilities overlap the objectives of that role* (if not, an *enactment* would be impossible). One can *split a role into two roles* where one defines what objectives are expected of an agent enacting the "partial" role and what is expected of an agent enacting the "full" role.

As mentioned in the Introduction, we propose at the end of this section *refined guidelines* (inspired by SDBC and the meta-model).

3.4 Refined Design Guidelines

In following SDBC (see Fig. 3) and implementing the meta-model (see Fig. 5), we propose refined design guidelines, and we limit ourselves only to considering the *strengths of agent technology with regard to the three categories of context-aware systems addressed* in the current paper.

The *three system behavior perspectives* (*SMCAS*, *UDCAS*, and *VSCAS* – see Sect. 2) considered in combination, referring to the *meta-model*, inspire a **proposed design vision**; according to it, a *system* uses an **AORTA Engine** considered for doing all three things simultaneously, as it is illustrated in Fig. 6.

As the figure suggests, the **user** is situated in the **environment** and the **system** uses **sensors** to receive info from the *environment*, including the *user*, and **actuators** to manage **sub-systems**, realizing *internal optimizations*. The **AORTA Engine** allows the *system* to conform to **rules** and **regulations** (*regulations* are encoded, reflecting the *values* the *system* should consider). The *AORTA Engine* checks the input from the **Analysis Engine** to identify *obligated*, *forbidden* or *allowed actions*, and what *organizational influence* those actions have (enacting a *role* involves notifying other *entities*). The output is provided as input to the **Planner** that in turn decides the *action* to be taken.

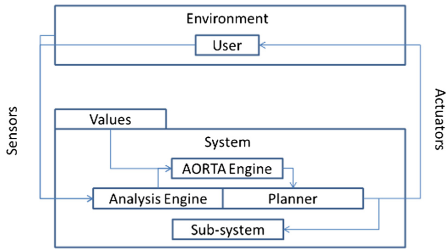


Fig. 6. A system combining features of SMCAS, UDCAS, and VSCAS

4 The Border Security Case

Unmanned aircraft (for example, **drones**) are a way to facilitate the surveillance along land borders, as according to a case example considered in [21].

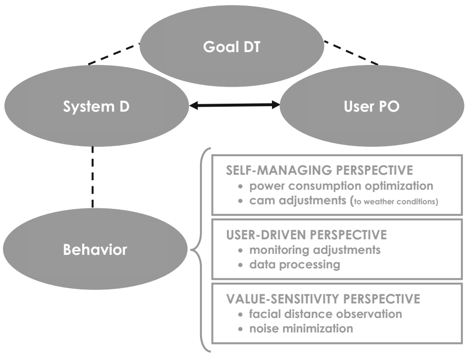


Fig. 7. Exemplifying the SMCAS, UDCAS, and VSCAS perspectives

Drones are capable of carrying and running infrared (and regular) surveillance cameras and based on their input, scarce capacity can be allocated. The goal is to find illegal activities. Despite drones' advantages, there can be some unintended effects: Drones can make noise and scare ordinary people who are passing the borders in a legitimate way, often at check-points; custom officials and/or police officers might be scared and/or disgraced by drones. Next to that, (video) recording activities might be violating the privacy of people. Hence, **it should be avoided that**: (i) drones come too close to legitimate activities; (ii) data that is not needed is recorded and shared. This asks for balancing between the *internal objectives of finding illegal activities* and the *societal demands* that are two-fold: it is needed *to avoid drones getting too close to people* and it is also needed *to ensure their privacy*.

We hence make the following explicit: (A) We do address **SYSTEM D** – *a drone system*, featuring a drone flying over a border and supporting border police officers; (B) *System D* is delivering services to **USER PO** – *the border police officers* who are patrolling the border; (C) The goal in context is: **GOAL DT**, featuring the *detection of illegal activities* in general (trespassers, in particular). This is reflected in Fig. 7.

As also seen from the figure, we have identified six case-specific behavior lines, two per each of the system behavior perspectives:

[POWER CONSUMPTION OPTIMIZATION] [SMCAS PERSPECTIVE]: When a drone is flying, it is exchanging information with the ground station – the drone is sending sensor data to the station that is in turn generating instructions. Hence, it is at the ground station where the flight is controlled such that it is guaranteed that the drone mission is completed. This requires that the power consumption is optimized, such that the drone has enough power to complete the mission and get back.

[CAM ADJUSTMENTS] [SMCAS PERSPECTIVE]: When a drone is flying, it counts on its cameras, such that it is capable of adjusting its flight accordingly. Hence, those cameras need to be adjusted when weather conditions change.

[MONITORING ADJUSTMENTS] [UDCAS PERSPECTIVE]: When a drone is flying, it is realizing its mission to support border police officers (the "user") and for this, it should adjust its monitoring to mainly cover those areas that are not close to where police officers are.

[DATA PROCESSING] [USCAS PERSPECTIVE]: When a drone is flying, raw sensor data is processed by the drone itself and/or by the ground station, such that higher-level context information is derived and delivered to border police officers.

[FACIAL DISTANCE OBSERVATION] [VSCAS PERSPECTIVE]: When a drone is flying, it is to act in a privacy-sensitive way, which means that without explicit instructions, the drone should not (video) capture people's facial information or if this happens, the photo/video material should be blurred accordingly.

[NOISE MINIMIZATION] [VSCAS PERSPECTIVE]: When a drone is flying, it should avoid noise-polluting residential environments, which means that if this wouldn't be mission-critical, the drone should avoid approaching residential areas.

Acting adequately with regard to all those perspectives is considered important because it wouldn't be acceptable neither "sacrificing" the drone, nor compromising the mission, nor disregarding values. Nevertheless, it is sometimes impossible to satisfy all this. For example: If from a SMCAS perspective, the drone should immediately turn back (with indications of insufficient power to go on) but from a UDCAS perspective, the drone should go on to approach an area of interest, then what is the solution? Resolving such TENSIONS is considered non-trivial because we argue that "universal rules" cannot work in cases like this. Thus, it is a matter of *sophisticated prioritization* to come to the "right" solution. If, for example, a border control mission is on and a noise pollution is observed, then if the mission is just a routine surveillance, probably the noise-pollution should be avoided but if the mission is targeting trespassers, then the neighborhood silence may be sacrificed and for sure the residents would understand and appreciate such actions. For this reason, the AORTA drone mission modeling is to be complemented by a "prioritization scheme" which is not shown in the current section, for the sake of brevity.

What we are demonstrating is the OVERALL model (simplified to what Fig. 7 suggests) where all three perspectives are "super-imposed" (such that *tensions* could be straightforwardly identified) – this is presented in Table 1.

Table 1. Exemplifying the overall behavior, by applying AORTA.

role(*drone*, *{reportTrespassing(Trespasser, Location, Time), optimizeVision(Camera), optimizePower}*)
role(*officer,{apprehend(Trespasser)}*)
obj(*reportTrespassing(Trespasser, Location, Time)*, *{patrol(Location,Time), record (Trespasser, Location, Time)}*)
dep(*drone*, *officer*, *reportToOfficer)}*)
dep(*drone*, *camera*, *optimizeVision)}*)
cond(*camera*, *lightSensing*, *optimizeVision*, *clearWeather* ∧ *daytime*)
cond(*camera*, *darknessSensing*, *optimizeVision*, *cloudyWeather*)
cond(*drone*, *keepDistance*, *record*, *recordingFace*)
cond(*drone*, *reduceNoise*, *patrol*, ¬*alarm*)

5 Related Work

AORTA is based upon the *OperA* meta-model for *agent organizations*. *MOISE+* [26] is another meta-model for *agent organizations* that is well known in *AI* (*Artificial Intelligence*)-communities, and which is implemented in the agent-programming platform *JaCaMo* [2]. With regard to *agent-based AI systems*: "*Pepper*" is a *robot* working together with *humans* in a *socially acceptable way*. *Pepper* perceives its environment through *visual sensors* and can also receive *input through a touch screen*. *Pepper* is able to *make simple gestures*, *speak*, and *move* around on wheels. *Pepper*'s behavior is programmed using the agent-based language *GOAL* [9] in which an *agent* is specified in terms of a *mental state* with *personal info*. In contrast, we propose using *AORTA* in which an *agent maintains personal information and organizational information separately*. This relates to *user-driven context-aware systems* where for example *Body-Area Networks* are implemented to *capture vital signs* from the body of a patient from distance, such that this information is processed and communicated in an intelligent way [1] while *self-managing systems* were widely used in *home appliances* that adjust their operation with regard to some *goal* that concerns a unit, such as neighborhood [24]. This all concerns *values* that are *abstract* and *non-functional* and that is how they are considered to date [7], not aligning them adequately with the processes that concern the design of software. We believe that the current work is a step forward in approaching methodologically *context-aware systems*, in general and particularly – the three categories of context-aware systems addressed in this paper.

6 Conclusions

Furthering previous research that touches upon the enterprise-modeling-driven software specification, we have particularly addressed context-aware (software) systems in the current paper. We have identified and studied three categories of context-aware systems, featuring context-based system behavior adaptations that concern the optimization of system-internal processes, the maximization of the user-perceived effectiveness, and the consideration of relevant public values. Super-imposing those three system behavior perspectives and in synch with previous work (the SDBC approach), we have identified concepts and we have enriched them from an agent-technology perspective (in line with the AORTA framework), reflecting this in a meta-model that is considered a major contribution of the paper. Inspired by the meta-model, we have provided a partial refinement of the SDBC design guidelines, taking an agent technology perspective and focusing on the three above-mentioned system behavior perspectives, and we have partially illustrated this by means of a case example featuring land border security. As future work we plan to CONSOLIDATE our SDBC-AORTA-driven proposal into one dedicated design approach that is specific to context-aware systems.

Acknowledgements. This work is supported by: (i) the *TU Delft - Delft Pilot project*; (ii) *Technical University of Denmark* and the *PDC A/S project*. We would like to thank *Jeroen van den Hoven* for his support and guidance.

References

1. AWARENESS: Freeband AWARENESS Project (2008). http://www.freeband.nl
2. Boissier, O., Bordini, R.H., Hübner, J.F., Ricci, A., Santi, A.: Multi-agent oriented programming with JaCaMo. Sci. Comput. Program. **78**, 6 (2013)
3. Brun, Y., Di Marzo Serugendo, G., Gacek, C., Giese, H., Kienle, H., Litoiu, M., Müller, H., Pezzè, M., Shaw, M.: Engineering self-adaptive systems through feedback loops. In: Cheng, Betty H.C., de Lemos, R., Giese, H., Inverardi, P., Magee, J. (eds.) Software Engineering for Self-Adaptive Systems. LNCS, vol. 5525, pp. 48–70. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02161-9_3
4. Cockburn, A.: Writing Effective Use Cases. Addison-Wesley, Boston (2000)
5. Dey, A.K.: Understanding and using context. Pers. Ubiquit. Comput. **5**(1), 4–7 (2001)
6. Dignum, V.: A model for organizational interaction: based on agents, founded in logic. Ph. D. thesis, Utrecht University (2004)
7. Friedman, B., Hendry, D.G., Borning, A.: A survey of value sensitive design methods. In: A Survey of Value Sensitive Design Methods, Foundations and Trends, vol. 1, p. 76 (2017)
8. Google Dictionary: The website of Google Dictionary (2018). http://www.google.com
9. Hindriks, K.V.: Programming rational agents in GOAL. In: El Fallah Seghrouchni, A., Dix, J., Dastani, M., Bordini, R.H. (eds.) Multi-Agent Programming: Languages, pp. 119–157. Springer, Tools and Applications (2009). https://doi.org/10.1007/978-0-387-89299-3_4
10. Huebscher, M.C., McCann, J.A.: A survey of autonomic computing - degrees, models, and applications. ACM Comput. Surv. **40**(3) (2008). Article no. 7

11. Jensen, A.S., Dignum, V., Villadsen, J.: A framework for organization-aware agents. Auton. Agent. Multi-Agent Syst. **31**(3), 387–422 (2017)
12. Kephart, J.O., Chess, D.M.: The vision of autonomic computing. Computer **36**(1), 41–50 (2003)
13. Kruchten, P.: The Rational Unified Process: An Introduction. Addison-Wesley, Boston (2003)
14. Mahdavi-Hezavehi, S., Avgeriou, P., Weyns, D.: A classification framework of uncertainty in architecture-based self-adaptive systems with multiple quality requirements. In: Mistrik, I., Ali, N., Kazman, R., Grundy, J., Schmerl, B. (eds.) Managing Trade-offs in Adaptable Software Architectures, 1st edn. Elseiver Inc. (2016)
15. MDA: The OMG Model Driven Architecture (2018). http://www.omg.org/mda
16. Muehl, G., Werner, M., Jaeger, M.A., Herrmann, K., Parzyjegla, H.: On the definitions of self-managing and self-organizing systems. In: ITG/GI Symposium on Communication in Distributed Systems - 15, Bern, Switzerland, pp. 1–11 (2007)
17. Shishkov, B.: Enterprise Information Systems, A Modeling Approach. IICREST, Sofia (2017)
18. Shishkov, B., Janssen, M.: Enforcing context-awareness and privacy-by-design in the specification of information systems. In: Shishkov, B. (ed.) BMSD 2017. LNBIP, vol. 309, pp. 87–111. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78428-1_5
19. Shishkov, B., Mendling, J.: Business process variability and public values. In: Shishkov, B. (ed.) BMSD 2018. LNBIP, vol. 319, pp. 401–411. Springer, Cham (2018)
20. Shishkov, B., Janssen, M., Yin, Y.: Towards context-aware and privacy-sensitive systems. In: BMSD 2017, 7th International Symposium on Business Modeling and Software Design. SCITEPRESS (2017)
21. Shishkov, B., Mitrakos, D.: Towards context-aware border security control. In: 6th International Symposium on Business Modeling and Software Design, BMSD 2016. SCITEPRESS (2016)
22. Shishkov, B., van Sinderen, M.: From user context states to context-aware applications. In: Filipe, J., Cordeiro, J., Cardoso, J. (eds.) ICEIS 2007. LNBIP, vol. 12, pp. 225–239. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88710-2_18
23. Shishkov, B., Van Sinderen, M.J., Quartel, D.: SOA-driven business-software alignment. In: IEEE International Conference on e-Business Engineering, ICEBE 2006. IEEE (2006)
24. Shishkov, B., Warnier, M., Van Sinderen, M.: On the application of autonomic and context-aware computing to support home energy management. In: Proceedings of ICEIS 2010 - the 12th International Conference on Enterprise Information Systems, Funchal, PT, 8–12 June 2010. SCITEPRESS, Setúbal (2010)
25. UML: The Unified Modeling Language (2018). http://www.uml.org
26. van Riemsdijk, M.B., Hindriks, K., Jonker, C.: Programming organization-aware agents. In: Aldewereld, H., Dignum, V., Picard, G. (eds.) ESAW 2009. LNCS (LNAI), vol. 5881, pp. 98–112. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10203-5_9
27. Wooldridge, M., Jennings, N.R.: Intelligent agents: theory and practice. Knowl. Eng. Rev. **10**(2), 115–152 (1995)

Increasing the Visibility of Requirements Based on Combined Variability Management

Andreas Daniel Sinnhofer[1(✉)], Felix Jonathan Oppermann[2], Klaus Potzmader[2], Clemens Orthacker[2], Christian Steger[1], and Christian Kreiner[1]

[1] Institute of Technical Informatics, Graz University of Technology, Graz, Austria
a.sinnhofer@alumni.tugraz.at, steger@tugraz.at
[2] NXP Semiconductors, Gratkorn, Austria
{felix.oppermann,klaus.potzmader,clemens.orthacker}@nxp.com

Abstract. Nowadays, consumer-oriented industries like the Internet of Things, are highly affected by short product cycles and high pricing pressure. Agile and process-oriented organizations are known to perform better in such flexible environments. However, especially industries which are focused on delivering low cost systems are facing big challenges if the according Business Processes are not aligned with the capabilities of the product. Furthermore, non-functional requirements like safety and security are often not integrated in the early stages of a project, but later on added as a kind of extension, leading to a lower product maturity. With this work, we extend our framework for combined variability management in order to create an integrated view on the product variability from an organizational point, as well as from a technical view, including security requirements from the early development.

1 Introduction

We are living in an ever changing and interconnected world. The dawn of the Internet of Things (IoT) further increases the trend for organizations to deliver feature rich systems in high quantities and at low cost. Further, current markets demand an increasing number of new features with every product release, leading to very tight development cycles. Due to the pricing pressure and the shortened development time, methods have to be investigated which allow modular and highly configurable systems such that the products can be adapted and easily extended to the current market requirements. Additionally, the question about privacy and security is crucial, since IoT devices are usually deployed in unsupervised or untrusted environments. Nowadays, it is common practice to start the certification process of a product in a very late phase of the development, leading to huge costs in case of a negative attestation [30].

C. Kreiner—Deceased 5th of March, 2018.

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 203–220, 2018.
https://doi.org/10.1007/978-3-319-94214-8_13

Business Process (BP) oriented organizations are known to perform better regarding highly flexible demands of the market and fast production cycles [1–4]. These goals are achieved through the introduction of a management process in which business processes are modeled, analyzed and optimized in iterative improvement processes. During recent years, business process management is further coupled with a workflow management in order to monitor the correct execution of the process and to integrate responsibilities to the process models. Additionally, context aware business process modeling techniques were introduced in order to cope with fast changing requirements [5]. Such context aware systems rely on gaining flexibility by analyzing the context states of the environment and by mapping the according processes to their related software systems. One of the problems of this approach is that such software systems are often developed independently from each other, although they share a similar software architecture.

Software Product Lines (SPL) have proven to be essential for the development of flexible product architectures which can be adapted to the current requirements [6]. Thus, Software Product Line Engineering promises to create diverse, high-quality software products of a product family in short time and at low price. This is achieved through the use of a common architecture and reusable product features. The most critical phase during the design and the implementation of a product line is the identification of the variable parts and the common parts of the product family [6]. Consequently, a lot of effort is invested to identify the domain requirements of the final product portfolio. Equally important is the selection of the according features during the application engineering: It has to be guaranteed, that the customer requirements are fully met; and all unnecessary features need to be excluded in order to ensure low production costs of the final product. Since the identification of the domain requirements is usually carried out from developers, an integrated view on the organizational goals is often missing or incomplete. This means that a stable feature architecture is only achieved after a few iterations. Consequently, the efficiency of the product line is reduced since additional effort needs to be invested in order to create a product line adhering to the current requirements.

This work focuses on the development of a framework which aims to enforce a link between the variability of the business processes and the variability of the product platform. As such, we propose a combined variability modeling in which the requirements for the organization as well as for the development of the product platform are identified together in an integrated fashion. Consequently, developers and business process experts are getting insight into the different domain, leading to a more mature development process. Furthermore, a traceable link is established between the development artifacts and the security requirement in order to increase the visibility of these requirements within the whole organization. Additionally, this is a first step to agile security product certification schemes.

This work is based on our previous works in which we already defined systems for modeling variability of business process models (see [7]) as well as a framework for generating software configurations based on order processes (see [8–10]).

This work is structure in the following way: We present related work in Sect. 2. Section 3 summarizes basic concepts about business process modeling as well as software product line engineering, and the concept of Common Criteria security certifications that is used later in this paper. Section 4 summarizes our approach to link variable order process models to variable software architectures in an automated way. In Sect. 5 we describe how the proposed framework can be used to support an agile security certification according to Common Criteria. We will present an exemplary case study in Sect. 6 which describes how to apply the described concepts. And finally, we conclude this work in Sect. 7.

2 Related Work

Traditionally, business process modeling languages do not explicitly support the representation of families of process variants [11]. As a consequence, a lot of work can be found which tries to extend traditional process modeling languages with notations to build adaptable process models. As such, adaptable process models can be customized according to domain requirements by adding or removing fragments to the model and by explicitly transforming this model to dedicated process variants which can be executed in the field. This promises to increase the flexibility of business process oriented organizations with respect to highly flexible requirements of the market. Having such a variability modeling for business process models builds the foundation of this work. Thus, related work which is utilizing similar modeling concepts is presented in the following:

Derguech [12] presents a framework for the systematic reuse of process models. In contrast to this work, it captures the variability of the process model at the business goal level and describes how to integrate new goals/sub-goals into the existing data structure. The variability of the process is not addressed in his work.

Gimenes et al. [13] presents a feature based approach to support e-contract negotiation based on web-services (WS). A meta-model for WS-contract representation is given and a way is shown how to integrate the variability of these contracts into the business processes to enable process automation. It does not address the variability of the process itself but enables the ability to reuse business processes for different e-contract negotiations.

While our used framework to model process variability reduces the overall process complexity by splitting up the process into layers with increasing details, the PROVOP project [14–16] focuses on the concept, that variants are derived from a basic process definition through well-defined change operations (ranging from the deletion, addition, moving of model elements or the adaptation of an element attribute). In fact, the basic process expresses all possible variants at once, leading to a big process model. Their approach could be beneficial considering that cross functional requirements can be located in a single process description, but having one huge process is also contra productive (e.g. the exchange of parts of the process is difficult).

The work of Gottschalk et al. [17] presents an approach for the automated configuration of workflow models within a workflow modelling language. The term workflow model is used for the specification of a business process which enables the execution in an enterprise and workflow management system. The approach focuses on the activation or deactivation of actions and thus is comparable to the PROVOP project for the workflow model domain.

La Rosa et al. [18] extends the configurable process modelling notation developed from [17] with notions of roles and objects providing a way to address not only the variability of the control-flow of a workflow model but also of the related resources and responsibilities.

The Common Variability Language (CVL [19]) is a language for specifying and resolving variability independent from the domain of the application. It facilitates the specification and resolution of variability over any instance of any language defined using a MOF-based meta-model. A CVL based variability modeling and a BPM model with an appropriate model transformation could lead to similar results as presented in this paper.

The work of Zhao and Zou [20] shows a framework for the generation of software modules based on business processes. They use clustering algorithms to analyze dependencies among data and tasks, captured in business processes. Further, they group the strongly dependent tasks and data into a software component.

3 Background

This section summarizes the basic concepts about Software Product Line Engineering, and Business Process Modeling that build the foundation of the introduced concepts. Additionally, the concept of incremental Common Criteria certification schemes is summarized which is used in this paper.

3.1 Software Product Line Engineering

Software Product Line Engineering (SPLE) applies the concept of product lines to software products. As a consequence, SPLE promises to create diverse, high-quality software products of a product family in short time and at low cost [6]. Instead of writing software for every individual system, a Software Product Line (SPL) is used to automatically generate software products by combining the required domain artifacts. The principal concept can be split into two main phases: the Domain Engineering and the Application Engineering [6,21].

During the Domain Engineering, the domain artifacts, the variabilities and the commonalities of the according domain are identified and implemented. In the Application Engineering phase, the final products are created by combining the domain artifacts which were implemented in the previous phase. In contrast to the domain engineering, the application engineering is mainly focused on reusing domain artifacts. Based on the current requirements of the product, specific domain artifacts are chosen and assembled to the final product. Ideally, the application engineering makes use of software generators to automatically derive product variants without the need of implementing any new logic. This enables a rapid creation of high-quality products within a defined product family. The amount of reused domain artifacts greatly depends on the application requirements. Hence, a major concern of a SPL is the detection of deltas between the application requirements and the available capabilities.

3.2 Business Process Modeling

Business Processes (BP) are a specific sequence of activities or (sub-) processes which are executed in a dedicated sequence to produce output with value to the customer [2,9]. In this work, we use the concept defined by [22] to model BPs: BPs are modeled in different layers, where the top level (macroscopic level) is a highly abstract description of the overall process and the lower-levels (microscopic level) are more detailed descriptions of the sub-processes. A reasonable level of detail is reached, if the process description on the lowest levels can be used as work-instructions for the responsible employees. This leads to the fact that the higher levels of the process description are usually independent of the production facility and the supply chains; while the lower levels are highly dependent on the production facility and its capabilities. As a consequence, the macroscopic level is more stable with respect to changes and can be reused in different contexts and production environments. The microscopic levels need to be updated in order to reuse them in different contexts.

Variability of such process structures can be modeled through a variable process structure (i.e. by adding/removing activities in a process) or by replacing sub-process refinements with different sub-processes.

Domain specific modeling languages are usually used to model all the activities, resources and responsibilities within a Business Process. In the scope of this work, the Business Process Model and Notation (BPMN, [23]) is used to model processes, but the general concept of this work is not limited to this notation. The key concepts which are used in this work, are [9,23,24]: Events that occur during the execution of a process and affect the flow of a process; Activities which are a specific amount of work that has to be performed by an organization, that can either be atomic or non-atomic (i.e. a sub-process); Gateways that are used to control how the process flows through difference sequences of activities; Data objects that represent the information flow through the process; And Pool and Lanes to highlight responsibilities for activities.

Key Concept Traceability. Traceability are key notions of all process assessment and improvement models [25]: Foundation of every improvement process is the collection of data with respect to the product quality, customer satisfaction, and process maturity. Based on the collected data – and a high degree of traceability – it is possible to identify the weak points of a system and to systematically tackle them in improvement projects. Furthermore, functional safety standards like the ISO/IEC 15504 require a (bidirectional) traceability to show how safety requirements are defined, implemented and tested during the development of a product. Due to this strong link, the impact of changes on the overall system can be distinguished. Additionally, Security certifications usually require evidence that shows how confidential (customer) data is processed and handled during the production process. As stated by Raschke et al. [26], traceability of security requirements is a key concept for incremental security evaluations, allowing the identification of the impact of change to the security target.

3.3 Common Criteria Certification

In the domain of information security, the Common Criteria[1] is widely used to evaluate the security measures of a product. The CC defines a common set of requirements that need to be implemented by the security functionality of the product [27]. Consequently, the evaluation process creates a level of confidence to the security functionality which is implemented in hardware, software, or both. For higher security levels, the CC evaluation also considers the maturity of the development processes, the production processes, as well as the used tools and toolchains. Consequently, the CC evaluates not only the implemented security features, but also the overall maturity of the whole organization. In the context of this publication, it is of specific interest, that the CC allows modular and incremental evaluation processes which enable beneficial development strategies which we will highlight later in this paper. In order to understand the concepts, we will briefly summarize the Delta Evaluation in the following:

The delta evaluation [28] is a CC evaluation process used to maximize the reuse of previously compiled evidences. To do so, the according product and supporting documentation needs to be shared with the evaluation facility as well as a descirption of the applied evaluation methodology and a delta report highlighting the impact of changes to the Security Target. Thus, traceability of requirements is essential to use this evaluation process. Providing the required data means that only those parts of a system need to be re-evaluated, that are affected by a change. One drawback of this approach is that technical descriptions of the evaluation methodology are usually proprietary to an evaluation facility. This means that using a different evaluation facility for a re-evaluation is in practice often difficult.

4 Combining Process Variability and Software Variability

It is common practice for organizations to maintain multiple variants of business processes which are based on a common template [11]. New process variants are created based on a copy and clone strategy of such templates. Consequently,

[1] https://www.commoncriteriaportal.org.

maintaining these variants can be very time consuming since every single variant has to be manually updated by process designers. Furthermore, copy and clone strategies often lead to a fragmented and outdated documentation. To solve these issues, we proposed to use a framework which allows to create process variants based on a process model and a defined set of transformation actions [7–9] in order to derive variability models of Software Product Lines.

The general framework for creating business process models can be split into four phases: A process modeling phase during which process designers are responsible to design process templates, including also additional meta information like documentation artifacts or responsibilities; In the second phase, a domain expert imports these process templates into a Software Product Line tool and translates it into a feature model; the definition of the feature model is done with the help of the according process designer in order to identify the variable and static parts of a process; Variability of the process is modeled by either adding or removing specific tasks in a process sequence, or by exchanging sub-processes with different variants; In the third phase, process variants are automatically derived from the feature model based on the current requirements of the organization; As such, a process variant is automatically derived from a core process model without any manual step, meaning that changes to core parts of the model can be automatically propagated to all derived process variants; And the last phase deals with the maintenance and evolution of the process model based on the data that was collected during the production of the product. The last phase usually results in process improvement processes during which the process model, or the rule-set of the feature model are adapted and optimized.

Having a framework to model the variability of business processes is a first step towards combined variability management. In order to reach the goal of a combined variability modeling, an integrated view is necessary in which the variability of the business process model is reflected in the feature model of the product. More specific, the configuration possibilities of the order processes need to be reflected by the capabilities of the customization product line. In order to automate this process, we proposed to apply mapping rules to translate the variability of the order process model to a feature model of the final product (see [9, 10]) and vice versa. As a brief summary, process data is converted into required configuration settings of specific features, and features are derived based on the process structure: Atomic activities are converted to features; Gateways are used to model mandatory/optional features, and conditions between them; And Pool and lanes are used to identify the origin of the necessary configuration data. Non-atomic features are processed recursively until the microscopic level is reached, leading to a hierarchical representation of the features.

After identifying the requirements of the product line, web-based order-entry forms can be automatically generated based on the order-process model and some defined translation rules (see [9]). As such, the type of input data needs to be annotated by the process designers to the according process model. The order-entry forms need to be completed by internal and external customers during the placement of an order. After the order forms are completed and submitted, the provided data is collected and used to automatically configure the customization product line of the final product. As such, the overall production costs can be significantly reduced. Additionally, through the use of automated flows, it is possible to create a traceable link from the order placement to the actual configuration data and production steps. The overall concept of combining the variability modeling is highlighted in Fig. 1.

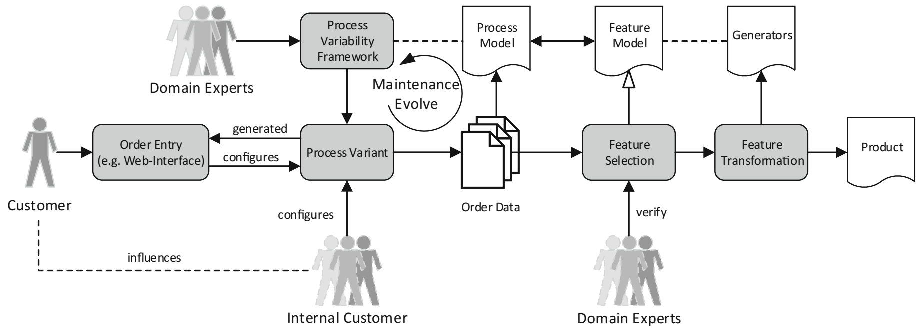


Fig. 1. Overall framework for combining the process variability with the product variability [10]

Our today's business environment is focused on creating sustainable value by increasing the revenue of business drivers. The identification of such business drivers – or the identification of the drivers which are able to destroy value – is an essential step for an organization [29]. Otherwise, staying competitive or even survive on a flexible market is not possible. The combination of business variability and software variability is a promising way to improve the identification of such drivers: Linking domain artifacts and their according implementation/maintenance efforts to the respective business processes can reveal improvement opportunities that need to be tackled in process improvement processes. Further, having a combined view of the requirements helps to increase the overall efficiency of the product line since the visibility of requirements are increased throughout the whole organization: Managers and process designer get a better understanding of the impact to development teams; and development teams are getting a better understanding of the overall business goals.

5 Supporting an Incremental Common Criteria Security Certification

Nowadays, agile product development techniques are used to provide a rapid and steady progression of incremental improvements [30]. Consequently, the final product architecture promises to be modular and easy to extend, which

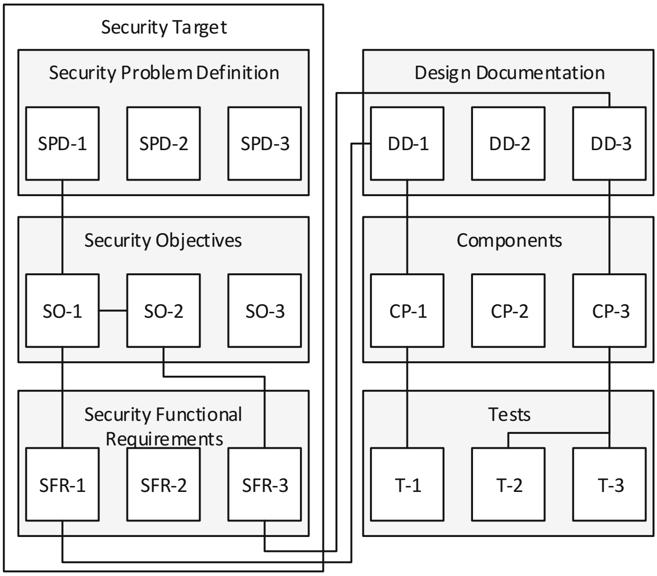


Fig. 2. An explanatory Security Model to highlight the relationships between the security requirements and their related components and tests. Dependencies between the different security requirements are not highlighted.

will lead to a faster time to market and lower development costs. However, a problem arises with this approach in case a security certification is required. Traditionally, the certification process of a product is issued in a very late phase of the development, leading to high costs in case the certification body gives a negative attestation. Another problem is the long period of time an evaluation process can take, even if the certification body gives a positive attestation. This can lead to a delayed release if a certificate is a condition for the sale of a product, or a big gap between the date of the release and the date a certificate is issued. Either way, both situations can potentially result in a loss of customers if a competitor is already selling a certified product [30].

In case of Common Criteria (CC) certifications, this effort can be reduced if an incremental product certification is used [26]. As already stated in Sect. 3.3 it is necessary to have a high degree of traceability of the security requirements, to the actual software/hardware components, and their according tests in order to be able to perform incremental product certification processes. It is important to understand, that modules can interact with each other and hence, not only the directly changed modules need to be reconsidered, but also all dependent modules need to be addressed during an iteration. Identifying the impact set can be achieved by applying the change impact analysis proposed by Raschke et al. [26], which also supports an automatic generation of the Impact Analysis Report which is required for the CC certification. The change detection analysis is based on the Security Model, which describes the relationship of the evidences, based on the CC security target.

An example is highlighted in Fig. 2. As highlighted in the figure, components CP-1 and CP-2 are directly affected by changes to the first Security Problem Definition. Usually more interestingly is the other way around: Changes to components CP-1 or CP-2 will directly affect a security goal. Further, there could be additional dependencies between the different security requirements which would need to be considered as well in case of changes.

Besides considering the security requirements and their according implementations, CC security certifications usually also require evidence that confidential (customer) specific data is processed correctly during the production of the product. This means, that the data is not leaked during the production process, nor confused with any other data. Using the framework proposed in Sect. 4 helps in order to generate evidence on how data was processed and handled starting from the product order until shipment to the customer.

6 Case Study

In this Section, we will take a look at an exemplary case study to highlight the concepts that were described within this work: A company is selling multi-application Smart Cards which can be ordered in different variants. All variants are certified according to Common Criteria. For simplicity of this example, we can assume that the following variants are offered by the company:

– **Variant A:** Is a classical Smart Card used for payment (e.g. an ATM card, or a credit card, or a combination of both), following the EMV standard, supporting different credit card institutions. Customers (e.g. Banks, or credit card companies) are buying these variants with different payment Applications installed. Before such a card is sent out to the end-customer, all unnecessary payment applications are disabled.
– **Variant B:** Is a combination of Variant A, with the possibility to use it also for public transport. That means that an end-customer can for example enable the card to be used as transport ticket in Sweden, which means that the price of the ticket is automatically paid by presenting the card to a reader, without the need of buying a ticket at a ticket machine.

6.1 Exemplary Security Model

In order to get an idea how the security model is structured, an exemplary and simplified version is given. **Security Problem Definition (SPD):** Smart Cards which are certified according to Common Criteria are usually compliant to a CC Protection Profile[2] which defines basic requirements that need to be met by the Hardware and Software of the product. The core assets for such products are the protection of the confidentiality and integrity of the user data, as well as the requirement for sufficient authentication for the activation/execution of specific functionality of the device (e.g. providing a PIN before being able to get

[2] https://www.commoncriteriaportal.org/pps/.

money from an ATM). The main threats to these assets are the manipulation of user data including also parts of the operating system (e.g. introduction of malicious functions) and the disclosure of user data. Table 1 highlights the **Security Objectives (SO)** that can be derived based on the SPD. In a real scenario, also hardware related Security Objectives would need to be considered.

Table 1. Security Objectives of the exemplary case study, derived from the Security Problem Definition.

Security objective	Description
SO.1	The Smart Card must protect the confidentiality of user data and credentials
SO.2	The Smart Card must protect the integrity of user data and credentials
SO.3	The Smart Card must provide means to limit the access to specific functionality of the card to authorized users only
SO.4	The card must provide secure communication interfaces which provide confidentiality and integrity protection of the sent data
SO.5	The protection of core assets during the production of the product has to be ensured, to ensure a secure configuration of initial user data

The way how to securely store the initial user data in a secure manner is an own research topic and is out of scope for this publication. For simplicity, we can assume that a script based approach is used for the configuration of the initial user data (SO-5), which relies on a standard API of the card. Confidential data within the scripts is encrypted such that the data is not leaked to an operator.

The SOs are further refined in the exemplary **Security Functional Requirements (SFRs)** which are highlighted in Table 2. Note that the algorithms described in this section are only examples and may be different depending on the domain of the product and the actual security requirements.

During the development of the product, all these Security Requirements are linked to the according component(s) of the implementation (like cryptographic co-processors, or functions of the operating system), their documentation – including threats, attack vectors and according countermeasures – and the according component and integration tests. In order to be able to do an incremental and agile security certification, the maturity degree of the above requirements need to be continuously tracked by highlighting the completed implementation artifacts and passed test cases. New requirements can be incorporated by analyzing the impact of change and by providing the related report to the certification facility. As such, the security certification can already start in parallel to the actual implementation of the system, leading to a faster time to market of the product.

Table 2. Security Functional Requirements of the exemplary case study, derived from the Security Objectives.

SFR	Parent SO	Description
SFR.1	SO.1 SO.4 SO.5	The card shall perform encryption and decryption in accordance with the specified cryptographic algorithm Advanced Encryption Standard (AES) using key sizes of 128, 192, and 256 Bit
SFR.2	SO.2 SO.5	The card shall perform the calculation of secure hashes with a specified hash algorithm in accordance to the following selection: [SHA-256, SHA-512]
SFR.3	SO.3	The card shall perform the calculation of digital signatures in accordance with a specified digital signature algorithm supporting the following public key cryptography algorithms: [RSA, ECC]
SFR.4	SO.4	The card shall support the computation of Message Authentication Codes (MAC) in accordance with the specified cryptographic algorithm defined in SFR.1

The presented SO are basic objectives that need to be addressed during the whole lifecycle of the product. On top of that, additional requirements need to be derived in order to provide a secure way how customers can share the confidential user data that needs to be available during the production of the card. In order to understand these requirements in more detail, we will take a look at the order processes and the necessary variability of the production process in order to cope with different context specific requirements.

6.2 Variability of the Configuration Process

For illustration purposes, Fig. 3 shows a simplified configuration process, showing the customization of the described product variants. Each payment or public transport application that shall be part of the final product can be selected by the customer and will be installed to the card and pre-configured with initial customer related data. This initial data contains cryptographic keys via which e.g. a secure communication channel can be established to securely process transactions in the field. In a real scenario, each selected application may require different data that needs to be configured. For simplicity of the example, it can be assumed that only one file is shared, which contains a single cryptographic key.

Anyways, the customization processes need to be modeled in accordance to SO.4 and SO.5, meaning that the secure channel implementation is based on the security requirements SFR.1, SFR.2, and SFR.4 to ensure confidentiality and integrity of the user data. This direct dependency between the configuration process and the security model of the product can be modeled in our proposed framework in order to increase the visibility of the security requirements in the organization. Further, the impact of change can be analyzed by using this

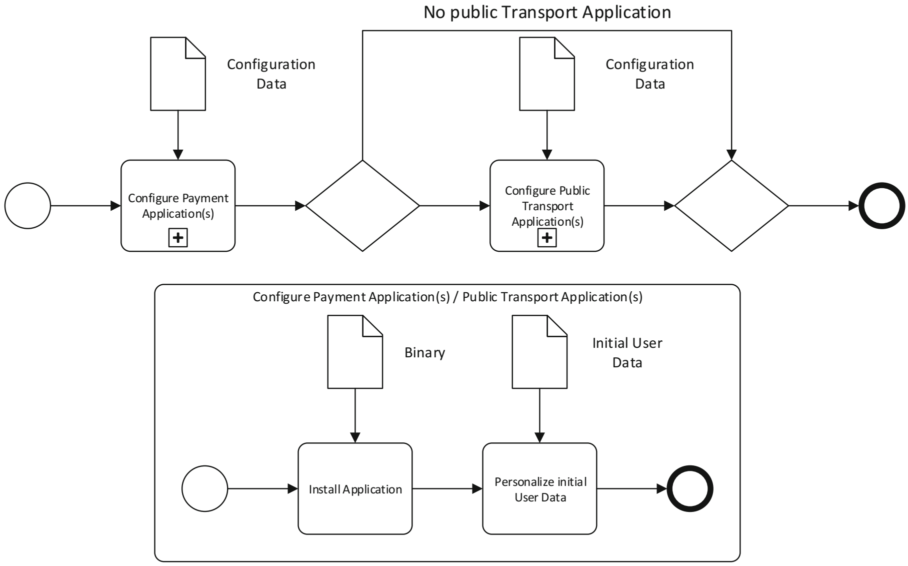


Fig. 3. Exemplary configuration process, showing the customization of the described two product variants (Variant A, Variant B).

traceable link: For example, if a customer requests the use of a specific secure channel protocol, SFR.1 may need to be adapted in order to include also other encryption algorithms. Changing the supported encryption algorithm has an impact to the Security Objectives and thus, it needs to be investigated if the Security Problem Definition still holds. In case an internal security audit has shown that the change is acceptable, a CC delta evaluation can be issued in order to use the most efficient evaluation process.

Besides the consideration of how to store and send the initial user data on the card during the production, it is equally important to consider how customers are able to transfer this data to the company. In principal, there are many ways on how to share or generate the initial cryptographic keys. For illustration, a few possibilities are given below:

- **Option A:** Assuming that all cards are sharing the same key configuration, a customer may share this key in a secure manner (e.g. pgp encrypted) with the vendor of the card.
- **Option B:** In practice, option A is usually not meaningful since breaking the keys of a single card would break the security of all ordered cards. Due to this, a customer may request to use dedicated key hierarchies, in which each card gets an own key. These key hierarchies are derived from Master Keys using specified key derivation functions. This has the advantage that, a customer still only needs to share one Master Key in a secure manner with the vendor, but each card has a unique key.

– **Option C:** The Smart Card vendor is generating the Master Key on behalf of the customer in order to derive the card specific keys. As such, only the Master Key needs to be exchanged in a secure manner with the customer.

Technically, all three options can be supported by the vendor of the card and may be modeled by business process designers. However, depending on the security requirements defined in the security model, Option A may violate the Security Problem Definition of the product as defined in Sect. 6.1. As such, either the assurance of the final product is lower, or this method is limited for first demonstration cards which may only be used for test purposes (e.g. for testing the reader infrastructure).

The link to the security requirements is an important information that needs to be explicitly highlighted in the according business process model such that it is clear that a security certification may not be available for certain variants of a product. Based on the process model, a web-based order form can be generated by applying the mapping rules proposed in our framework (cf. [10]) in an automated fashion. The resulting order form will have the following features: Since the configuration of Payment Applications is mandatory, a file upload button is generated via which the according binary can be uploaded, as well as a file upload button for the initial user data. Additionally, the same is generated for the configuration of Public Transport Applications, but hidden or visible based on a check box that needs to be selected by the customer.

In this case, the default behavior of our framework would result in an order form which may be usable, but does not fully fit to the requirements of the domain. As such, the model needs to be manually annotated such that the mapping rules generate a more meaningful order form. Thus, the binary of the Payment Application needs to be changed to a selection based on a list of supported applications. Additionally, the key exchange processes (Option A-C) need to be linked to the initial User Data, which means that the customer can decide on how to exchange the initial User Data. Figure 4 shows an exemplary web-form: The Master Key is provided by the customer pgp encrypted in order to configure the selected 'Example Application 1'.

As described earlier, the process for loading the initial data is based on a script approach, where a script is executed in order to configure a single card with the initial data. By applying our proposed framework from [10], the according feature model of the product line can be automatically generated based on the annotated process model, leading to two implementation artifacts: A generic script that is able to install an application on the card based on e.g. ISO 7816-4 commands; and a generic script via which specific data can be securely sent to the card using e.g. a secure channel as defined by the Global Platform specification.

After pressing the submit button in the web-form, the provided configuration data is packed together into a zip folder, containing an XML describing the actual order and additionally the resources that were uploaded by the customer, or were implicitly selected via the according selection dialog. This data can be directly mapped to the feature model of the configuration process in order to automatically derive the script that needs to be executed during the production

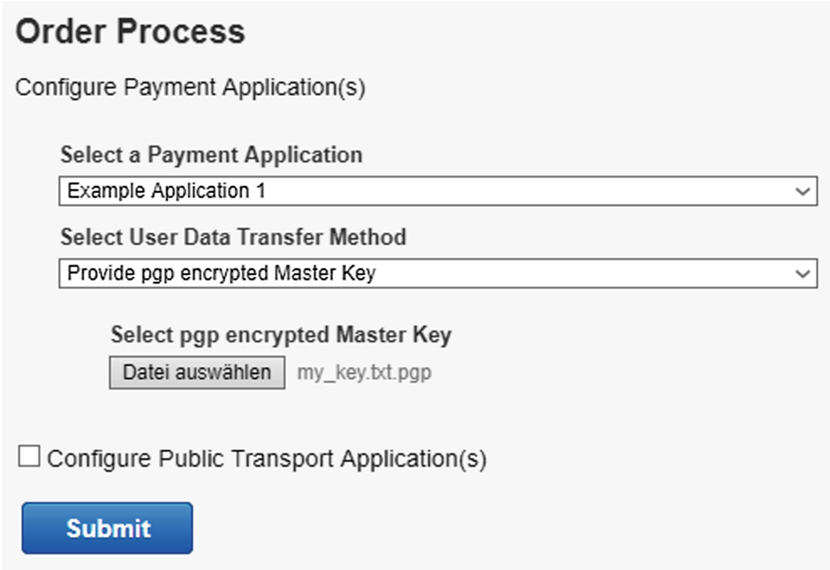


Fig. 4. Exemplary web-based order form that is generated based on the annotated order process model.

process. This process is in principal fully automated, but due to the reason that such products may contain highly complex configurations, it is reasonable to include a manual verification step via which possible miss-configurations can be identified and clarified with the customer.

Since configuration safety is an important topic for security certifications, the provided or generated key data can be tagged with a product identifier which is checked during the configuration process. This means that only user data can be loaded to the device, which was actually intended for this product and was not confused with any other data. This check can be done automatically which means that human mistakes in the production can be minimized.

To summarize the shown example: Using a combined variability management of business process models and software product line engineering, enables an automatic customization of the product line based on the data that was provided by internal or external customers via an order-entry form. This order-entry form can be generated from the process model by applying certain mapping rules which can be tailored for the specific use case by adding additional information to the process model. Due to the automatic generation of the order form and the actual execution of the customization process, it is possible to generate a report which describes how data was provided and processed during the manufacturing process. Additionally, this traceable link allows highlighting which features of a product are ordered by customers and how much revenue is generated per feature. As such, the business drivers – or equally important the drivers that destroy revenue – can be identified and optimized in regular improvement processes. Further, it was shown how security requirements are linked to the variability models in order to increase the general awareness of these requirements in the whole organization which is also a key concept for modular and incremental security certifications.

7 Conclusion

Today's industry is defined by fast changing requirements regarding functional requirements. But also non-functional requirements like safety and security are gaining more momentum in the recent years. One of the biggest challenges organizations are currently facing is how to integrate such non-functional requirements in an integrated product development process. With this work, we proposed to use combined variability management in order to create an integrated view of the organizational goals from a technical perspective, as well as from a business view. Further, traceable links are establish via which the impact of changes can be investigated leading to a better understanding on what parts of a system are affected by changes prior of costly investigations of multiple different development teams. Additionally, we highlighted how to integrate security requirements from the early product development in order to support agile and incremental security certifications according to Common Criteria. The main benefit of having an agile certification process is that the security related requirements are integrated from the beginning, leading to a better communication and awareness of these requirements in the whole development team. Through the use of traceability concepts, it is possible to automatically generate delta reports in order to efficiently address re-certifications.

In a future work, we will investigate how to integrate other non-functional requirements – like functional safety – into this framework in order to allow a more mature product development. Further, we want to extend the framework to not only consider the variability of order processes, but also from other processes in order to improve the process awareness of development teams. Additionally, we think that development teams can improve their way of working significantly if the corresponding business goal is well understood and linked to the development activities. And on the other hand, managers can get a better understanding of the impact of changes and the overall progress by monitoring the completeness of development artifacts.

Acknowledgements. The project is funded by the Austrian Research Promotion Agency (FFG). We want to gratefully thank pure::systems for their support and especially Danilo Beuche.

References

1. McCormack, K.P., Johnson, W.C.: Business Process Orientation: Gaining the E-Business Competitive Advantage. Saint Lucie Press, Boca Raton (2000)
2. Hammer, M., Champy, J.: Reengineering the Corporation - A Manifesto For Business Revolution. Harper Business, New York (1993)

3. Valença, G., Alves, C., Alves, V., Niu, N.: A systematic mapping study on business process variability. Int. J. Comput. Sci. Inf. Technol. (IJCSIT) (2013)
4. Willaert, P., Van den Bergh, J., Willems, J., Deschoolmeester, D.: The process-oriented organisation: a holistic view developing a framework for business process orientation maturity. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 1–15. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75183-0_1
5. Saidani, O., Nurcan, S.: Towards context aware business process modelling. In: 8th Workshop on Business Process Modeling, Development, and Support (BPMDS 2007), CAiSE, vol. 7, p. 1 (2007)
6. Pohl, K., Böckle, G., van der Linden, F.: Software Product Line Engineering: Foundations, Principles and Techniques. Springer, New York (2005). https://doi.org/10.1007/3-540-28901-1
7. Sinnhofer, A.D., Pühringer, P., Kreiner, C.: varBPM—a product line for creating business process model variants. In: Proceedings of the Fifth International Symposium on Business Modeling and Software Design - Volume 1: BMSD 2015 (2015) 184–191
8. Sinnhofer, A.D., Pühringer, P., Potzmader, K., Orthacker, C., Steger, C., Kreiner, C.: A framework for process driven software configuration. In: Proceedings of the Sixth International Symposium on Business Modeling and Software Design - Volume 1: BMSD 2016, pp. 196–203 (2016)
9. Sinnhofer, A.D., Pühringer, P., Potzmader, K., Orthacker, C., Steger, C., Kreiner, C.: Software configuration based on order processes. In: Shishkov, B. (ed.) BMSD 2016. LNBIP, vol. 275, pp. 200–220. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57222-2_10
10. Sinnhofer, A.D., Höller, A., Pühringer, P., Potzmader, K., Orthacker, C., Steger, C., Kreiner, C.: Combined variability management of business processes and software architectures. In: Proceedings of the Seventh International Symposium on Business Modeling and Software Design - Volume 1: BMSD, pp. 36–45. INSTICC, SciTePress (2017)
11. Rosa, M.L., Van Der Aalst, W.M., Dumas, M., Milani, F.P.: Business process variability modeling: a survey. ACM Comput. Surv. **50**(1), 2:1–2:45 (2017)
12. Derguech, W.: Towards a framework for business process models reuse. In: The CAiSE Doctoral Consortium (2010)
13. Gimenes, I., Fantinato, M., Toledo, M.: A product line for business process management. In: International Software Product Line Conference, pp. 265–274 (2008)
14. Hallerbach, A., Bauer, T., Reichert, M.: Guaranteeing soundness of configurable process variants in Provop. In: IEEE Conference on Commerce and Enterprise Computing, CEC 2009, pp. 98–105. IEEE (2009)
15. Hallerbach, A., Bauer, T., Reichert, M.: Issues in modeling process variants with Provop. In: Ardagna, D., Mecella, M., Yang, J. (eds.) BPM 2008. LNBIP, vol. 17, pp. 56–67. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00328-8_6
16. Reichert, M., Hallerbach, A., Bauer, T.: Lifecycle management of business process variants. In: vom Brocke, J., Rosemann, M. (eds.) Handbook on Business Process Management 1. IHIS, pp. 251–278. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-642-45100-3_11
17. Gottschalk, F., van der Aalst, W.M.P., Jansen-Vullers, M.H., La Rosa, M.: Configurable workflow models. Int. J. Cooper. Inf. Syst. **17**, 177–221 (2007)

18. La Rosa, M., Dumas, M., ter Hofstede, A.H.M., Mendling, J., Gottschalk, F.: Beyond control-flow: extending business process configuration to roles and objects. In: Li, Q., Spaccapietra, S., Yu, E., Olivé, A. (eds.) ER 2008. LNCS, vol. 5231, pp. 199–215. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87877-3_16
19. Haugen, O., Wasowski, A., Czarnecki, K.: CVL: common variability language. In: Proceedings of the 17th International Software Product Line Conference, SPLC 2013 (2013)
20. Zhao, X., Zou, Y.: A business process-driven approach for generating software modules. Softw.: Pract. Exp. **41**(10), 1049–1071 (2011)
21. Weiss, D.M., Lai, C.T.R.: Software Product-line Engineering: A Family-based Software Development Process. Addison-Wesley Longman Publishing Co., Inc., Boston (1999)
22. Österle, H.: Business Engineering - Prozeß- und Systementwicklung. Springer, Heidelberg (1995). https://doi.org/10.1007/978-3-662-06188-6
23. Object Management Group: Business Process Model and Notation (BPMN). version 2(0), pp. 1–538 (2011). http://www.omg.org/spec/BPMN/2.0/
24. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-33143-5
25. Bíró, M.: Functional safety, traceability, and open services. In: Madeyski, L., Ochodek, M. (eds.) Software Engineering from Research and Practice Perspective, pp. 73–82. Wyd. Nakom, Poznan (2014)
26. Raschke, W., Zilli, M., Baumgartner, P., Loinig, J., Steger, C., Kreiner, C.: Supporting evolving security models for an agile security evaluation (2014)
27. Sinnhofer, A.D., Raschke, W., Steger, C., Kreiner, C.: Patterns for common criteria certification. In: Proceedings of the 20th European Conference on Pattern Languages of Programs, EuroPLoP 2015, pp. 33:1–33:15. ACM (2015)
28. Criteria, C.: Common criteria information statement - reuse of evaluation results and evidence (2002)
29. Strnadl, C.F.: Aligning business and it: the process-driven architecture model. Inf. Syst. Manag. **23**(4), 67–77 (2006)
30. Sinnhofer, A.D., Raschke, W., Steger, C., Kreiner, C.: Evaluation paradigm selection according to common criteria for an incremental product development. In: International Workshop on MILS: Architecture and Assurance for Secure Systems, vol. 1, pp. 1–5 (2015). http://mils-workshop-2015.euromils.eu/

Situational Method Engineering for Constructing Internet of Things Development Methods

Görkem Giray[1(✉)] and Bedir Tekinerdogan[2]

[1] İzmir, Turkey
gorkemgiray@gmail.com
[2] Information Technology, Wageningen University, Wageningen, The Netherlands
bedir.tekinerdogan@wur.nl

Abstract. Developing Internet of Things (IoT) systems is not trivial and needs to be performed systematically to derive an IoT system that meets the required functional and non-functional concerns. Since IoT is applied to different heterogeneous domains usually a one-size-fits-all method is less feasible. For some cases a lightweight method with a few method artefacts are sufficient while in other cases a detailed set of method artefacts over the whole lifecycle might be required. So far, a few IoT system development methods (SDM) have been provided that include the steps necessary for guiding the development of IoT systems but these do not explicitly consider the situational needs for the required IoT method. In this paper we propose a situational method engineering (SME) approach for developing a method base that includes a broad set of method fragments which can be reused to develop customized methods. We illustrate the development of the method base using the existing IoT methods that have been proposed in the literature so far. Further we show how the method base can be used to develop methods for two different cases.

Keywords: Internet of Things · IoT system development method
Situational method engineering

1 Introduction

The internet of things (IoT) is now being applied in more and more domains including smart home/city, healthcare, transportation, retail, agriculture, etc. [53]. Such different domains call for IoT systems fulfilling different functional and non-functional requirements. In addition, IoT systems may involve development, manufacturing, acquisition, and/or integration of various software, hardware, and communication components, which need to be explicitly considered during a project possibly by different stakeholders. To develop a system, it is important to apply a systematic development method (SDM) that includes the steps for guiding the development of the system in order to achieve the targeted functional and non-functional concerns of the various stakeholders. For developing IoT systems a few IoT SDMs have been indeed developed with various degrees of detail and focus. In general it seems that it is tacitly

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 221–239, 2018.
https://doi.org/10.1007/978-3-319-94214-8_14

assumed that these methods are suitable in all circumstances. Yet different application domains and different context usually have different needs and requirements for an IoT SDM and as such a one-size-fits-all approach would not be feasible [1, 54, 55]. Situational factors define the context in which a project will operate and these factors should be considered when constructing a method. Hence, rather than using an existing general IoT SDM, an alternative would be to allow the development of an IoT SDM that directly considers the situational factors and the context of the project at hand. Figure 1 illustrates a high-level view of such an approach in which a method is constructed based on situational factors. Hereby, a method is constructed based on situational factors and project requirements. The constructed method is then applied in a specific project and the feedback are used for revising the method.

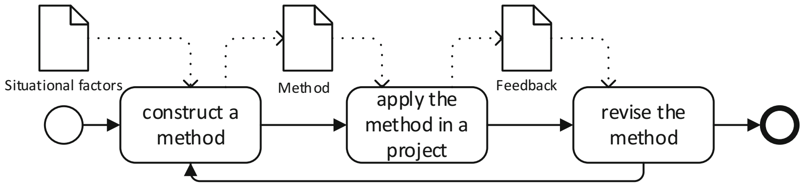


Fig. 1. Conceptual model for situational method engineering

Elaborating on the current practical needs of many IoT projects we propose a situational method engineering (SME) approach [2] for developing a method base that includes method fragments for developing IoT systems. The approach that we propose consists of two key activities. First of all, we define the process for developing the method base including reusable method fragments. For developing the method base we have used the identified IoT SDMs in the literature. Second, we describe the process for reusing the method fragments from the method base to develop a method. We illustrate our approach for the development of two situationally different IoT cases.

The remainder of the paper is organized as follows. Section 2 briefly describes the situational method engineering for IoT. After a short background it presents the process for developing a method base using the IoT SDMs in the literature, and the process for reusing the method fragments of the method base to develop an IoT SDM. In Sect. 3, we describe the application of the processes as defined in Sect. 2, that is, method base construction, and the process for the development of a method using the method base. We present a case study, in which we construct a method base for the IoT domain using the existing IoT SDMs. Afterwards, in Sect. 4, we construct two sample IoT SDMs for precision farming using the method base we have formed. In Sect. 5, we discuss the practical implications of the approach both from an industrial and academic perspective. Section 6 includes the related work and finally Sect. 7 concludes the paper.

2 Situational Method Engineering

Method engineering (ME) is the engineering discipline to design, construct, and adapt methods, techniques, and tools for systems development [3]. Situational method engineering (SME) is a subtype of ME and encompasses all aspects of creating a development method for a specific situation [4]. SME aims to propose a solution to the problem of the selection of a proper development method for an organization and/or its projects [2]. Engineering a method in SME usually encompasses selecting method parts from a *method base* and constructing a specific method for a specific organization and/or project by taking into account some situational factors [2]. Based on the literature on ME we present our approach for developing an IoT SDM based on situational factors. For this we first present the process for developing a method base. Then we present the process for developing a method using the method fragments in the method base.

2.1 Method Base Construction

In essence, methods can be constructed either from scratch based on experience, or composed based on modular method parts that reside in a method base [5]. In our case we focus on the development of a method base to support the customized method development. Figure 2 illustrates the process that we propose based on existing literature on ME. The process starts with two parallel steps, identification of the existing methods and construction of a list of situational factors. The identification of existing methods encompasses searching relevant knowledge sources and finding out method specifications. A method specification explains a method in a structured, semi-structured or unstructured way. Meanwhile, a list of situational factors can be constructed, possibly in the form of a taxonomy. The situational factors can be associated with the method parts and hence guide a method engineer in selecting proper method parts for a specific situation.

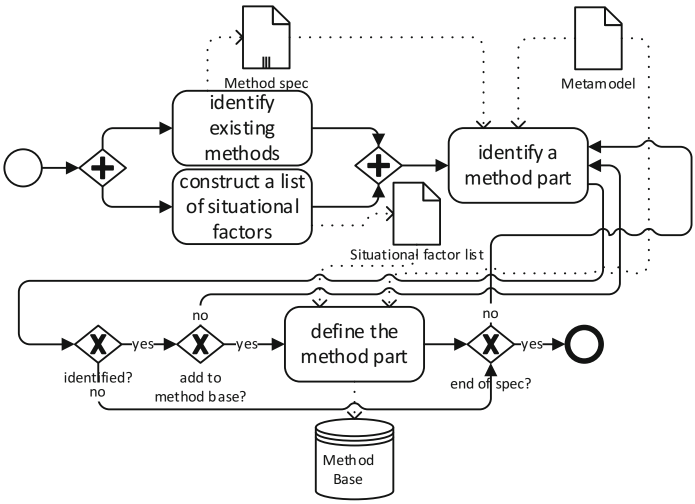


Fig. 2. Process for constructing a method base

The subsequent step involves the identification of method parts based on a metamodel. The metamodel defines the types of concepts and relationships to be extracted from existing method specifications. For instance, such a metamodel can propose to extract tasks to be executed along with the roles contributing to the execution of these tasks. Examples of such a metamodel include ISO/IEC 24744 standard [6], SPEM [7], Open Process Framework [8] and Essence Framework [9]. It is evaluated whether the identified method part can be added to the method base. A method part can be omitted since it does not possess sufficient information to be defined or it has already been added before.

The last step encompasses the definition and addition of a method part to the method base, if it is decided to do so at the previous decision point. The content and structure of the method part should conform to the underpinning metamodel to ensure modularity. For instance, each method part should have a specified number of named attributes, if ISO/IEC 24744 is used as a metamodel. Moreover, a method part can be associated with the situational factors, which can recommend or hinder the use of that method part. The end product of this process is a method base including modular method parts to be reused for method construction.

2.2 Method Construction Using a Method Base

There are some approaches in the literature for method construction [4]. These include assembly-based [10], paradigm-based [11, 12], and extension-based [13] approaches. In this study, we selected the assembly-based approach since it follows the reuse strategy and there are existing IoT SDMs both originated from the industry and academia to be considered for reuse. Figure 3 illustrates a process for method construction based on the assembly-based approach.

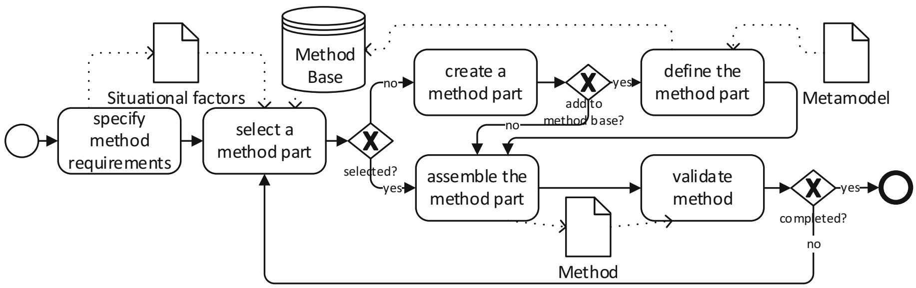


Fig. 3. Process for constructing a method

Before constructing a method, it is necessary to specify the requirements of the intended method [14] (as shown in the first step of our process). These requirements are generally based on organizational and project-specific characteristics, namely situational factors, which typically does not change extremely during a project.

Selecting a method part encompasses querying a method base to retrieve a set of method parts conforming to the situational factors at hand. Some query languages have been reported in the literature [15, 16]. When there are proper method parts retrieved from a method base, these parts are assembled in a coherent manner. If the method base does not contain a part that is needed, a new one can be created. This part is assembled to the previous parts. If this new part has a reuse potential, it can be defined according to the metamodel chosen and added to the method base.

The validation step is about conducting some checks to determine whether the constructed method is ready to use. There is some research on method quality (Sect. 8.4 of [4]), which can help a method engineer for validating a method.

The process illustrated in Fig. 3 is executed manually by a method engineer. She/he uses a method base as a library containing many components along with their reuse context (defined by situational factors).

3 Applying SME for the IoT: Constructing a Method Base Using Existing IoT SDMs

We followed the process illustrated in Fig. 2 to construct a method base for the IoT domain. We formed a sample list of situational factors by exploring the related disciplines, namely software engineering [17, 18], SME [8, 19, 20], project management [21], and IoT [22]. As illustrated in Table 1, we can unify a list of situational factors from various related disciplines to be associated with the method parts.

We also identified six IoT SDMs in the literature [23] that we will use as a basis for the development of the method base. Table 2 lists the names of these six SDMs, the abbreviation used in this paper, and the references documenting these SDMs. These references refer to the method specifications that are used as the sources for identifying and defining method parts. Method parts can be at various level of granularity, namely fragment, chunk, and component [4]. Method fragments refer to a single concept in the metamodel and may have a process, a product or a producer focus [4]. We preferred to use method fragments as the building blocks of our method base since fragments offer more flexibility and reusability at the slight expense of an additional effort for method construction [4] (for a comparison of fragments, chunks, and components refer to the Sect. 2.4 of [4]). Another reason is that the existing IoT SDMs are non-modular and their specifications do not include the sufficient information to form higher level reusable method parts (method chunks and components). Last but not least, [36, 37] report that method bases, which are made up of fragments, have been successfully used in some industry projects.

We adopted ISO/IEC 24744 standard [6] as the metamodel for identifying and defining method fragments. As illustrated in Fig. 4, we used the three major concepts (along with their related concepts) of ISO/IEC 24744 standard. These concepts are work units, work products, and producers, which also form the basis of some other standards, such as SPEM [7] and OPF [8]. A work unit is a task to be performed, process to be executed or technique to be applied within a project. A work product is an intermediate or resulting output produced during a project and can be a document, model, etc. A producer is responsible for executing work units and can be represented as a role (a set of responsibilities a producer can own), a tool (an enabler for producers to fulfill their responsibilities), and a team (an organized group of producers sharing a common objective).

Table 1. A sample list of situational factors

Category	Situational factor	Value	References
Business context & requirements	Regulations	None, low, medium, high	[17, 22]
	Standards	None, low, medium, high	[21, 22]
	Requirements stability	Volatile, stable	[8, 17, 19]
Organization	Size	Small, medium, large	[18–20]
	Maturity	Low, medium, high	[8, 18, 21]
	Management commitment	Low, medium, high	[17–20]
	Structure	Functional, divisional and matrix	[8, 18, 19]
Team	Size	Small, medium, large	[17, 18]
	Geographic distribution	One location, distributed	[17, 18]
	Domain experience	None, low, medium, high	[17, 18, 20, 21]
	Technical experience	None, low, medium, high	[17, 18, 20, 21]
Customer	Availability	On-site, frequently, as per plan, rare	[17, 20]
	Domain experience	None, low, medium, high	[17, 18]
	Resistance	None, low, medium, high	[18, 20]
System	Size	Small, medium, large	[8, 18, 21]
	Complexity	Low, medium, high	[8, 17, 18, 20–22]
	Reuse	None, component, application, product line	[18, 20]
	Technology maturity	None, emergent, example uses exist, mature	[18, 19]
	Existing IoT devices	No, yes	adapted from [18]
	Existing backend services	No, yes	adapted from [18]
	Degree of innovation	Low, medium, high	[17]

We use a plain format (a simplified version of the one used in [40]) for representing the method fragments. This representation conforms to ISO/IEC 24744 standard. Tables 3 and 4 illustrate the detailed representations of two method fragments. These details provide guidance to method engineers for new method construction using the method base.

A method engineer should be able to search a method base in order to have access to the candidate method fragments to be used for new method construction.

Table 2. IoT SDMs in the literature

Method	Abbr.	References
The ignite | IoT methodology	Ignite	[22]
The IoT methodology	IoT-Meth	[24]
IoT application development	IoT-AD	[25, 26]
Event-driven lightweight distilled state charts-based agents methodology	ELDAMeth	[27–29]
A software product line process to develop agents for the IoT	SPLP-IoT	[30–33]
A general software engineering (SE) methodology for IoT	GSEM-IoT	[34, 35]

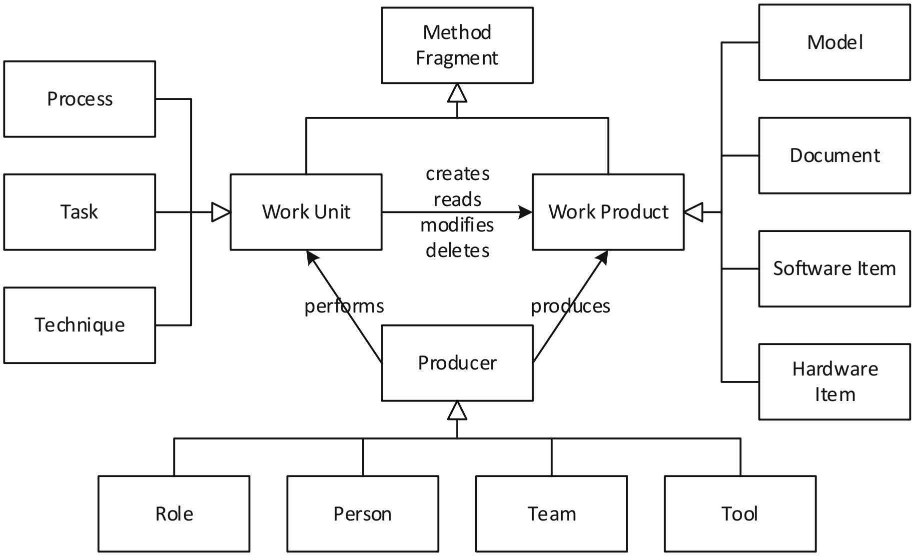


Fig. 4. A partial conceptual model of ISO/IEC 24744 standard used in this work (adapted from [38, 39])

Table 3. An example of a work product

Name	Quantity structure
Description	It specifies the planned change (increase, decrease) in the number of key entities, such as users, messages, IoT devices, etc.
Specification	Consider including a table showing the estimated numbers per period (quarterly, yearly, etc.)
Relationships	None

A descriptor is associated to each method fragment to state the context in which it can be reused. As illustrated in Fig. 5, a method fragment is identified through a name, a type (work unit, work product, producer), and a discipline. Disciplines for method

Table 4. An example of a work unit

Name	Design a system architecture
Purpose	Come up with an architecture fulfilling requirements
Description	A high level design of the system to be developed is critical for guiding the implementers while implementing the system. IoT Reference Architecture and IoT DB, which contains previous designs, can be used while designing an architecture. IoT Canvas provides some of the high level requirements
Relationships	Causes (action)
	Creates architectural design
	Reads IoT Canvas
	Reads IoT Reference Architecture
	Reads IoT DB
	Results (outcomes)
	Architecture has been designed

fragments can be defined at a level of granularity that is required. A sample taxonomy can be borrowed from Rational Unified Process [41]: Business Modeling, Requirements, Analysis and Design, Implementation, Test, Deployment, Configuration and Change Management, Project Management, and Environment. A set of situational factors, each defined by a name and value pair, represents a project specific context in which that method fragment is recommended to be reused. Each situational factor can have a corresponding predefined list of values.

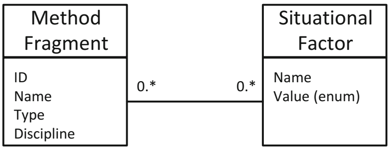


Fig. 5. Method fragment descriptor

Using the descriptor in Fig. 5, a sample search query to be executed on a method base can be as follows: `SELECT method fragment WHERE Type = … AND Discipline = … AND FOR EACH Situational Factor = …`

It aims to select all the method fragments that conform to the context defined by the situational factors. After selecting these reusable fragments, a method engineer has a set of candidate fragments to be used during a project specific method.

4 Applying SME for the IoT: Constructing an IoT SDM

To illustrate the construction of an IoT SDM from the constructed method base, we will use two cases for precision farming in which IoT is used in farm management information systems. Hereby IoT is used to increase the amount of production and economic returns, often also with the goal to reduce the impact on the environment. Several important benefits of precision farming have been provided in the literature including optimizing production efficiency, optimizing quality of the crop, minimizing environmental impact, minimizing risk, conservation of resources, reducing cost, increasing profit, and better management decisions [42]. Obviously precision farming is a specific domain that requires the consideration of proper steps to develop the system that meets the stakeholder requirements. Various different contexts might require various different IoT SDMs. In some cases, it would be sufficient to select a light IoT SDM while other cases might be more stringent and require detailed method steps. Whatever the situation will be, in essence we could use the method base that was developed so far. For the first case, we assume that an IoT SDM is needed for a small-scale farm information system. For the second case, we construct a method for a large-scale integrated farm management system.

4.1 Case 1: An IoT SDM for Small-Scale Farm Management System

A small-scale farm needs an information system to support its agricultural decision making process. A small team with a high domain knowledge in precision farming will develop the system. The farmers are the foremost customers and two of them are chosen as full-time customer representatives to be involved in this project. The degree of innovation is medium, hence some explorative approach is needed to mitigate project risks. Existing IoT devices and backend services will be used in the project.

Following the process illustrated in Fig. 3, we first identified a set of situational factors (listed in Table 5) based on our scenario.

Table 5. A set of example situational factors defining a project-specific context for case 1

#	Situational factor	Value
SF1	[Team].[Size]	Small
SF2	[Team].[Domain experience]	High
SF3	[Customer].[Availability]	On-site
SF4	[System].[Degree of innovation]	Low
SF5	[System]. [Existing IoT devices]	Yes
SF6	[System]. [Existing backend services]	Yes

Table 6 shows the method fragments that have been selected for this case. Based on the retrieved method fragments, the derived process flow is obtained as illustrated in Fig. 6. We can represent the business perspective using a lightweight format, which is IoT Canvas in this case, since the team has high domain experience. A lightweight format (user story in this case) can be used to specify requirements and these requirements can be detailed orally with the two customer representatives, who will be on-site. Since the system will use existing IoT devices and an existing IoT platform, it might be beneficial to assess the characteristics of these IoT devices and IoT platform. The system architecture can be designed based on the requirements as well as the IoT reference architectures available. These reference architectures guide the designers in applying best solutions to common problems in the IoT domain. For each iteration, the user stories to be developed can be selected and an incremental IoT system can be presented to the customer to receive feedback and revise the system accordingly. As can be seen in the figure, a method engineer can create new method parts and add them to the method, such as “develop”, “review”, “revise and enhance” in this case.

4.2 Case 2: An IoT SDM for Large-Scale Integrated Farm Management System

An IT company aims to develop and market an integrated farm management system. The system is targeted to individual farmers who want to manage their farm by

Table 6. A set of example method fragments selected for case 1

ID	Name	Type	Discipline	Recommendation on situational context
MF001	Produce an IoT Canvas	Work unit	Business modeling	
MF002	IoT Canvas	Work product	Business modeling	
MF003	Business process spec.	Work product	Business modeling	Especially recommended when [Team].[Domain experience] = none, low [17]
MF004	Business glossary	Work product	Business modeling	Especially recommended when [Team].[Domain experience] = none, low [17]
MF005	Specify system requirements	Work unit	Requirements	
MF006	User story	Work product	Requirements	Recommended when [Team].[Size] = small; [Customer].[Availability] = on-site [17]
MF007	Use case (short form)	Work product	Requirements	Recommended when [Team].[Size] = small; [Customer].[Availability] = on-site [17]
MF008	Use case (long form)	Work product	Requirements	Especially recommended when [Team].[Size] = small; [Customer].[Availability] = as per plan, rare [17]
MF009	System requirements specification	Work product	Requirements	Especially recommended when [Team].[Size] = small; [Customer].[Availability] = as per plan, rare [17]
MF010	Design a system architecture	Work unit	Analysis and design	

Table 7. A set of example situational factors defining a project-specific context for case 2

#	Name	Value
SF1	[Team].[Size]	Medium
SF2	[Team].[Domain experience]	Low
SF3	[Customer].[Availability]	As per plan
SF4	[System].[Degree of innovation]	Medium
SF5	[System]. [Existing IoT devices]	No
SF6	[System]. [Existing backend services]	No

planning, monitoring, recording, tracking, and analyzing all farming activities. The company allocated a medium-sized team with low domain experience for this project. The requirements will be elicited with some customers from some representative farms. The company will be able to get limited collaboration with these customers. There is no hardware and infrastructure present to be used in the project. The situational context for this case is illustrated in Table 7.

Based on the situational factors of the second case, we have selected the method fragments listed in Table 8 from the method base.

Based on these situational factors we constructed another method. Some of the steps of this method are illustrated in Fig. 7. Since the team has low domain experience, it might be beneficial to document business requirements along with business processes to create a certain level of business understanding. Moreover, these documents can be used for supporting the communication in a medium-sized team. Use cases can be used for representing requirements. A long format can be preferred since the customers are only available as per plan. Therefore, the customers cannot be always accessible for detailing requirements while developing the system. A system architecture can be designed using IoT reference architectures. Since neither IoT devices nor an IoT platform exists, an assessment and selection should be done for both. Moreover, a site survey can be done to explore the physical environment in which IoT devices will reside. Afterwards, a prototype can be developed and reviewed with the customers.

4.3 Case 1 vs. Case 2

As shown in these two cases, it is possible to select different method fragments for different situations and integrate them to construct a project-specific method. A method base containing reusable method fragments is an enabler for such a method construction.

In case 2, the project team needs to increase its knowledge of the business process. Some work products, such as business process specification, can be used to help the project team in understanding the business context in which IoT system will operate. Other options for such a purpose are business vision, business use case, business use case model, and business glossary from RUP.

Since the company in case 2 plans to develop a system to be used by many farmers, it is required to project the growth in key areas, such as number of assets, users, events, etc. Quantity structure work product as proposed by the Ignite method could be, for example, used for this purpose.

Table 8. A set of example method fragments selected for case 2

ID	Name	Type	Discipline	Recommendation on situational context
MF001	Produce an IoT Canvas	Work unit	Business modeling	
MF002	IoT Canvas	Work product	Business modeling	
MF003	Business process spec.	Work product	Business modeling	Especially recommended when [Team].[Domain experience] = none, low [17]
MF004	Business glossary	Work product	Business modeling	Especially recommended when [Team].[Domain experience] = none, low [17]
MF005	Specify system requirements	Work unit	Requirements	
MF008	Use case (long form)	Work product	Requirements	Especially recommended when [Team].[Size] = small; [Customer].[Availability] = as per plan, rare [17]
MF009	System requirements specification	Work product	Requirements	Especially recommended when [Team].[Size] = small; [Customer].[Availability] = as per plan, rare [17]
MF010	Design a system architecture	Work unit	Analysis and design	
MF011	Carry out a site survey	Work unit	Analysis and design	Especially recommended when [System]. [Existing IoT devices] = no
MF012	Site survey document	Work product	Analysis and design	Especially recommended when [System]. [Existing IoT devices] = no
MF013	Assess and choose an IoT platform	Work unit	Analysis and design	
MF014	Develop a prototype	Work unit	Implementation	Recommended when [System]. [Degree of innovation] = medium, high [17]

In case 2, user stories are replaced with use cases for requirements specification. The rationale for replacing this method fragment is that user stories are not detailed and need further elaboration with the customers, which is not feasible since the customers are not easily accessible in case 2.

Since the first case has some IoT devices and an IoT platform to be reused, it is required to analyze the capabilities of this infrastructure. For the second case, an assessment and selection process should be executed to form a proper infrastructure that can fit the system architecture designed and fulfill the business requirements. While selecting the IoT devices, a site survey can be conducted to provide an understanding of the environment in which IoT devices will operate.

In both cases, an incremental development cycle has been selected to receive feedbacks from the customers and revise the system accordingly. The periods of the feedback cycles can be different in both project, possible shorter in the first case since there are on-site customers.

5 Discussion

5.1 Why not Selecting an Existing IoT SDM?

It is generally accepted that every project is unique and requires a unique method fulfilling its special requirements. The existing methods accept this fact and may present a guideline for tailoring the method for a specific project. On the other hand, there are many types of projects having different situational contexts. It is practically impossible for a method to include all of the possible work units and products. Therefore, a method base can be used to enhance an existing method and obtain a new method.

5.2 Why a Method Base Instead of a Complete Method?

Many disciplines, such as system engineering, software engineering, project management agree that each project has its own dynamics and asks for a tailored method. On the other hand, there are some approaches to be reused under certain situations. These are generally named as “best practices” in the industry. We propose to share these approaches and best practices in the form of method parts to be reused. A complete method for a specific project can be constructed using these method parts, whereas a complete method meeting all project requirements under every condition is not realistic.

5.3 What Are the Lessons Learned from Method Base Construction?

Ignite is documented using a book which contains many details and examples. Even if the book has been structured around sections, we had many difficulties in separating the descriptions of the method parts from project specific details. Moreover, some of the method parts are not documented well and this makes them difficult to understand and reuse. For instance, Ignite mentions two roles, business analyst and solution architect, but does not clearly define their responsibilities, the work units they take part in and the work products they work with. IoT-Meth is described only by a short presentation, which is unstructured and far from guiding a project team. Moreover, the method parts defined in IoT-Meth are not described sufficiently. We think that standards like ISO/IEC 24744 can guide for constructing such methods in providing understandable and reusable method parts.

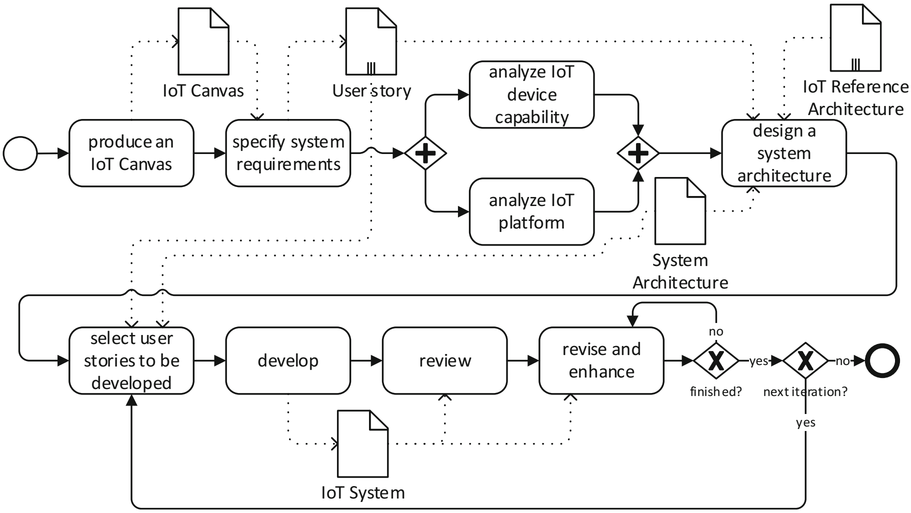


Fig. 6. The derived process flow made up of the method fragments in Table 6

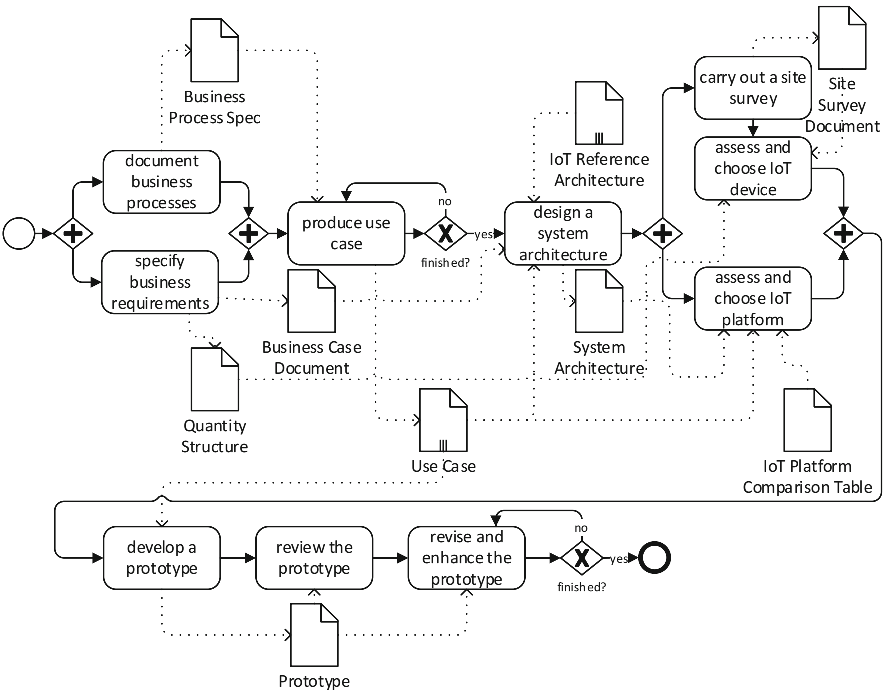


Fig. 7. The derived process flow made up of the method fragments in Table 8

5.4 Does This Work Make Any Preference on Plan-Driven and Agile Paradigms?

No! Both plan-driven and agile paradigms include a systematic approach hence a method. Plan-driven paradigm prefers to follow a more detailed plan/process/method compared to agile paradigm. Agile paradigm is in favor of distributing the responsibility to individuals and a more autonomous project team [43]. On the other hand, this does not mean that agile paradigm overlooks the importance of methods. The Scrum Guide [44] mainly defines roles, work products, and work units, which are actually reusable method parts. Moreover, Scrum itself is a lightweight framework and should be complemented with other method parts such as user story, use case. As a summary, both plan-driven and agile paradigms can use a method base to construct a proper method at varying level of details.

5.5 What Is the Implication of this Work for Industrial Use?

Some industrial attempts for constructing IoT SDMs are present [22, 24]. These SDMs are mainly based on the existing method knowledge of system and software engineering. For instance, Ignite includes many work products (use cases, UI mockups, domain model) that are well-known even before IoT exists. Ignite enriches its content with some new work products (such as quantity structure) specific to the IoT domain. As a result, the creators of Ignite are already applying SME approach implicitly. This implicit application of SME ends up with non-modular an IoT SDM that hinders reusability.

Another attempt in IoT industry to support reusability is the concept of “reference architecture”. There are many IoT reference architectures designed so far [45]. These work products can also be catalogued in a method base and served for reuse. Considering these observations, we think that having a method base can provide significant value to IoT industry by preventing many project teams from “reinventing the wheel”.

6 Related Work

Software reuse has been an important goal in many industrial practices to increase productivity and quality of the software artifacts. Several reuse approaches have been provided such as abstract data types, module-based programming, component-based software development, reusable libraries and design patterns. Unlike these earlier software reuse approaches, software product line engineering (SPLE) aims to provide pro-active, pre-planned reuse at a large granularity to develop applications from a core asset base [46]. A product line is defined as a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way. An important activity in the product line engineering process is development of the core asset base which aims to store the reusable software assets that can be reused for developing individual products. Our approach adopts a similar approach in which we focus on the development of a method base. The method base however does not include software assets for developing software systems, but method assets for developing a method with which we can further build IoT systems.

In [1] the authors propose to construct a practice library (a similar concept to method base) for the IoT domain. They present a proof of concept using some method parts of Ignite. Essence Language is used in [1] as a metamodel. [47] uses the same idea of having a practice library based on Essence Framework and proposes an initial library after partially analyzing two IoT SDMs. Unlike [1, 47], In this work, we extracted method parts from the six IoT SDMs based on ISO/IEC 24744 standard and associated these with situational factors.

The Essence Framework specification provides some examples of method parts extracted from existing methods (Scrum, Unified Process) and represented in Essence Language [9]. Park provided the representation of the method parts of Scrum using Essence Language [48]. Moreover, Park provided a hypothetical case for constructing a method using some method parts from Scrum, XP, and DevOps. In our earlier work, we provided an approach for extracting method parts from existing methods and representing them using Essence Language [49]. This work also includes a partial representation of the method parts of Nexus [50] in Essence Language. [51] extracted and represented the method parts of Synthesis-based Software Architecture Design Approach [52] in Essence Language. [40] uses the Australian Standard methodology metamodel of AS 4651 to represent the method parts extracted from XP, Scrum and Crystal. As distinct from these studies, we focus on IoT SDMs within the scope of this work.

In our earlier work [54, 55] we have focused on variability in current IoT systems. Thereby we have discussed the need for different component models for IoT systems and the existence of different IoT protocols that are required for different situational factors. The focus of this study that targets the development of IoT methods as such is complementary to our earlier studies.

7 Conclusions and Future Work

Developing IoT systems is not trivial and requires the consideration of multiple different stakeholder concerns. Hence adopting a systematic method in which the steps as well as the targeted artifacts are clearly defined is important. However, IoT systems can range from a simple system consisting of a few components to a large scale system that integrates multiple different systems. Obviously for different situations different methods will be required. To address this issue, we have provided a situational method engineering approach for developing IoT SDMs. We have distinguished two different processes including the construction of the method base, and the application of the method base to construct an IoT SDM. We have adopted six different IoT SDMs in the development of the method base. The development of the method base itself provides important insights in the IoT SDMs and paves the way for further discussion and research. From the practical perspective we could state that once the method base was ready we could easily construct methods for the situation at hand. We have illustrated the development of two distinct IoT SDMs with different situational requirements. We believe that the method base can be applied to a broader set of IoT application domains.

In our future work we will enhance the method base with input from other IoT SDMs and develop the corresponding toolset for this. Further we will use our method base to develop other IoT SDMs.

References

1. Jacobson, I., Spence, I., Ng, P.W.: Is there a single method for the Internet of Things? Queue **15**(3), 20 (2017)
2. Henderson-Sellers, B., Ralyté, J.: Situational method engineering: state-of-the-art review. J. Univ. Comput. Sci. **16**(3), 424–478 (2010)
3. Brinkkemper, S.: Method engineering: engineering of information systems development methods and tools. Inf. Softw. Technol. **38**(4), 275–280 (1996)
4. Henderson-Sellers, B., Ralyté, J., Ågerfalk, P., Rossi, M.: Situational Method Engineering. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-41467-1
5. Ralyté, J.: Towards situational methods for information systems development: engineering reusable method chunks. In: Vasilecas, O., Caplinskas, A., Wojtkowski, W., Wojtkowski, W.G., Zupancic, J., Wrycza, S. (eds.) Proceedings of the 13th International Conference on Information Systems Development, pp. 271–282 (2004)
6. ISO/IEC JTC 1/SC 27: ISO/IEC 24744:2014. Software engineering – metamodel for development methodologies, Geneva (CH) (2014)
7. Object Management Group: Software & systems process engineering meta-model specification, Version 2 (2008). http://www.omg.org/spec/SPEM/2.0
8. Firesmith, D., Henderson-Sellers, B.: The OPEN Process Framework: An Introduction. Pearson Education, London (2002)
9. Object Management Group: Essence - kernel and language for software engineering methods, Version 1.1 (2008). http://www.omg.org/spec/Essence
10. Ralyté, J., Rolland, C.: An assembly process model for method engineering. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) CAiSE 2001. LNCS, vol. 2068, pp. 267–283. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45341-5_18
11. Rolland, C.: A user centric view of Lyee requirements. In: Fujita, H., Johannesson, P. (eds.) New Trends in Software Methodologies, Tools and Techniques. IOS Press (2002)
12. Ralyté, J., Rolland, C., Ben Ayed, M.: An approach for evolution-driven method engineering. In: Krogstie, J., Halpin, T., Siau, K. (eds.) Information Modeling Methods and Methodologies (Advanced Topics in Database Research), pp. 80–100. Idea Group, Hershey (2005)
13. Deneckère, R., Souveyet, C.: Patterns for extending an OO model with temporal features. In: Rolland, C., Grosz, G. (eds.) OOIS 1998. Springer, London (1998). https://doi.org/10.1007/978-1-4471-0895-5_14
14. Gupta, D., Prakash, N.: Engineering methods from method requirements specifications. Requir. Eng. **6**, 135–160 (2001)
15. Rolland, C., Prakash, N.: A proposal for context-specific method engineering. In: Brinkkemper, S., Lyytinen, K., Welke, R.J. (eds.) Method Engineering. ITIFIP, pp. 191–208. Springer, Boston, MA (1996). https://doi.org/10.1007/978-0-387-35080-6_13
16. Brinkkemper, S., Saeki, M., Harmsen, F.: A method engineering language for the description of systems development methods. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) CAiSE 2001. LNCS, vol. 2068, pp. 473–476. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-45341-5_33

17. Kalus, G., Kuhrmann, M.: Criteria for software process tailoring: a systematic review. In: Proceedings of the 2013 International Conference on Software and System Process. ACM (2013)
18. Clarke, P., O'Connor, R.V.: The situational factors that affect the software development process: towards a comprehensive reference framework. Inf. Softw. Technol. **54**(5), 433–447 (2012)
19. Khan, H.H., bin Mahrin, M.N., Chuprat, bt, S.: Review of support to situational requirement engineering from standards and models. Int. J. Digit. Inf. Wirel. Commun. (IJDIWC), **4**(1), 79–94 (2014)
20. Kornyshova, E., Deneckère, R., Salinesi, C.: Method chunks selection by multicriteria techniques: an extension of the assembly-based approach. In: Ralyté, J., Brinkkemper, S., Henderson-Sellers, B. (eds.) Situational Method Engineering: Fundamentals and Experiences. ITIFIP, vol. 244, pp. 64–78. Springer, Boston, MA (2007). https://doi.org/10.1007/978-0-387-73947-2_7
21. Office of Government Commerce: Managing successful projects with PRINCE2. The Stationery Office (2009)
22. Slama, D., Puhlmann, F., Morrish, J., Bhatnagar, R.M.: Enterprise IoT Strategies and Best Practices for Connected Products and Services. O'Reilly Media, Inc. (2016)
23. Giray, G., Tekinerdogan, B., Tüzün, E.: IoT system development methods. In: Hassan, Q.F., Khan, A.R., Madani, S.A. (eds.) Internet of Things: Challenges, Advances and Applications, pp. 141–159. Chapman and Hall/CRC (2018)
24. Collins, T.: A methodology for building the Internet of Things. http://www.iotmethodology.com/. Accessed 4 Oct 2017
25. Patel, P., Cassou, D.: Enabling high-level application development for the Internet of Things. J. Syst. Softw. **103**(C), 62–84 (2015)
26. Patel, P.: Enabling high-level application development for the Internet of Things, Ph.D. diss., Universite Pierre et Marie Curie - Paris VI (2014)
27. Fortino, G., Russo, W.: ELDAMeth: an agent-oriented methodology for simulation-based prototyping of distributed agent aystems. Inf. Softw. Technol. **54**(6), 608–624 (2012)
28. Fortino, G., Rango, F., Russo, W.: ELDAMeth design process. In: Cossentino, M., Hilaire, V., Molesini, A., Seidita, V. (eds.) Handbook on Agent-Oriented Design Processes, pp. 115–139. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-39975-6_5
29. Fortino, G., Guerrieri, A., Russo, W., Savaglio, C.: Towards a development methodology for smart object-oriented IoT systems: a metamodel approach. In: IEEE International Conference on Systems, Man, and Cybernetics (2015)
30. Ayala, I., Amor, M.: Self-configuring agents for ambient assisted living applications. Ubiquit. Comput. **17**(6), 1159–1169 (2012)
31. Ayala, I., Amor, M., Fuentes, L.: An agent platform for self-configuring agents in the Internet of Things. In: 3rd International Workshop on Infrastructures and Tools for Multi-agent Systems, ITMAS, Valencia, Spain, pp. 65–78 (2012)
32. Ayala, I., Amor, M., Fuentes, L.: Towards a CVL process to develop agents for the IoT. In: Hervás, R., Lee, S., Nugent, C., Bravo, J. (eds.) UCAmI 2014. LNCS, vol. 8867, pp. 304–311. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-13102-3_51
33. Ayala, I., Amor, M., Fuentes, L.: A software product line process to develop agents for the IoT. Sensors **15**(7), 15640–15660 (2015)
34. Zambonelli, F.: Towards a discipline of IoT-oriented software engineering. In: 17th Workshop "From Objects to Agents", Catania, Italy, pp. 1–7 (2016)
35. Zambonelli, F.: Key abstractions for IoT-oriented software engineering. IEEE Softw. **34**(1), 38–45 (2017)

36. Bajec, M., Vavpotic, D., Krisper, M.: Practice-driven approach for creating project-specific software development methods. Inf. Softw. Technol. **49**(4), 345–365 (2007)
37. Coulin, C., Zowghi, D., Sahraoui, A.-E.-K.: A situational method engineering approach to requirements elicitation workshops in the software development process. Softw. Process Improv. Pract. **11**(5), 451–464 (2006)
38. Ruy, F.B., de Almeida Falbo, R., Barcellos, M.P., Guizzardi, G.: An ontological analysis of the ISO/IEC 24744 metamodel. In: FOIS, pp. 330–343 (2014)
39. Henderson-Sellers, B., Gonzalez-Perez, C.: Granularity in conceptual modelling: application to metamodels. In: Parsons, J., Saeki, M., Shoval, P., Woo, C., Wand, Y. (eds.) ER 2010. LNCS, vol. 6412, pp. 275–288. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16373-9_16
40. Tran, Q.N.N., Henderson-Sellers, B., Hawryszkiewycz, I.: Agile method fragments and construction validation. In: Handbook of Research on Modern Systems Analysis and Design Technologies and Applications, pp. 243–270. IGI Global (2009)
41. Kroll, P., Kruchten, P.: The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP. Addison Wesley, Boston (2003)
42. Verdouw, C.N., Wolfert, S., Tekinerdogan, B.: Internet of Things in agriculture. CAB Rev.: Perspect. Agric. Vet. Sci. Nutr. Nat. Resour. **11** (2016)
43. Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W. et al.: Manifesto for agile software development (2001). http://agilemanifesto.org/. Accessed 5 Oct 2017
44. Schwaber, K., Sutherland, J.: Scrum Guide (2016)
45. Weyrich, M., Ebert, C.: Reference architectures for the Internet of Things. IEEE Softw. **33** (1), 112–116 (2016)
46. Clements, P., Northrop, L.: Software product lines. Addison-Wesley, Boston (2002)
47. Giray, G., Tekinerdogan, B., Tüzün, E.: Adopting the essence framework to derive a practice library for the development of iot systems. In: Mahmood, Z. (ed.) Connected Environments for the Internet of Things. CCN, pp. 151–168. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70102-8_8
48. Park, J.S., Mcmahon, P.E., Myburgh, B.: Scrum powered by Essence. ACM SIGSOFT Softw. Eng. Notes **41**(1), 1–8 (2016)
49. Giray, G., Tüzün, E., Tekinerdogan, B., Macit, Y.: Systematic approach for mapping software development methods to the essence framework. In: 5th International Workshop on Theory-Oriented Software Engineering, pp. 26–32 (2016)
50. Schwaber, K.: Nexus Guide (2015)
51. Giray, G., Tekinerdogan, B., Tüzün, E.: Modeling synthesis-based software architecture design approach using essence framework (in Turkish). J. Sci. Eng. **19**(55), 1 (2017)
52. Tekinerdogan, B.: Synthesis-based software architecture design, Ph.D. diss., Twente University (2000)
53. Tekinerdogan, B.: Engineering Connected Intelligence: A Socio-Technical Perspective. Wageningen University and Research, Wageningen, p. 40 (2017). http://edepot.wur.nl/401115. ISBN 9789463430494
54. Kaya, M.C., Saeedi Nikoo, M., Suloglu, S., Tekinerdogan, B., Dogru, A.H.: Managing heterogeneous communication challenges in the Internet of Things using connector variability. In: Mahmood, Z. (ed.) Connected Environments for the Internet of Things. CCN, pp. 127–149. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70102-8_7
55. Köksal, Ö., Tekinerdogan, B.: Feature-driven domain analysis of session layer protocols of Internet of Things. In: 2017 IEEE International Congress on Internet of Things (ICIOT), Honolulu, HI, pp. 105–112 (2017)

Short Papers

Towards Blockchain Support for Business Processes

Jan Mendling(✉)

Wirtschaftsuniversität Wien, Welthandelsplatz 1, 1020 Vienna, Austria
jan.mendling@wu.ac.at

Abstract. Blockchain technology bears the potential to support the execution of inter-organizational business processes in an efficient way. Furthermore, it addresses various notorious problems of collaboratively designing choreographies and overcoming lack of trust. In this paper, we discuss this potential in more detail and highlight several research challenges that future research has to address towards generic blockchain support for inter-organizational business processes in various application scenarios.

Keywords: Business process management · Blockchains
Smart contracts · Inter-organizational business processes

1 Introduction

Business processes are collections of inter-related events, activities, and decision that collectively create value for a customer [1]. Many business processes involve several parties, which makes it difficult to manage them from the holistic perspective of an overarching choreography [5]. Classical approaches to business process management (BPM) are mainly concerned with the discovery, analysis, implementation and monitoring of *intra*-organizational processes. Yet, for *inter*-organizational processes, challenges of collaborative design and a lack of trust have limited the uptake of BPM techniques.

Emerging *blockchain* technology has the potential to provide substantial improvements for inter-organizational business processes. Originally, blockchains belong to a specific category of distributed database technology that builds on tamper-proof lists of timestamped transaction records. They are best known for being used for cryptocurrencies such as Bitcoin [8]. Blockchains offer a way to execute processes in a trustworthy manner even if partners do not have mutual trust in each other. Thanks to their capabilities, blockchains have the potential to enable inter-organizational process support in completely new ways.

This paper describes opportunities and challenges of blockchains for BPM. Section 2 discusses the specific benefits that blockchain technology provides for process execution. Section 3 discusses challenges of executing processes on the blockchain along the different phases of the *BPM lifecycle phases* [1]. Finally, Sect. 4 summarizes the discussion and highlights directions for future research.

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 243–248, 2018.
https://doi.org/10.1007/978-3-319-94214-8_15

2 Potential of Blockchains for Executing Processes

A key idea of business process management is that new information technology can be used to make the execution of business processes more efficient and effective. The utilization of new information technology can have three types of impact on a given business process: automational, informational or transformational [7]. Automational effects emerge when a new technology is used to automate tasks that have been previously done manually or with partial system support. Informational effects materialize from better tracking, monitoring, and analytical insights. Transformational effects relate to the changes in the mechanisms of coordination, including disintermediation, outsourcing or offshoring. For blockchain technology, transformational effects are most relevant, because blockchains provide a fundamentally different way of coordination for business processes [4], which could lead to fully new ways of organizing inter-organizational business processes.

Blockchain is a distributed ledger technology that maintains a tamper-proof list of timestamped transaction records. It is used, for instance, for cryptocurrencies like Bitcoin [8]. Since it is tamper-proof, it allows parties to transact with others they do not trust on a network in which nobody is trusted. To provide this feature, blockchains build on peer-to-peer networks, consensus-making, cryptography, and market mechanisms as underlying technologies.

The potential of blockchains to transform inter-organizational processes builds on the concept of smart contracts [10]. Consider the example of a buyer ordering 200 items from the vendor. A failure of the vendor to deliver on time might entitle the buyer to receive a compensation. Such conditional logic can be expressed using smart contracts, which only use information that is stored on the blockchain. When a smart contract is deployed to the blockchain, it is immutable.

The support for smart contracts can also be used to implement more complex logic as required for executing business processes. Model-driven approaches that take a BPMN specification as an input and automatically generate the corresponding blockchain artifacts as an output are described in [2,11]. Also a business process management system called caterpillar has been developed that fully runs on the blockchain [3].

To illustrate this aspects, consider the BPMN process model from Fig. 1. It shows a simplified supply chain scenario, where a bulk buyer orders goods from a manufacturer. The manufacturer, in turn, orders supplies through a middleman, which are sent from the supplier to the manufacturer via a special carrier. The different modeling elements that are used for the specification of this inter-organizational business process can be translated into Solidity smart contract code as defined in [11]. The central idea is to store relevant status information on the blockchain and to represent all information exchange as transactions on the blockchain. Resulting from this idea is a pattern-based transformation of different BPMN primitives such as gateways, control sequence, and different types of tasks to smart contract code.

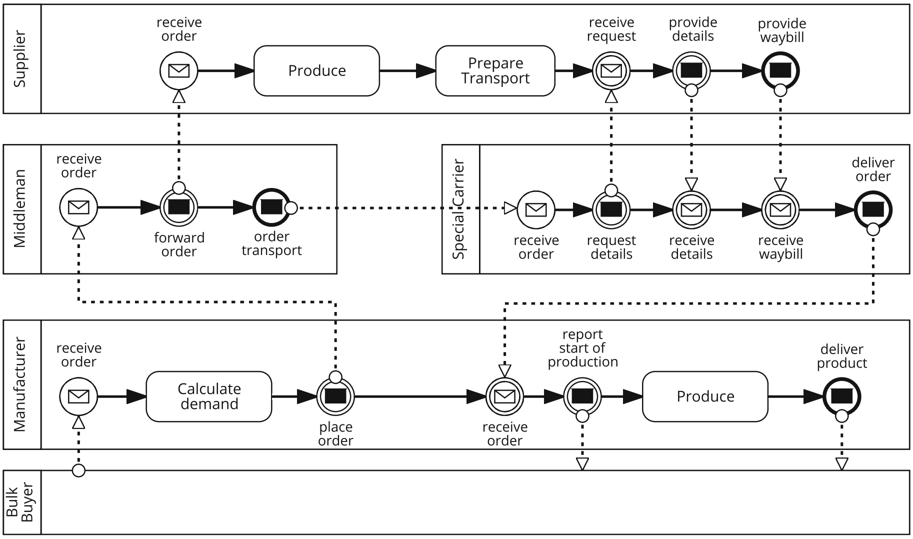


Fig. 1. Supply chain scenario from [11]

This manufacturing scenario underlines the potential of blockchains to have a transformational effect on the way inter-organizational business processes are implemented and executed [4]. Up until now, the implementation of such inter-organizational processes required the collaborative design of choreographies [5]. The scenario shown in Fig. 1 is simple, but involves already five participants [11]. The complications of a collaborative design explain why choreographies have seen hardly a broader uptake in practice. Regarding execution, the situation is even more complicated. Inter-organizational processes can become subject to conflicts. For instance, the manufacturer might receive the materials three days later than agreed. In such a case, the manufacturer will likely trust more what is stored in his own information systems and less what the supplier or the middleman might have recorded. Having the transactional data in the blockchain is a reliable way to store process-related data and provide a single point of truth over that no single party has exclusive control.

This example illustrates that business processes provide a useful anchor for discussing the potential benefits of blockchain technology. In this context, several phases of the BPM lifecycle are of specific importance [1]:

- In the redesign phase, blockchains provide the potential to design to-be process models of inter-organizational processes in such a way as if they were usual intra-organizational processes using the standard BPMN notation.
- In the implementation phase, transformation techniques like [11] and systems like caterpillar [3] offer an efficient implementation of blockchain-supported business processes in a model-driven way.

– During monitoring, blockchains provide a single point of truth for tracing the progress of a process instance. There is no need for message exchange and there are no messages that can be lost or corrupted.

Some of these potential benefits have been demonstrated by prototypical implementations. Still, there are several challenges along the way towards making blockchain-based execution of business processes an everyday approach in corporate information technology.

3 Challenges for Blockchain-Based Process Support

Though there is great potential of blockchain technology for executing business processes, there are also challenges. First, there are generic challenges of blockchain technologies. These include problems of throughput, latency, bandwidth, security, usability, wasted resources and handling of hard forks [9]. Many of these challenges are subject of current research.

Second, there are more specific challenges regarding the way how business processes can be supported with blockchains. A recent research commentary identifies seven challenges for blockchain-based execution of business processes [6]. These include the following:

1. *Execution and monitoring systems:* Currently, there is limited support available in terms of *execution and monitoring systems* for blockchains. A specific challenge will be fragmentation and encryption of execution data. For effective monitoring, data on the blockchain likely has to be integrated with local off-chain data. Design science and algorithm engineering are required here in order to provide novel insights and better tooling.
2. *Methods for analysis and engineering:* Up until now, there are only few methods for analysis and engineering for the specification of business processes on blockchain technology. Transaction data on the blockchain contain potentially valuable data for process analysis. However, how easily the data can be translated into a format that permits process analysis? Design science and software engineering is required towards providing easier deployment and formal analysis.
3. *Redesigning processes:* While there are first insights how business processes can run on the blockchain, it is hardly understood how business processes can be best innovated using the potentials of blockchains. Research on process improvement has identified various process redesign heuristics, and likely there are specific heuristics for adopting blockchains for specific types of processes. Both design science and management science provide to foundations to arrive at substantial insights in this context.
4. *Evolution and adaptation:* The definition of appropriate methods for evolution and adaptation is an important challenge. For instance, it might be a desirable scenario to adapt blockchain processes in a predefined way. Formal concepts are needed in order to provide important guarantees, building on insights from theoretical computer science and verification.

5. *Adoption:* Up until now, it is not clear in which circumstances business processes shall be best put on the blockchain and to which extent this will prove valuable. Different process stakeholders might be risk-averse and less willing to use a novel technology that potentially reduces their control of specific processes. Empirical research is needed to investigate which characteristics of blockchain as a technology best meet requirements of specific processes.
6. *Strategy and governance:* Blockchains will likely enable new governance models with an overall impact on business strategy. It is an open question if blockchain-related activities should best be allocated to a separate business unit and how promising prototypes can be handed over into production. Empirical research is required to investigate this topic building on insights from organizational science.
7. *Corporate culture:* Blockchain technology will likely bring a culture shift towards openness in the management and execution of business processes. Blockchains will likely promote an organizational culture that emphasizes flexibility and is outward-looking. Research in this area will have to investigate how corporate culture changes with the introduction of blockchains, building on empirical research methods.

The spectrum of these challenges is broad. While many of them relate to design and engineering questions, it is clear that blockchains also have to be understood as part of larger socio-technical systems that are open and worldwide. This fact poses various challenges that require an empirical research agenda that integrates perspectives of the social sciences.

4 Conclusions

This paper has briefly summarized important potentials of blockchain technology to support the execution of business processes. Potential benefits are efficient design of inter-organizational business processes in a model-driven way, in which the blockchain is a single point of truth for tracing the process execution. Furthermore, we presented a digest of seven challenges including execution and monitoring systems, analysis and engineering techniques, redesign methods, evoluation and adaptation concepts, adoption in practice, and corporate culture. Interdisciplinary research is required building on formal, empirical and engineering methods in order to address these challenges.

References

1. Dumas, M., Rosa, M.L., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management, 2nd edn. Springer, Heidelberg (2018). https://doi.org/10.1007/978-3-662-56509-4
2. García-Bañuelos, L., Ponomarev, A., Dumas, M., Weber, I.: Optimized execution of business processes on blockchain. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 130–146. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_8

3. López-Pintado, O., García-Bañuelos, L., Dumas, M., Weber, I.: Caterpillar: a blockchain-based business process management system. In: Clarisó, R., Leopold, H., Mendling, J., van der Aalst, W.M.P., Kumar, A., Pentland, B.T., Weske, M. (eds.) Proceedings of the BPM Demo Track and BPM Dissertation Award. CEUR Workshop Proceedings, vol. 1920, CEUR-WS.org(2017). http://ceur-ws.org/Vol-1920
4. Mendling, J., Decker, G., Hull, R., Reijers, H., Weber, I.: How do machine learning, robotic process automation and blockchains affect the human factor in business process management? Communications of the AIS **42** (2018)
5. Mendling, J., Hafner, M.: From WS-CDL choreography to BPEL process orchestration. J. Enterp. Info. Manag. **21**(5), 525–542 (2008)
6. Mendling, J., Weber, I., van der Aalst, W.M.P., Brocke, J.V., Cabanillas, C., Daniel, F., Debois, S., Ciccio, C.D., Dumas, M., Dustdar, S., Gal, A., García-Bañuelos, L., Governatori, G., Hull, R., Rosa, M.L., Leopold, H., Leymann, F., Recker, J., Reichert, M., Reijers, H.A., Rinderle-Ma, S., Solti, A., Rosemann, M., Schulte, S., Singh, M.P., Slaats, T., Staples, M., Weber, B., Weidlich, M., Weske, M., Xu, X., Zhu, L.: Blockchains for business process management - challenges and opportunities. ACM Trans. Manag. Inf. Syst. **9**(1), 4:1–4:16 (2018)
7. Mooney, J.G., Gurbaxani, V., Kraemer, K.L.: A process oriented framework for assessing the business value of information technology. DATA BASE **27**(2), 68–81 (1996)
8. Nakamoto, S.: Bitcoin: a peer-to-peer electronic cash system (2008)
9. Swan, M.: Blockchain: Blueprint for a New Economy. O'Reilly Media Inc., Sebastopol (2015)
10. Szabo, N.: Formalizing and securing relationships on public networks. First Monday **2**(9) (1997)
11. Weber, I., Xu, X., Riveret, R., Governatori, G., Ponomarev, A., Mendling, J.: Untrusted business process monitoring and execution using blockchain. In: La Rosa, M., Loos, P., Pastor, O. (eds.) BPM 2016. LNCS, vol. 9850, pp. 329–347. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_19

Uncover and Assess Rule Adherence Based on Decisions

Johan Silvander(✉) and Mikael Svahnberg

Software Engineering Research Lab Sweden, Blekinge Institute of Technology, Karlskrona, Sweden
{Johan.Silvander,Mikael.Svahnberg}@bth.se

Abstract. Context: Decisions taken by medical practitioners may be based on explicit and implicit rules. By uncovering these rules, a medical practitioner may have the possibility to explain its decisions in a better way, both to itself and to the person which the decision is affecting.
Objective: We investigate if it is possible for a machine learning pipe-line to uncover rules used by medical practitioners, when they decide if a patient could be operated or not. The uncovered rules should have a linguistic meaning.
Method: We are evaluating two different algorithms, one of them is developed by us and named "The membership detection algorithm". The evaluation is done with the help of real-world data provided by a hospital.
Results: The membership detection algorithm has significantly better relevance measure, compared to the second algorithm.
Conclusion: A machine learning pipe-line, based on our algorithm, makes it possibility to give the medical practitioners an understanding, or to question, how decisions have been taken. With the help of the uncovered fuzzy decision algorithm it is possible to test suggested changes to the feature limits.

Keywords: Agglomerative merging · Assess rules adherence
Fuzzy decision making · Shannon entropy · Uncovering rules

1 Introduction

In the area of medicine, laws require medical practitioners to be able to explain their decisions when a patient has complaints. Another vital aspect is the possibility to give the medical practitioners an understanding, or to question, how decisions have been taken. Automated help to uncover how implicit and explicit rules are used by medical practitioners would facilitate the documentation of this decision rationale.

We would like to uncover rules used by a medical practitioner, to enlighten medical practitioners about how the decisions were taken. The algorithms we use should give an explanation to the decision in a human understandable format. Our ideas of finding rules defined in a human understandable format are rooted

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 249–259, 2018.
https://doi.org/10.1007/978-3-319-94214-8_16

in the area of fuzzy sets [16]. The rule values of the features used in a rule logic are expressed in a linguistic way which makes it practical for humans to reason about a feature, for example a value range of the feature “the age of a person” can be defined as “very old”. How well a certain value belongs to a linguistic description of a feature, is defined by mathematical definition, called a membership function.

Normally, the linguistic classifications of a feature are stated by medical practitioners, and the membership functions can be calculated based on this knowledge. In the area of medicine, this approach has been used in several studies (See for example [9]). Instead of defining a rule logic, and its rule values, which shall be used by medical practitioners, we would like to evaluate how a stated rule is used. How well a medical practitioner is following a stated rule can bee seen as a form of trust. How to measure trust is elaborated in several studies, e.g. by Ritu and Jain [10]. These studies explain how to find degrees of trust, but are not aiming to achieve a “strict understanding” according to Pask’s conversation theory [8].

In order to achieve a “strict understanding” we develop a possibility to give the medical practitioners an understanding, or to question, how decisions have been taken, together with a possibility to test suggested changes to the feature limits with the help of the fuzzy decision algorithm. In order to find member functions of the fuzzy decision algorithm, which describe how the stated rules are used, we are evaluating two different algorithms. We use an algorithm commonly used in the area of fuzzy classification, and an algorithm we have named *the membership detection algorithm*. The membership detection algorithm is based on a machine learning algorithm used for feature transformation, in combination with the use of Shannon entropy [13].

The evaluation is done with the help of a real world data set, and a rule logic used by the medical practitioners. The data set contains patient features, and decisions about if an operation should be performed or not. The evaluation showed the membership detection algorithm to be the best choice in order to uncovering rules, and assess rule adherence, based on data in a data set.

The remainder of this article is organized as follows. The methodology is presented in Sect. 2 and the execution results and an analysis thereof are presented in Sects. 3 and 4. The results are further discussed in Sect. 5 and conclusions and future work are presented in Sect. 6.

2 Methodology

We conduct an analysis of a data set containing information about patients, and the decision if an operation should be performed or not. We try to evaluate if it is possible to uncover rules, used by a medical practitioner when deciding if a patient can have an operation or not. This is a binary classification problem but the rules of the classification must be understandable by humans, and it should be possible to make changes to these rules, in the form of changing the rule values. The uncovered rules is constrained by a given rule logic. The given rule logic makes it possible to assess rule adherence based on the decisions.

We were provided with a csv-file consisting of real-world data from a hospital, together with the rule logic used for judgement. The data consists of a class variable and four features; CRP-value[1], age, sex, and weight. The class variable is a binary variable with the decision if an operation should be performed or not. The rule logic is the statement, "If a patient has very high CRP or is very old or has underweight or has overweight then no operation will be performed.".

During our analysis of the data set, the following information were obtained. Table 1 shows two features with missing data, and less than 30% of the class variable values are indicating a positive result from the rule logic, i.e. No operation should be performed.

Table 1. Statistics for the data set.

	CRP	Age	Sex	Weight	No Op
Count	50	55	55	43	55
Mean	28.88	71.22	0.44	68.25	0.29
Min	4.0	29.0	0	42.2	0
Max	137.0	94.0	1	96.0	1

In Table 2 we can see that the correlation between the features and the class variable are not strong. The strongest correlation between a feature and the class variable is 0.46 (the CRP feature). The strongest correlation between the features are between the weight and the sex (−0.40).

Table 2. Correlations for the data set.

	CRP	Age	Sex	Weight	No Op
CRP	1.00	−0.02	0.20	−0.13	0.46
Age	−0.02	1.00	−0.04	−0.15	0.32
Sex	0.20	−0.04	1.00	−0.40	0.16
Weight	−0.13	−0.15	−0.40	1.00	−0.18
No Op	0.46	0.32	0.16	−0.18	1.00

The Shapiro-Wilk test on the CRP feature and the Age feature indicates that a non-parametric correlation test is needed (Table 3). We use Kendall's tau since it is robust for small data sets. The Kendall's tau test show no evidence of correlation between the investigated features and the class variable. There is no evidence of strong correlation between the Weight and the Sex features when we perform a Kendall's tau test. The correlation between the other features are not strong.

[1] C-reactive protein (CRP) is an annular (ring-shaped), pentameric protein found in blood plasma, whose levels rise in response to inflammation [15].

Table 3. Shapiro-Wilk and Kendall measures.

	Shapiro	Kendall's tau
CRP	0.76, 1.30e−07	0.36, 0.0002
Age	0.92, 0.001	0.29, 0.001

Since the data is skewed towards a representation of a negative classification, we decide to duplicate the rows which contains a positive classification. The duplication of the rows gives a data set which contains 45% of positive classifications. The weak correlation between the features, and between the features and the class variable, indicates that we will not introduce any correlation errors when we duplicate the rows with a positive class variable value [2].

Ross [11] describes different algorithms, commonly used in the area of fuzzy sets in order to find membership functions. We decide to use the inductive reasoning algorithm, since it generates membership functions based solely on the data provided [11]. Improvements for this algorithm when handling missing data are suggested in several studies, e.g. Jurado et al. [4]. Since there is a weak correlation between the features, we add a sentinel value for missing data [2], instead of changing the algorithm describe by Ross [11]. With the help of the sentinel value, the remaining information in each sample can be used for creating the membership functions. When the fuzzy decision making rule encounter a sentinel value of a feature, it will assign the feature sample a membership degree of 0.45 to "No operation".

Since finding the feature value limits for the rule logic can be seen as a discretisation problem, we are investigating if a feature transformation algorithm named agglomerative merging [3] can be used to find membership functions. An optimization of the agglomerative merging algorithm have been developed by Li et al. [5]. However, we use the implementation described by Flach [3] since our main concern is to find membership functions which describe rules in a human understandable form.

We have developed an algorithm named *the membership detection algorithm*. This algorithm is based on the agglomerative merging algorithm described by Flach [3], and combined with the use of Shannon entropy [13]. We have augmented the agglomerative algorithm with information about the ranges of a bin, and the rank of the classes in a bin (Pseudocode 1.1). As a post-processing step, we merge a bin at the time, as long as the merged bins is below a merge-entropy limit (Pseudocode 1.2). The bin which caused the entropy to raise above the merge-entropy limit, is examined based on its own entropy. This entropy decides which feature value to be used by the membership function variables. If the entropy is above a value named "value selection entropy" the mean value of the bins feature values is used, otherwise the feature value, adjacent to the merge bin, is used. In order to select the first bin to be merged, we use the knowledge of the rule logic. By using the knowledge of the rule logic, the membership detection algorithm is able to find evidence in the data, which supports the stated rule logic.

```
Agglomerative_merging(featureValueList, scoringFunction, stoppingCriterion)
// featureValueList : A list of feature values, and their associated result value, sorted by the feature value.
// scoringFunction: A function used to evaluate the merging of objectBins (we use chi square values).
// stoppingCriterion: A value used to stop the merging (we use a chi square value).
// binObject: A bin containing a lower feature value, an upper feature value, #positive results, and #negative results.
1. initialise bins to feature values with the same value
2. merge consecutive pure objectBins
3. repeat
4.    evaluate the scoringFunction values on consecutive binObject pairs
5.    merge the objectBin pairs with best scoringFunction value unless stoppingCriterion is invoked
6. until no further merges are possible
7. return a list of binObjects
```

Pseudocode 1.1. The agglomerative merging algorithm

```
Post_processing(binObjectList, mergeEntropyLimit, valueSelectionEntropy, mergeDirection)
// binObjectList : A list of binObjects
// mergeEntropyLimit: If the Shannon entropy is above this limit, no merging shall be performed.
// valueSelectionEntropy: Determine how the feature value of the mergeBin shall be calculated.
// mergeBin: A bin containing a lower feature value, an upper feature value, #positive results, and #negative results.
1. initialise the mergeBin with the starting binObject in the binObjectList.
2. repeat
3. use the next binObject in the binObjectList, as the nextBinObject
4. calculate the Shannon entropy when combining the mergeBin with the nextBinObject
5. update the mergeBin based on the nextBinObject information unless mergeEntropyLimit is reached
6. until the mergeEntropyLimit has been reached
7. calculate the Shannon entropy of the nextBinObject
8. update the mergeBin based on the valueSelectionEntropy of the nextBinObject
9. return mergeBin
```

Pseudocode 1.2. The post-processing algorithm

Feature selection algorithms were considered as well but they do not seem to be useful for the data set, and the problem, at hand. All of the features are part of the rule logic, and the correlation shows no strong indication of a specific feature which dominates the classification.

There are no real-time requirements when uncovering the rules. We have decided to use Python since it have many machine learning contributions, for example Scikit [12]. The implementation is done with the help of Jupyter notebook, from the Anaconda distribution for Mac (x86 64-bits), which supports Python 3.5.2.

3 Results

We have implemented a machine learning pipe-line which makes it possible to uncovering rules, based on decisions made by medical practitioners when deciding if a patient can have an operation or not.

The first part of the pipe-line is used to find the values of the variables, used in the membership functions which indicate "no operation" shall be performed. We use two different approaches. First we use the membership detection algorithm, secondly we use the inductive reasoning algorithm [11]. The values of the variables are shown in Table 4 (column a, b, c, and d).

Table 4 shows how membership functions can be constructed for the combination of an algorithm and a feature. The last three columns tell which of the Eqs. 1–3 is intended to be used. The columns a, b, c, and d, are the variables used in the equations.

Table 4. Information used to create membership functions.

Algorithm	Feature	a	b	c	d	μ_{left}	μ_{mid}	μ_{right}
Membership detection	CRP	-	-	8.0	86.0	-	-	no_operation
	Age	-	-	74.0	89.0	-	-	no_operation
	Weight	47.0	52.8	85.0	85.7	no_operation	-	no_operation
Inductive reasoning	CRP	-	-	34.0	79.0	-	-	no_operation
	Age	-	-	50.5	77.5	-	-	no_operation
	Weight	48.0	54.0	71.8	85.3	no_operation	-	no_operation

In the second part of the pipe-line we construct the membership functions. In order to construct the membership functions, we apply Eqs. 1, 2, and 3 [6], according to the information in Table 4. We use these membership functions to perform fuzzy decision making on the data set.

$$\mu_{left}(x) = \begin{cases} 1 & \text{if } x < a \\ \frac{b-x}{b-a} & \text{if } a \leq x \leq b \\ 0 & \text{if } x > b \end{cases} \tag{1}$$

$$\mu_{mid}(x) = \begin{cases} 0 & \text{if } x < a \\ \frac{x-a}{b-a} & \text{if } a \leq x < b \\ 1 & \text{if } b \leq x < c \text{ and } b \neq c \\ \frac{d-x}{d-c} & \text{if } c \leq x \leq d \\ 0 & \text{if } x > d \end{cases} \tag{2}$$

$$\mu_{right}(x) = \begin{cases} 0 & \text{if } x < c \\ \frac{x-c}{d-c} & \text{if } c \leq x \leq d \\ 1 & \text{if } x > d \end{cases} \quad (3)$$

In the third part of the pipe-line, we use fuzzy logic to evaluate if a patient should have an operation or not. If a patient should have an operation or not, is based on the following rule logic: "If a patient has very high CRP or is very old or has underweight or has overweight then no operation will be performed". We evaluate the logic by using a fuzzy logic "OR", according to Eq. 4, with the criterion "no operation should be performed" if $\mu_{no\_op}(x) > 0.5$.

$$\mu_{no\_op}(x) = max(\mu_{crp\_no\_operation}, \mu_{age\_no\_operation}, \mu_{weight\_no\_operation}) \quad (4)$$

In the last part of the pipe-line we use a confusion matrix to evaluate the measures of the fuzzy decision making, when the different algorithms have been used to find the membership functions. The results are shown in Table 5. In Table 5 the recall shows the proportion of patients which were not given an operation when their health did not allow an operation, and the specificity shows the proportion of the patients which were given an operation when their health did allow an operation. The harmonic mean of precision and recall is given by the F1-measure. In Table 5 we can see that the F1-measure is better for the membership detection algorithm, despite the fact that the inductive reasoning algorithm has a recall of 1. The accuracy is better for the membership detection algorithm, as well. This indicates that the membership detection algorithm uncover rules which have better adherence to the decisions.

Table 5. Measures when using the different algorithms.

Algorithm	Accuracy	Precision	Recall	F1	Specificity
Membership detection	0.7606	0.7027	0.8125	0.7536	0.7179
Inductive reasoning	0.5211	0.4848	1	0.6531	0.1282

4 Analysis

In order to understand the poor measures for the inductive reasoning algorithm (Table 5) we translate the membership functions to crisp thresholds, and calculate the proportions of all samples classified as "no_operation". Table 6 contains the crisp rule values of the features when we use Eq. 4 with the rule "If $\mu_{no\_op}(x) > 0.5$ then no operation should be performed". The columns in Table 6 have the following meanings. If a feature value is $\leq$ low_threshold or $\geq$ high_threshold, no operation should be performed.

In Table 5, the no_operation column shows the proportion of all samples which were classified as "no_operation". The inductive reasoning algorithm has a proportion of 0.8 of "no_operation" for the age feature, but the true proportion of "no_operation" is 0.45. This shows how the inductive reasoning algorithm creates false positives in almost half of the samples. This explains the poor numbers for the inductive reasoning algorithm in Table 5. At the same time, it gives an indication of why no false negatives were found, and therefor the good recall rate of the inductive reasoning algorithm in Table 5.

Table 6. Crips thresholds and classification proportions.

Algorithm	Feature	low_threshold	high_threshold	no_operation
Membership detection	CRP	0	47.0	0.24
	Age	0	81.5	0.23
	Weight	49.7	85.35	0.20
Inductive reasoning	CRP	0	56.5	0.20
	Age	0	64.0	0.80
	Weight	51.0	78.55	0.37

An agglomerative merging algorithm calculates the minimum number of bins (value ranges) a feature has in the data set, based on a chi square value. Since an agglomerative merging algorithm is part of the membership detection algorithm, the number of bins for each feature can be given as an intermediate result of the membership detection algorithm.

The intermediate result of the membership detection algorithm shows that CRP has three bins, Age has seven bins, and Weight has seven bins. When there exists more than three bins for a feature, a class variable is described by more than one value range, and these value ranges are not next to each other. Since in the rule logic, the weight feature has an upper and a lower bound, it can be possible to find two membership functions which correctly describe the rules used for the weight feature. For the age feature it is not possible to find one membership function which correctly describe the rule used for the age feature.

The inductive reasoning algorithm can be seen as a classification algorithm. It expects to find an optimal split of the data based on the classifications, and it expects the starting point of the optimality search to be an average measure of the data. These criteria create a risk for the inductive reasoning algorithm to find local optimum. In the case of the age feature, the inductive reasoning algorithm has forced itself to find a pattern which does not exist in the data set.

The membership detection algorithm is not a classification algorithm. Instead the algorithm tries to find an optimal split in a region of the data set which is appointed by the rule logic. The membership detection algorithm is used to find evidence in the data which supports the stated rule logic. In order to achieve this, the membership detection algorithm uncovers the rule values of the given rule logic and assess the given rule logic's adherence to the decision data.

The numbers in Tables 5 and 6, indicates the use of the membership detection algorithm to be a superior choice when the data set does not contain a distinct classification of the data.

5 Discussion

When decision values of the membership functions (Table 6) are used in the rule logic: "If a patient has a CRP value ≥ high_threshold or has an age ≥ high_threshold or has a weight ≤ low_threshold or has a weight ≥ high_threshold then no operation will be performed", it is easy for the medical practitioners to understand the uncovered rules of the feature values. It is possible to test suggested changes to the feature limits with the help of the fuzzy decision algorithm. However, the main outcome is the possibility to give the medical practitioners an understanding, or to question, how decisions have been taken.

When we make use of the correlation between sex and weight, the model created by the pipe-line using the membership detection algorithm has a accuracy of 0.77, a precision of 0.76, a recall of 0.8, and a F1-measure of 0.78. Even if the numbers improved compared to those in Table 5, the numbers indicate that the model cannot be used as the complete description of when a patient should have an operation or not.

We compare these results with the results from an algorithm which is not limited by the stated rule logic. We use a tree algorithm named CART [1]. The CART algorithm is chosen since it can handle continuous data and a tree is easy for humans to interpret. The result from a stratified 10-fold cross validation with the CART algorithm shows a worst-case below 0.6, and an average just above 0.8, for the different measures. This indicates a lack of features in the provided data set. These features are needed to create rules which completely describe the decisions. Since decisions sometimes are related to beliefs and personal values of the medical practitioners [7], we cannot expect all features, used for the decisions, to be included in the data set. When we consider the limitations imposed by the stated rule logic, the membership detection algorithm performs at the same level as the CART algorithm. We conclude the performance of the membership detection algorithm as very good[2], when uncovering rules. Based on these results we conclude the membership detection algorithm to be a good choice to assess rule adherence base on decisions.

We have used statistical tools and procedures (Sect. 2), to ensure the data set itself, do not impose threats to the validity of this study.

According to Pask's conversation theory [8], one way an actor can create knowledge is to explaining its understanding, based on instructions, and information, given by another actor. The results show that the membership detection algorithm can be used for this purpose. Since Pask's conversation theory is a corner stone in the realization of Intent-Driven Systems [14], the membership detection algorithm can be a vital component in the realization of Intent-Driven Systems.

[2] The membership detection algorithm showed better measures than using the rule values stated by a medical practitioner.

6 Conclusion and Future Work

The analysis shows that the membership detection algorithm has significantly better relevance measure, compared to the inductive reasoning algorithm. The membership detection algorithm extends the possibility to reason about how a deviation from the stated rules will affect the trust of a medical practitioner by making it possibility to give the medical practitioners an understanding, or to question, how decisions have been taken. Another vital extension is the possible to test suggested changes to the feature limits with the help of the uncovered fuzzy decision algorithm.

We believe our developed algorithm is a valuable contribution for uncovering rules, and assess rule adherence, based on data in a data set. In our future work we will continue to investigate how to create knowledge, and understanding, in information-poor environments.

References

1. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth & Brooks, Monterey (1984)
2. Djurfeldt, G., Barmark, M. (eds.): Statistikverktygslåda - multivariat analys. Studentlitteratur (2009)
3. Flach, P.: Machine Learning. Cambridge University Press, Cambridge (2012)
4. Jurado, S., Nebot, À., Mugica, F., Mihaylov, M.: Fuzzy inductive reasoning forecasting strategies able to cope with missing data: a smart grid application. Appl. Soft Comput. J. **51**, 225–238 (2017)
5. Li, Y., Le, J., Wang, M.: Improving CLOPE's profit value and stability with an optimized agglomerative approach. Algorithms **8**(3), 380–394 (2015)
6. Mendel, J.M.: Uncertain Rule-Based Fuzzy Systems, 2nd edn. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-51370-6
7. Nardi, R., Fabbri, T., Belmonte, G., Leandri, P., Mazzetti, M., Pasquale, A., Reta, M., Rizzi, C., Scanelli, G., Iori, I., Gussoni, G., Pedace, C., Mathieu, G., Mazzone, A.: Medicina interna, paziente complesso, evidence based medicine e le non evidenze, December 2009
8. Pask, G.: Conversation Theory. Elsevier, Amsterdam (1976)
9. Rakus-Andersson, E.: Advanced Computational Intelligence Paradigms in Healthcare 3. Springer, Heidelberg (2008)
10. Ritu, Jain, S.: A trust model in cloud computing based on fuzzy logic. In: IEEE Conference on Recent Trends in Electronics Information Communication Technology, pp. 47–52 (2016)
11. Ross, T.: Fuzzy Logic with Engineering Applications, 4th edn. Wiley, Hoboken (2017)
12. Scikit-learn.org: Scikit Learn (2016). http://scikit-learn.org/stable/index.html. Accessed 19 Feb 2018
13. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**(3), 379–423 (1948)
14. Silvander, J., Wilson, M., Wnuk, K., Svahnberg, M.: Supporting continuous changes to business intents. Int. J. Softw. Eng. Knowl. Eng. **27**(8), 1167–1198 (2017)

15. Wikipedia: C-Reactive Protein (2018). https://en.wikipedia.org/wiki/C-reactive_protein. Accessed 19 Feb 2018
16. Zimmerman, H.J.: Fuzzy Set Theory - And Its Applications, 4th edn. Springer, New York (2001)

Towards the Component-Based Approach for Evaluating Process Diagram Complexity

Jernej Huber(✉), Gregor Polančič, Mateja Kocbek, and Gregor Jošt

Faculty of Electrical Engineering and Computer Science, University of Maribor, Koroška cesta 46, 2000 Maribor, Slovenia
jernej.huber@um.si

Abstract. Various authors have defined an extensive set of measures regarding the complexity of process design, which can be objectively measured by using structural complexity metrics. Nevertheless, the research in this area indicates that the percentage of empirically and theoretically validated metrics is relatively small. This suggests that there are still no real examples of the metrics usage within organizations. Despite that we could just validate existing proposals, we feel that a new and enhanced approach to evaluate process diagram complexity is feasible and needed. Thus, in the present paper, we will discuss a possibility of developing a new component-based approach for evaluating process diagram complexity, for which we anticipate that it would be more precise than existing ones, since it will also assess such constituent parts of process diagrams which not only contribute to higher complexity (e.g. complex routing behavior), but also to lower complexity (e.g. decomposition). Moreover, we plan to thoroughly validate our approach both in theory and practice.

Keywords: BPMN · Complexity · Metrics · Decomposition
Workflow patterns

1 Introduction

Business process diagrams (hereinafter referred to as BPDs) are used to make the communication between stakeholders easier, however, in practice, they are commonly complex. Therefore, it is essential to address the measurement and management of their complexity. This is especially true, since complexity affects the time and effort for efficient understanding and maintenance of a BPD. Moreover, business process analysts can model different BPDs for the same problem domain, so the usage of less complex BPDs will cause less error when modeling or reading such diagrams.

Recent research has assessed process diagram comprehension by measuring different aspects of such diagrams. Factors with impact on comprehension include diagram characteristics (among them structural complexity, which is focus or this paper), modeling language characteristics and personal characteristics of a diagram reader [1].

BPDs can be modeled in several notations. The most widely used notation is Business process model and notation (BPMN), which has been identified as the de-facto standard in the BPM domain [2].

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 260–269, 2018.
https://doi.org/10.1007/978-3-319-94214-8_17

Moody [3] introduced the physics of notation theory, which consists of nine design aspects of modeling languages in order to prevent unnecessary extraneous cognitive load on the user of a BPD: semiotic clarity, graphic economy, perceptual discriminability, visual expressiveness, dual coding, semantic transparency, cognitive fit, cognitive integration and complexity management. The latter deals with ensuring that there is the possibility for decomposition and hierarchical structuring in case of the complex diagrams.

Cognitive research of the business process model comprehension has identified the phenomenon when two formal structures of one process model are the same, but there is a notable difference in the visual presentation. In this light, the modeling notation as a formal set of symbols is defined as the primary notation [4]. On the other hand, various visual cues (e.g. color), which are not part of a notation, are defined as the secondary notation. The results of a study performed by Petrusel et al. [5] show a clear positive impact of these visual cueing techniques on the ability of the study's participants to provide correct answers in a timely manner, as well as on the mental effort required to read the process models.

Regarding the process modeling language syntax, the concrete syntax deals only with representational aspects such as symbols, colors and position of the different types of nodes in a BPD (e.g., tasks, routing symbols, roles) [6]. In cognitive sciences, this corresponds to the cognitive dimension of secondary notation. On the other hand, the abstract syntax of a process model is describing the formal structure of BPD elements and their mutual relationships [6]. As such, it corresponds to the deep structure of a sentence, an underlying syntactic structure, as opposed to the concrete syntax, which corresponds to the surface form of a sentence, which is its visual presentation.

There are many complexity-coping mechanisms for making BPDs easier to understand, e.g., (1) good and bad practices for modeling [7], (2) patterns for reducing abstract complexity of diagram, related to inner structure and flow logic [6] and (3) patterns for reducing concrete complexity of diagrams, related to visual elements – secondary notation [8].

Decomposition, also known as modularization or fragmentation, is another important complexity-coping mechanism, which falls into the category of secondary notation. Authors in [9] report that "a decomposed process is more intuitive and easier to understand as each decomposition step incrementally focuses on a smaller number of overlapping concerns". This allows higher BPD reuse and increases the ability for successful BPD communication and analysis. Modelers must ensure that each decomposition step provides a consistent level of details, so that the resulting set of basic activities which comprise the bottom level of decomposition is the same.

In light of decomposing BPDs, authors [10] proposed a set of 20 workflow control-flow patterns, which provide a common ground for describing a various notation independent workflow routing implementations. This was reasonable since the distinctive features of different business process notations result in fundamentally different semantics.

Within systematic literature review (SLR), Figl [11] proposed the following influence factors for BPD comprehension: presentation medium (computer versus paper), notation (chosen modeling language, physics of notation, representational paradigm), secondary notation features (decomposition, highlighting, layout), labeling (naming conventions, label design), model characteristics (structural size measures, syntax rules, refactoring, etc.), task characteristics (type of the task, which user of a BPD must to solve) and user characteristics (modeling knowledge, education, domain knowledge).

Similarly, Dikici et al. [12] performed a SLR in which they identified factors which significantly influence the BPD comprehensibility, among them notation, structural complexity, modeling expertise, modularity, process perspective representation, modeling approach, visual layout, element labeling, use of modeling guidelines, cognitive abilities, learning motive, style, use of coloring and model element design.

Most of existing structural complexity metrics only consider up to two constructs (e.g. activities or control flows) for evaluation of the complexity of a BPD [13]. Best solution for measuring the overall complexity of BPDs seems to be the usage of a set of metrics, where each of the metrics addresses one of the constructs [13, 14]. Additionally, only approximately 3% of the metrics were validated in theory and practice and approximately 56% have been validated only empirically [13]. These results were also independently confirmed by Marin [15].

In this paper, we will focus on the structural complexity, which is one of the important influential factors of the BPD comprehensibility. Additionally, in the foreseen approach, we will limit our research proposal to the above-mentioned BPMN notation, since it includes the decomposition mechanisms. Moreover, we found no metrics which would consider the aspect of concrete syntax, i.e., the secondary notation of process modeling languages, such as the use of colors. In addition, metrics within the current known set, which could be applied to BPMN, do not consider the concept of decomposition, i.e., the usage of pools, lanes and message flows in their calculations.

The remaining part of the article is structured as follows: in the related work section, we will outline the state of the art in the field of structural complexity metrics area and the approaches on how to define new metric. In the third section, we will present the proposed solution – the component-based approach for measuring the structural complexity of BPDs. Within the discussion section, we will expose some discussion points for the possible directions of our research. In the last section, we will provide the contribution statement of the paper.

2 Related Work

In the literature, we can find many proposals for measuring the BPD complexity. They measure different aspects of complexity, e.g., complexity of control flow or structure, modular and cognitive complexity.

In the recent SLR [13], authors overviewed 66 metrics, out of which 53% metric were related to business process management domain and 23% to the software engineering, while others remained unclassified. These findings were confirmed by another SLR, performed in 2017 within a doctoral thesis by Marin [15].

Since complexity metrics provide a numerical value, which is not useful by itself, they should at least allow a comparison with other values or be classified into specific metric intervals, separated by thresholds. There are different techniques for such threshold identification, e.g., ROC curves and Bender method [16].

In regard to the assessment of the overall quality of BPDs, authors [17] identified a homogeneous set of understandability guidelines and a tool to check their validation. To this end, they conducted an extensive review of existing guidelines and assessed the suitable metrics and numeric thresholds, that allow the adherence to proposed guidelines.

Our proposal is similar to the idea of cognitive complexity measure, which was proposed by Gruhn and Laue [14]. Their proposal was never validated neither empirically nor in practice and it has many limitations, for example fixed weights for each BPD control structure (see Fig. 1).

BPD control structure	Weight
sequence of activities	1
XOR-split	2
XOR-split with more than 3 branches	3
AND-split and AND-join	4
OR-split	7
Composite task (subtask)	2
Multiple instances	6
Cancel activity	1
Cancel case	2 or 3

Fig. 1. Cognitive weights for BPD elements.

We feel that extending their underlying idea with our component-based approach will evaluate the complexity of BPDs in a more detailed and holistic manner. Therefore, we wish to obtain a broader insight into the BPD structure and evaluate it with empirically and theoretically validated values, which come with thresholds for each constituent part of the BPD. The more detailed information is presented in the third chapter.

Regarding the process of defining new metric, we found a consensus among authors [18–20], who defined steps, which are required to properly validate a new proposed metrics, namely: (1) metric definition, (2) theoretical validation and (3) empirical validation. Moreover, researchers in [20] proposed the following three sequential steps: (1) measurement entity identification (e.g. BPD), (2) entity attribute identification (e.g. number of nodes within the BPD) and (3) identification of a new metric, which allows the measurement of all the proposed attributes. There have been many methods proposed for theoretical validation (e.g. Weyuker properties [21], Briand's framework [22]). Similarly, for the empirical validation, there are also many methods available [18], e.g. survey, experiment, case study.

In [23], authors proposed the properties, which all properly defined metrics should encompass, i.e., validity, reliability, computability, ease of implementation, intuitiveness, and independence of other related measures. Considering the graph theory, it is possible to extend the proposed set of properties with ability to measure the complexity of iterative processes, modularity, additivity and independence of the level of detail in modeling.

3 Proposed Solution

The final long-term objective of our proposed approach is the development of a prototype tool, which would allow to import a BPD and calculate the corresponding component-based cognitive complexity with better precision compared to existing metrics.

The underlying idea towards implementing our approach (as seen from the Fig. 2) consists from the following steps: (1) we will evaluate the structural complexity of a chosen BPD with the use of existing metrics, (2) we will identify the components (hereinafter referred to as constituent parts) within a BPD, (3) we will assign our empirical data regarding cognitive complexity of constituent parts, (4) we will combine the values of structural complexity and refine the overall structural complexity with our data.

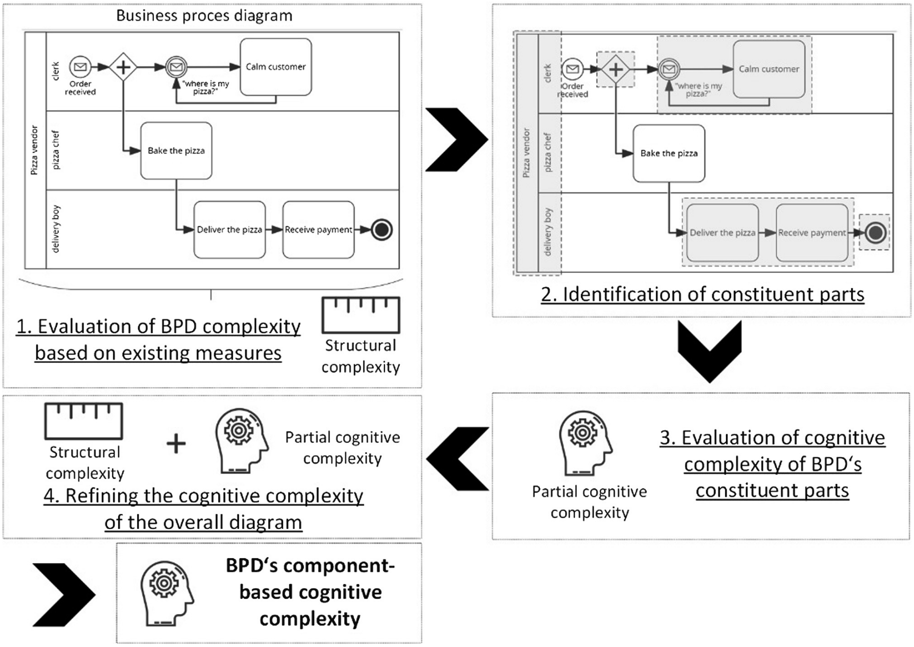


Fig. 2. Steps of the proposed approach

Aforementioned constituent parts (see Fig. 3) of a BPD can be either workflow patterns, such as sequence, explicit termination, splitting (e.g. parallel split) and looping (arbitrary cycles) mechanisms, decomposition related elements (e.g. the usage of grouping elements, subprocesses, lanes, etc.) and other elements of a BPD, which affect the complexity, either negatively or positively.

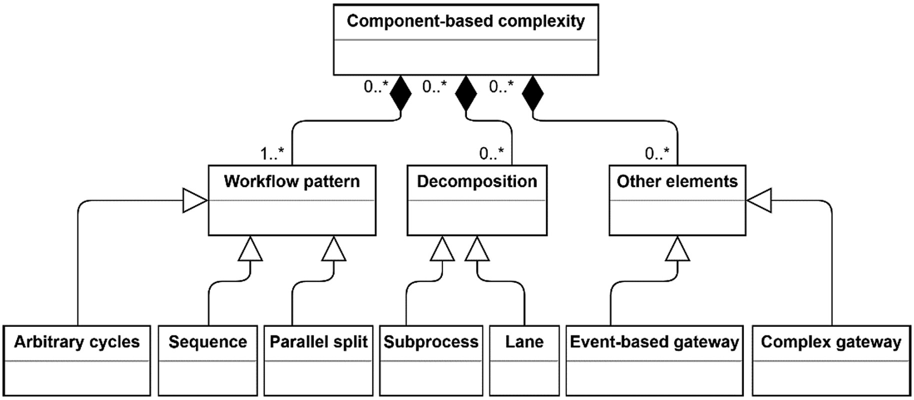


Fig. 3. Class diagram of the proposed approach

A proposed component-based complexity measure could be defined with the following equation (where variable n presents a total number of constituent parts and f(x) presents the complexity of each constituent part of a BPD):

$$C_{cb} = \sum\nolimits_{i=0}^{n} f(x) \tag{1}$$

Each constituent part of a BPD will have to be defined more specifically. As an example, we could define an equation for describing the complexity of a sequence workflow pattern as follows (where variable x presents number of elements within a single constituent part, whereas 15 and 30 are examples of threshold values, which can distinguish different levels of BPD complexity):

$$f(x) = \begin{cases} 1 & x < 15 \\ 5 & 30 > x > 15 \\ 5 + \frac{x}{x+1} & x \geq 30 \end{cases} \tag{2}$$

For each constituent part, such equations will have to be obtained empirically during the process of defining a new metric. In this sense, a structured questionnaire with specific understandability tasks within different categories will be defined (e.g., concurrency, exclusiveness, order and repetition). For example, finding one activity within a sequence of any size is trivial task. On the other hand, answering the question regarding which specific activity is performed first among a set of randomly chosen activities in a sequence (i.e. finding out if activity “concentrate” from the random set of activities is performed after activities “invent”, “trace” and “detect”), can have bigger cognitive impact to the reader of a BPD, if a sequence of activities is larger, especially if the activities within the sequence are not unique.

Among constituent parts, we will also include those, which are improving cognitive complexity (e.g. the usage of lanes or other component-based elements). In this light, it is worth mentioning again that the existing metrics found in the literature, which could

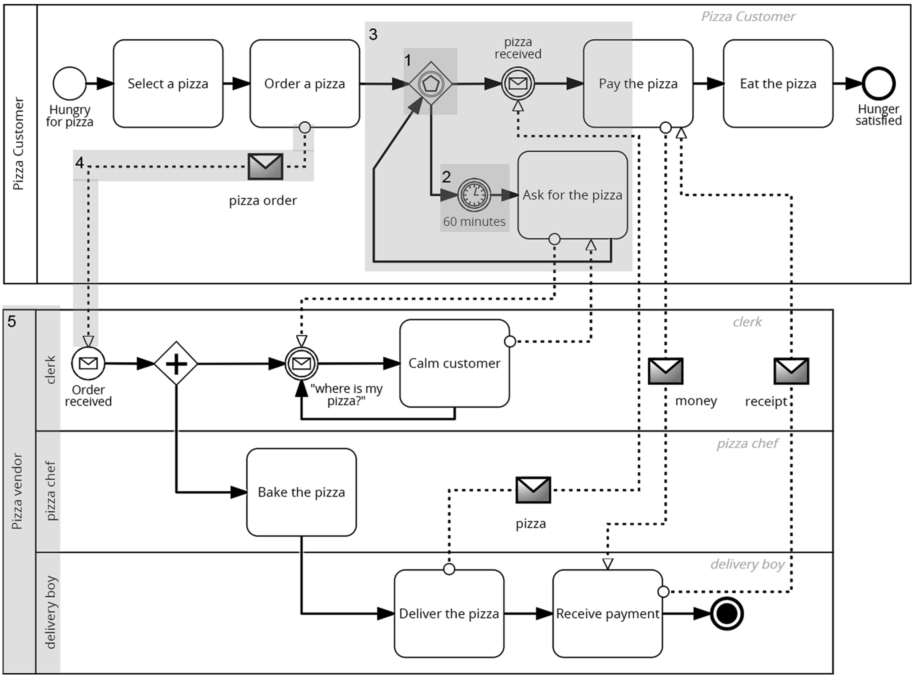


Fig. 4. Example BPMN diagram, taken from [24].

be applied to BPMN, do not consider pools, lanes and message flows in their calculations, so we will have to prepare the basis for calculating this aspect of the BPDs.

From the BPD in Fig. 4, we can extract a set of constituent parts, which are colored gray and numbered as follows: (1) event-based gateway sequence, (2) delay, (3) loop, (4) sending messages between independent actors and (5) decomposition. Each of the pattern possesses its own complexity. For example, displaying two lanes within the pool "Pizza vendor" contributes in lowering the overall complexity of the diagram, while event-based gateway mechanism in combination with a loop in "Pizza customer" pool contributes to higher overall complexity.

Therefore, we will perform experiments for obtaining the empirical data, which will allow us to assess the complexity of each constituent part of a BPD.

According to the proposed approach, we developed two presumptions:

- we presume that by considering the empirically gained cognitive complexity of constituent parts (and their complexity change thresholds as well) of a BPD, the proposed metric will be more precise as compared to the existing ones,
- we presume that by considering the constituent parts, which are lowering cognitive complexity, the proposed metric will be more precise as compared to the existing ones.

We will limit our research for identifying the predefined set of constituent parts to the BPMN modeling language.

4 Discussion

While this paper is merely just an outline, a roadmap and a discussion of a feasibility of a new approach on how to measure complexity of business process diagrams, we identified few discussion points.

- *The identification of the elementary constituent parts.* We propose workflow patterns as the most appropriate elementary constituent part. Nevertheless, we should not limit our choice just to these patterns and should include various concepts such as delays, event-based gateway, pools, lanes, regions, sub-processes, etc., which are not considered by the current metrics.
- *The measurement of the elementary constituent part complexity.* We propose to obtain the complexity values of each constituent part with the conduction of series of experiments with users solving various, but systematically assigned, understandability tasks within different categories.
- *The management of elementary constituent part variability.* Because each constituent part could possess variability regarding the complexity (e.g., sequence or parallel execution and synchronization can be done over a different number of activities), we will have to address this issue by obtaining the empirical data for different thresholds of complexity for each constituent part.
- *The inter-connectedness between constituent parts.* In the case, where the usage of two constituent parts is interwoven (the usage of event-based gateway and a loop), we must also address the composite complexity of such cases.
- *The calculation of the overall BPD complexity.* The overall BPD complexity will be calculated as a sum of complexity of each constituent part of a BPD. It should be noted that some constituent parts could negatively impact the overall BPD complexity (e.g. modularization and sub-processes).

5 Conclusion

The idea of measuring the complexity of BPD is not new. Researchers have proposed an extensive set of metrics and existing literature reviews suggest that those metrics are for the most part not validated neither empirically nor theoretically. As such, they do not appear to be used in practice as much as would be sensible. Since our component-based approach for assessing the BPD complexity will include such components of process diagrams, which also contribute to lowering the complexity (e.g. decomposition), we feel that it has the long-term potential to positively contribute to a much-needed wider adoption of the complexity metrics within the BPM community. In addition, the current metrics, which could be applied to BPMN, do not consider pools, lanes and message flows in their calculations – our approach will also address these components.

The main contribution of the paper is an outline of a brand-new approach on how to measure the structural complexity of the BPDs. However, we will like to stress that this is the work in progress and merely the first step towards establishing our proposed approach, thus we still must address various issues, such as the identification of the most appropriate constituent parts for measuring complexity, evaluation of their variability thresholds of complexity and the measurement of such variability.

References

1. Petrusel, R., Mendling, J., Reijers, H.A.: How visual cognition influences process model comprehension, Decis. Support Syst. **96**, 1–16 (2017). http://www.sciencedirect.com/science/article/pii/S016792361730012X
2. Geiger, M., Harrer, S., Lenhard, J., Wirtz, G.: BPMN 2.0: the state of support and implementation, Futur. Gener. Comput. Syst. **80**, 250–262 (2018). http://www.sciencedirect.com/science/article/pii/S0167739X17300250
3. Moody, D.: The Physics of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. IEEE Trans. Softw. Eng. **35**(6), 756–779 (2009)
4. Syntax highlighting in business process models. Decis. Support Syst. **51**(3), 339–349 (2011). http://www.sciencedirect.com/science/article/pii/S0167923611000042
5. Petrusel, R. Mendling, J. Reijers, H.A.: Task-specific visual cues for improving process model understanding. Inf. Softw. Technol. **79**, 63–78 (2016). http://www.sciencedirect.com/science/article/pii/S0950584916301173
6. La Rosa, M., Wohed, P., Mendling, J., ter Hofstede, A.H.M., Reijers, H.A., van der Aalst, W.M.P.: Managing process model complexity via abstract syntax modifications. IEEE Trans. Ind. Informatics **7**(4), 614–629 (2011)
7. Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG), Inf. Softw. Technol. **52**(2), 127–136 (2010). http://www.sciencedirect.com/science/article/pii/S0950584909001268
8. La Rosa, M., ter Hofstede, A.H.M., Wohed, P., Reijers, H.A., Mendling, J., van der Aalst, W.M.P.: Managing process model complexity via concrete syntax modifications. IEEE Trans. Ind. Inform. **7**(2), 255–265 (2011)
9. Caetano, A., Silva, A., Tribolet, J.: Business process decomposition - an approach based on the principle of separation of concerns. Enterp. Model. Inf. Syst. Archit. **5**, 44–57 (2010)
10. van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., Barros, A.P.: Workflow patterns. Distrib. Parallel Databases, **14**(1), 5–51, July 2003. https://doi.org/10.1023/A:1022883727209
11. Figl, K.: Comprehension of procedural visual business process models. Bus. Inf. Syst. Eng. **59**(1), 41–67 (2017). https://doi.org/10.1007/s12599-016-0460-2
12. Dikici, A., Turetken, O., Demirors, O.: Factors influencing the understandability of process models: a systematic literature review, Inf. Softw. Technol. **93**, 112–129 (2018). http://www.sciencedirect.com/science/article/pii/S0950584916302889
13. Polančič, G., Cegnar, B.: Complexity metrics for process models – a systematic literature review. Comput. Stand. Interfaces, **51**, 104–117, March 2017. http://linkinghub.elsevier.com/retrieve/pii/S0920548916302276
14. Gruhn, V., Laue, R.: Adopting the cognitive complexity measure for business process models. In: 2006 5th IEEE International Conference on Cognitive Informatics, vol. 1, pp. 236–241 (2006)
15. Marin, M.A.: Exploring complexity metrics for artifact-centric business process models. University of South Africa, Pretoria (2017). http://hdl.handle.net/10500/23179
16. Sánchez-González, L., García, F., Ruiz, F., Mendling, J.: A study of the effectiveness of two threshold definition techniques. In: 16th International Conference on Evaluation Assessment in Software Engineering (EASE 2012), pp. 197–205 (2012)

17. Corradini, F.: A Guidelines framework for understandable BPMN models, Data Knowl. Eng. **113**, 129–154, January. 2018. http://linkinghub.elsevier.com/retrieve/pii/S0169023X1630341X
18. Muketha, G.M., Ghani, A.A.A., Selamat, M.H., Atan, R.: A survey of business process complexity metrics. Inf. Technol. J. **9**, 1336–1344 (2010)
19. Rolón, E., Cardoso, J., García, F., Ruiz, F., Piattini, M.: Analysis and validation of control-flow complexity measures with BPMN process models. In: Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., Ukor, R. (eds.) BPMDS/EMMSAD -2009. LNBIP, vol. 29, pp. 58–70. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01862-6_6
20. Fenton, N.E., Pfleeger, S.L.: Software Metrics: A Rigorous and Practical Approach, 2nd edn. PWS Publishing Co., Boston (1998)
21. Weyuker, E.J.: Evaluating software complexity measures. IEEE Trans. Softw. Eng. **14**(9), 1357–1365 (1988)
22. Briand, L.C., Morasca, S. Basili, V.R.: An operational process for goal-driven definition of measures, IEEE Trans. Softw. Eng. **28**(12), 1106–1125 (2002). http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1158285
23. Latva-Koivisto, A.M.: Finding a complexity measure for business process models. Helsinki Univ. Technol. Syst. Anal. Lab., pp. 1–26 (2001). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.25.2991&rep=rep1&type=pdf
24. OMG: BPMN 2.0 by Example: non-normative OMG document with BPMN 2.0 examples (2010). http://www.omg.org/cgi-bin/doc?dtc/10-06-02

Differences Between BPM and ACM Models for Process Execution

Alexander Adensamer(✉) and David Rueckel

WBS Akademie, Lorenzweg 5, 12099 Berlin, Germany
alexander.adensamer@emini.at

Abstract. As the demand for the modeling of knowledge intensive processes grows, so does the necessity to support this task by appropriate models. Adaptive Case Management (ACM) was proposed as appropriate, whereas still confined mostly to theoretical work. This paper compares the execution of BPM process models to ACM models by applying agent based simulations to different processes based on support case handling. The simulation tools applied to the ACM process models were developed in the course of this work. The greater flexibility of ACM models is found to be more effective in processing executions with competent knowledge workers. Another key observation is the possible improvement of average case duration due to parallelization effects.

Keywords: Business process simulation · Business process management
Adaptive case management · Modeling

1 Introduction

Business Process Management is a common topic of constant relevance for any organization. Key capabilities of any organization are performed and maintained as processes. A process represents a set of tasks and activities to be performed to reach a goal. In order to increase the maturity of processes an organization has to understand them, be able to create process models and maintain these models.

Several approaches and frameworks were developed to support organizations and the responsible individuals. The most popular approach is the Business Process Model and Notation (BPMN) [1]. BPMN is a graphical notation that supports the definition of process workflows for process modeling, testing and optimization.

BPMN takes an imperative approach to process modeling. The workflow itself is directly represented in the model. It indicates all possible states that a process instance (case, case folder) can assume.

As process models increase in size due to the increase of the scope of the modelled process itself, it becomes increasingly hard to maintain them. This happens in particular for knowledge intensive processes when multiple possible states are considered [2]. In practice the need for further refinement leads either to huge process models that become unmaintainable or to simple models not covering the actual process in sufficient detail. This furthermore leads to a lack of flexibility. The process itself can be considered a complex system that is non-trivial to understand and maintain. [3–5] have

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 270–279, 2018.
https://doi.org/10.1007/978-3-319-94214-8_18

identified similar challenges for process modeling using BPMN or similar imperative modeling notations.

Swenson proposed Adaptive Case Management (ACM) as an alternative approach to react to mentioned issues [6]. ACM is based on observation that a knowledge worker executing a process often knows best what to do in a given situation without explicitly being told by the process. As the importance of knowledge work is increasing, the importance of flexible process models is increasing [7]. This represents an increase in flexibility at the loss of control of the process execution.

In ACM the perspective of the process is on the case and its outcome. Instead of explicitly modeling the workflow imperatively, conditions and goals are defined for activities' execution. The conditions ensure that cases always remain compliant. Goals ensure that process execution leads to the desired outcome. Based on these definitions the workflow emerges depending on the constraints imposed by the case and the knowledge worker executing it. This allows for ad-hoc activities and tasks while executing a certain process instance (a case). The process definitions are based on templates of activities and tasks [8].

ACM is still not widely used. Various research questions remain unanswered [9] and so far there is no standard for modeling ACM.

While BPM strictly speaking covers both imperative and declarative approaches due to dominance of BPMN typically imperative models are associated with Business Process Management. Mentioning of BPM within this text refers to imperative models in contrast to declarative models which are represented by ACM.

2 Research Goal and Methodology

The aim of this work is to gain deeper insight into the usage of ACM for knowledge intensive processes. This work focuses on the determination of differences that process models show while executing various processes. There is few prior work on this topic except a small number of case studies [10]. The following research question is proposed:

What is the effect of using ACM to model and run processes in comparison to BPM on the efficiency for knowledge intensive processes?

This question will be answered by simulating three different processes. The processes are modelled both in an imperative and declarative manner and results are compared. The analyzed scenarios contain different aspects prevalent in knowledge intensive processes. The simulation experiments for the BPMN models are executed using ARIS Simulation, part of ARIS Toolset [11]. The ACM models are simulated by an extension of the ACM process model supplied by dcrgraphs.net created for this work.

The following measurements are used to compare simulation results as they show expressive power regarding the effectiveness of the process:

- Open Cases: Number of open cases after simulation period. This indicates a lack of resources causing build-up of cases.
- Closed Cases: Number of cases closed within simulation period.

- Utilization of worker pool: Percentage that indicates the average load on each worker pool.
- Average case processing time: Average time for a case from initial triggering until completion. Note that with task parallelization the processing time can even be smaller than total time spent on working on case.
- Time to customer feedback: Time between the initial submit of a request and the answer the customer gets, which can vary from case duration.

In section Literature Review we point to prior work on BPM and ACM modeling. In section Empirical Study we will apply the simulation to the selected process models. We discuss results of the simulations and suggest future research in the final section Conclusion and Future Research.

3 Literature Review

In the current organizational environment business processes are modelled as workflows using BPMN or a similar notation. The way of modeling a business process in declarative manner represents a paradigmatic shift.

Imperative modeling focuses on how a process is executed. The process designer creates the workflow diagram and has the sequence of process states in mind when creating the model. In contrast to this declarative modeling focuses on the aim of the process. The process designer defines goals and the individual activities with their contributions to that goal in mind.

It is shown that imperative models are easier to use when processing sequential information and declarative models are superior when processing circumstantial information [12]. However, there is currently a strong bias towards imperative models. These models are already well known and understood by the people working in this area. In contrast, declarative models still are a novelty and have not yet reached mainstream publicity.

Imperative models have tendency to be over-designed in order to cover all possible situations. This causes frequent changes which can be avoided by using a declarative approach to process modeling [3].

One approach to model ACM processes was developed by the University of Copenhagen is called Decision Condition Response Graphs (DCR Graphs) [13, 14]. DCR Graphs define a declarative workflow modeling notation. The process models consist of events that represent goals, the completion of activities or external events. No sequential workflow is defined. Instead, a set of conditions on the events ensure the correctness of the execution order. The basis for DCR is Linear Temporal Logic, which supplies the foundation for formulating event based graphs. This approach was initially suggested by Pesic and van der Aalst [3]. Each event is assigned to one or more roles to link the model to the organizational structure. Hildebrand shows that DCR Graphs can be used to model ACM processes [15].

4 Empirical Study

Simulation of business processes is done by agent based discrete event simulation [11]. During a defined simulation time range, instances of the process are created and worked on by the simulation agents. These Agents represent the case workers in the context of the process. One of the main aspects of ACM is the transfer of workflow decisions to the case worker. The behavior shown by these workers is represented by different agent implementations: *Default Agent* that picks the oldest available task; *Random Agent* that picks a random task of all available tasks; *Competent Agent* that picks the ideal task with a certain percentage of correctness; and a *Preference Agent* with preferred tasks that are always executed over other types of tasks.

4.1 Support Process

The initial scenario that is simulated represents a typical support process acting as a baseline for the simulation environments.

The process covers three worker pools: Support Level 1 (SL 1), Support Level 2 (SL 2) and Development. A support request by the customer initiates the process. Three sequential tasks are executed by Support Level 1: "Ticket Creation", "Case Categorization" and "Knowledgebase Check". Based on the result of this check either the "Customer Feedback" is performed directly (70% probability in simulation) or the case is further handled by the other worker pools performing "Reproduction" (SL 2), "Update of the Knowledge Database" (SL 2) and "Detailed Problem Analysis" (Development). In some cases "Additional Information" task has to be carried out by Support Level 1 (40% probability in simulation).

The corresponding DCR Graphs model consists of the same tasks, but it leaves flexibility for the order of task execution. For example the "Additional Information" task can be done at any point before the actual analysis is done by development.

For the simulations SL 1 has 6 workers, SL 2 has 2 workers and 1 worker forms the development.

Table 1. Simulation results for support process models

Parameters								
#Cases	6/day		12/day		14/day		16/day	
Results	BPM	ACM	BPM	ACM	BPM	ACM	BPM	ACM
Open cases	6	2	11	6	28	33	80	105
Closed cases	594	598	1189	1194	1372	1367	1520	1495
Util. SL 1	36.1%	36.2%	72.9%	73.2%	84.8%	85.1%	96.4%	96.9%
Util. SL 2	28.7%	28.9%	59.7%	60.1%	68.2%	68.3%	75.1%	75.7%
Util. dev	43.0%	43.3%	88.8%	88.5%	98.1%	97.5%	98.9%	99.2%
Avg. case duration	13.7 h	10.9 h	16.1 h	14.9 h	30.8 h	30.9 h	62.5 h	67.0 h

The results in Table 1 show similar utilization levels and amounts of closed cases indicating the same amount of work to be done in both cases. The average case durations are lower for the ACM cases as long as the amount of cases is within the capacity of the worker pool. This is due to parallelization effects that are directly caused by higher flexibility of the declarative model.

Note that beyond 14 cases/day a high increase of case durations is observed. This is caused by exceeding the capacity of the worker pools leading to a build-up of cases waiting for execution. In general random effects average out during the simulation as enough capacity is available to deal with extra work for short periods. When the capacity is exceeded, however, this does not hold true anymore. Small build-ups of work can have big impact on the end result. Thus the results beyond this level are relevant mainly in the order of magnitude.

4.2 Ad-Hoc Decision

The second scenario focuses on the impact ad-hoc decisions can have on case execution, as this is a core concept of ACM processes. Let us assume a situation where the knowledge worker can take such an ad-hoc decision between two paths of action without an explicit rule defined by the process. The correctness of the decision is only known after the task is performed. In case of an incorrect choice the alternative path has to be taken as well. An example of such a scenario would be two variants of a single activity where one takes significantly less time to complete, but is only valid under certain circumstances which are not known in advance. Encoded in typical BPMN processes is the execution of the longer task that is always sufficient. This misses potential improvement of the process. The optimum can be reached by applying this to an ACM process under the assumption that knowledge workers can choose the correct path intuitively.

The reference BPM model represents a very simple process consisting of three sequential tasks: “Analysis”, “Research in Knowledge DB” and “Response to Customer”. All three tasks are performed within the same pool of resources.

The ACM model is based on the same linear workflow as the reference model. However, for each of these tasks a simplified alternative exists that can be chosen. Only after completion of the task it is known whether the chosen path was indeed correct. Since the simulation does not support information-based criteria, a slightly more complex variation of this process is used for simulation. This exposes the criteria for each of the three tasks to the simulation. It is modelled as an external event for which the simulation can decide upon case creation whether the simplified version of that task is possible. The probability for each of the tasks is part of the parameterization of the simulation runs.

Another important factor of the models is the difference of average duration between the simplified and the full task (cost structure) described in Table 2.

Agents with different competence levels are used to represent differently skilled knowledge workers. The levels of competence determines the probability of a correct decision between two versions of a task within the simulation. The simulation experiments are done with 2 cases per day and a single worker so that the resulting effects are clearly separated from parallelization effects.

Table 2. Average times of tasks

Task	Avg. time	Simplified task	Simplified low cost	Simplified high cost
Analysis	4 h	2 h	1 h	3 h
Research knowledge DB	3 h	2 h	1 h	2 h
Respond to customer	2 h	1 h	1 h	1 h

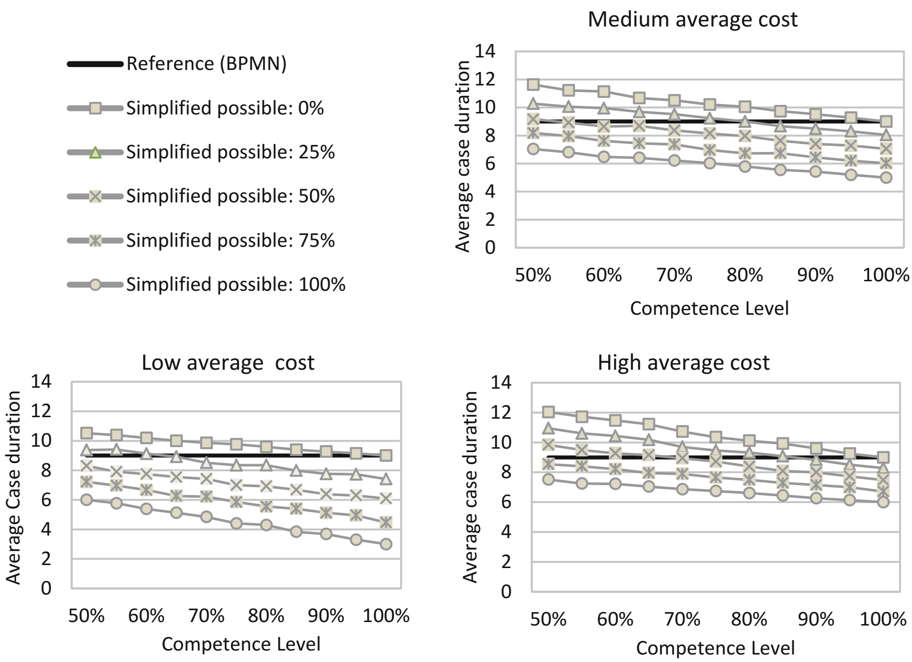


Fig. 1. Simulation results for ad-hoc decision

Figure 1 shows that the break-even points where the average case duration is improved compared to the reference BPMN model depends both on the competence of the case workers as well as the number of cases where the simplified tasks are beneficial. The relation between the competence of the case worker and the case duration is linear. The slope of the curves is correlated to the cost structure. This leads to a reinforcement effect: In the low cost case the penalty of a wrong decision is smaller as well as the gain in case of a good decision is higher. That means that the break-even happens at a lower competence of the case workers as well as for cases where the simplified version is only occasionally available.

4.3 Sub-processes

Sub-processes are another difference between imperative and declarative processes which can have an impact on process execution. Rather than a strict workflow within a closed unit a sub-process represents additional goals and tasks.

The starting event is again an incoming request analyzed initially by Support Level 1 (SL 1). Depending on the nature of the request, one to three issues resulting from the single request may have to be handled. Each separate topic is handled in the same manner by Support Level 2 (SL 2) using the same sub-process consisting of: "Problem Analysis", "Problems Resolution", "Documentation" and "Documentation Review". Once all required sub-processes are resolved the final feedback to the customer can be sent.

The DCR Graphs model features the same tasks, but differs in the structure of the sub-process. Not all of these tasks are relevant for the customer feedback, but for internal purposes, i.e. "Documentation" and "Documentation Review". The model allows to send customer feedback before all sub-processes are completely finished by requiring only the relevant tasks of the sub-process for the goal. In total the same amount of work has to be performed, but the order is less strict.

For this setup an additional indicator is used from the simulation results: The average time until the customer gets feedback from the moment that his request was placed. This is a key performance indicator for customer satisfaction.

The simulations are run with 8 cases per day, 2 workers for SL 1 and 6 for SL 2. The runs are done with one sub-process execution always being present a second with a 40% probability. The workload is increased over the runs by raising the probability for a third execution of the sub-process from 10% to 25%. The ACM runs are executed with Preference Agents that represent workers preferring to close cases or prefer handling customer feedback. Whenever such an agent gets to choose among different tasks the decision is based on the type of task. The Agent preferring the customer feedback will ignore documentation tasks as long as other tasks are required, the case closing agent behaves the opposite way.

The results in Fig. 2 on the left side show the same effect as seen previously that that the average case duration can be improved slightly by applying the declarative models. This is again attributed to the higher degree of parallelization of tasks. However, the knowledge worker behavior has an effect as it can be seen that preferring customer feedback leads to worse case closing times compared to the other behaviors as well as the BPM reference run. On the right side the BPM reference is identical to the average case duration as the case is closed with this task. All ACM variants improve on this base line because of task parallelization. In addition to this the effect that task selection has on the average time to customer feedback is significantly bigger than the difference in case closing times as the ability to reorder tasks according to the current needs has a bigger impact than parallelization effects.

For the runs with higher load the utilization level reaches $\sim$100% at 21% of 3$^{\text{rd}}$ execution probability. This is identical for all different agents as the total amount of work is independent of the agent behavior. The results show that the impact of the agent behavior increases as the workload increases. It can also be observed that customer feedback can be maintained at a reasonable level even beyond the 100% load for

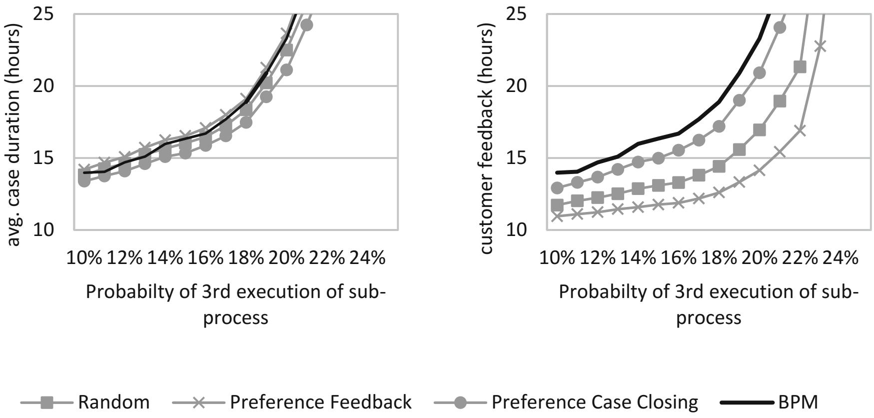


Fig. 2. Simulation results for sub-process scenario

the Agent preferring customer feedback. This comes at the cost of not closing cases completely anymore, because the worker time for customer feedback is shifted from the remaining tasks that would be required for case closing.

5 Conclusion and Future Research

5.1 Simulation Results

The first scenario shows a small improvement of the average processing time of the ACM process model. This means that, while the overall behavior of both models is similar in terms of resource utilization, the more flexible model leads to shorter average case durations. This holds true for the underlying assumption that the cost of information transfer within a case is negligible and thus complete parallelization is possible.

Another important aspect of ACM are ad-hoc decisions, which were studied in the second example. It was shown that their usage can have a significant positive impact on the average process durations if workers reach a sufficient level of competence. The process cannot consider all different circumstances that might apply in this specific situation. Imperative modeling is not flexible enough to model this in a maintainable way.

The third effect shown is that declarative process modeling can allow sub-processes to become more flexible, while still being modelled and maintained as sub-processes. When a small negative effect on the average process duration is accepted, additional freedom within the process allows preferred events (e.g.: customer feedback) to be reached significantly faster. Also for work-loads close to 100% of the capacity and even slightly above, the flexible process model can maintain reasonable times to achieve certain goals. These effects and their magnitude are highly dependent of the structure of the actual process.

The results of all the simulation runs have to be seen in perspective of the limitations of the simulation model. The focus of the models is the workflow, while the information and data aspects are ignored. Also human interactions that affect process execution are not modelled, which affects the ability to parallelize tasks.

It is possible to create functionally identical BPM models to the ACM models in some cases, but these have much higher complexity to create and are much harder to maintain. The reason for this is that imperative models are not designed for this degree of flexibility as the states rather than the circumstances are focused.

5.2 Answer of Research Question

It was found that there are scenarios where the ACM process can be more effective than the imperatively modelled processes. They can reduce the average case duration or even the total load on a worker pool. This potential was observed in all three scenarios. Under the assumption that information transfer between workers within a case is low, higher parallelization is reached. This was shown to lead to faster case execution times even if the overall work stays the same.

Faster process times can be reached due to the potential of higher parallelization in declarative models. The remaining observed effects are situational and depend highly on the worker behavior. The reduction of total load on a worker pool can occur if the knowledge worker reach a sufficient amount of competence. Reaching sub-goals faster also depends on the behavior of the knowledge worker.

5.3 Further Research

It is expected that ACM usage in the industry will increase in the near future. This will lead to the need of further understanding of the differences between ACM and BPM so that decision makers have the information available to base their decisions on. To obtain more detailed and precise results, the comparison of ACM and BPM models has to be investigated in further detail. Good metrics have to be found that represent the complexity and effort involved in creation and maintenance of the process models.

Additional aspects that set ACM apart have to be taken into consideration, which requires much further refined simulation tools as well as case studies. In particular, consideration of case content based simulation or more complex Agent behavior.

Also the impact on further result dimensions is relevant for evaluating the benefit of using ACM based processes. In this work, the process flows were evaluated mainly by efficiency and in one case customer satisfaction. Many other aspects are highly important as well e.g.: worker motivation, customer satisfaction, transparency and accountability.

References

1. Chinosi, M., Trombetta, A.: BPMN: an introduction to the standard. Comput. Stand. Interfaces **34**, 124–134 (2012)
2. Işik, Ö., Mertens, W., Van den Bergh, J.: Practices of knowledge intensive process management: quantitative insights. Bus. Process Manag. J. **19**, 515–534 (2013)

3. Pesic, M., van der Aalst, W.M.P.: A declarative approach for flexible business processes management. In: Eder, J., Dustdar, S. (eds.) BPM 2006. LNCS, vol. 4103, pp. 169–180. Springer, Heidelberg (2006). https://doi.org/10.1007/11837862_18
4. Slaats, T.: Flexible Process Notations for Cross-organizational Case Management Systems (2015)
5. Desel, J., Erwin, T.: Modeling, simulation and analysis of business processes. In: van der Aalst, W., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 129–141. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45594-9_9
6. Swenson, K.D.: Mastering the Unpredictable: how Adaptive Case Management Will Revolutionize the Way That Knowledge Workers get Things Done. Meghan-Kiffer Press, Tampa (2010)
7. Motahari-Nezhad, H.R., Swenson, K.D.: Adaptive case management: overview and research challenges. In: 15th Conference on Business Informatics (CBI) 2013, pp. 264–269. IEEE (2013)
8. Dorst, W.: Adaptive-Case-Management: Leitfaden und Nachschlagewerk (2013)
9. Hauder, M., Pigat, S., Matthes, F.: Research challenges in adaptive case management: a literature review. In: 18th International Enterprise Distributed Object Computing Conference Workshops and Demonstrations 2014. IEEE (2014)
10. Fischer, L. (ed.): How Knowledge Workers Get Things Done: Real-World Adaptive Case Management. Future Strategies, Lighthouse Point (2012)
11. Jansen-Vullers, M., Netjes, M.: Business process simulation–a tool survey. In: Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools, Aarhus, Denmark (2006)
12. Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus declarative process modeling languages: an empirical investigation. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) BPM 2011. LNBIP, vol. 99, pp. 383–394. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_37
13. Debois, S., Hildebrandt, T., Marquard, M., Slaats, T.: The DCR graphs process portal. In: Business Process Management Forum 2016 (2016)
14. Hildebrandt, T.T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. Electron. Proc. Theor. Comput. Sci. **69**, 59–73 (2011)
15. Hildebrandt, T., Marquard, M., Mukkamala, R.R., Slaats, T.: Dynamic condition response graphs for trustworthy adaptive case management. In: Demey, Y.T., Panetto, H. (eds.) On the Move to Meaningful Internet Systems: OTM 2013 Workshops, pp. 166–171. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41033-8_23

General Architectural Framework for Business Visual Analytics

Yavor Dankov(✉) and Dimitar Birov

Faculty of Mathematics and Informatics,
Sofia University "St. Kliment Ohridski", Sofia, Bulgaria
{yavor.dankov,birov}@fmi.uni-sofia.bg

Abstract. An incredible amount of data has constantly been generated. There is a gap between the abundance of data and the needs of users. At the same time a particular user needs not only a relevant part of the available data but (s)he needs higher-level (inferred) data in order to use this data in support of decision-making. Analytics comes to bridge that gap, by means of tools. Those tools **link** what the user needs to what is available as data; **generate higher-level (inferred) data** based on the raw data; **provide visualization** that is useful especially to users that have limited knowledge in mathematics and data science. The abundance of definitions that concern analytics as well as the numerous terms and concepts confuse users. Often different concepts are inconsistent with regard to each other. Inspired by this problem, the current paper: provides a systematic analysis featuring analytics in general and business analytics, in particular, developing this also from the perspective of visualization-related needs. We propose a layered business analytics architecture named **General Architectural Framework for Business Visual Analytics.** The framework presents the main interrelationships between the elements for designing and modeling **Business Visual Analytics.** We propose a conceptual definition that combines the semantics of the multiple recurring and overlapping definitions of analytics, named **Business Visual Analytics (BVA)**. This is a new conceptual viewpoint mainly focused on the innovative data visualization techniques, business analytics capabilities and achieving business goals and performance. The paper is reporting research in progress and for this reason, further architectural developments and related validations are planned for further research.

Keywords: Analytics · Business analytics · Visual analytics

1 Introduction

In the Digital age of emerging technologies, rapidly changing markets and consumer requirements, there are many business challenges for users and stakeholders that need to be resolved quickly, efficiently and in real time. Current business challenges relate to processing of data and extracting valuable information and knowledge from it. Every day, an incredible amount of data is constantly generating. The data can be from different data sources (internal, external, operational and others) or can be from any type of data (structured, semi-structured or unstructured data). There is a gap between the abundance of data and the needs of users (there is too much data about anything:

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 280–288, 2018.
https://doi.org/10.1007/978-3-319-94214-8_19

accounting/finance enterprise data, sensor data, surveillance data, and so on. At the same time a particular user needs not only a relevant part of the available data but (s)he needs higher-level (inferred) data in order to use this data in support of decision-making. The human brain would hardly cope with understanding the raw, unprocessed data and extracting knowledge from it.

Analytics comes to bridge that gap, by means of tools. Those tools:

- **Link** (similarly to brokerage systems) what the user needs to what is available as data;
- **Generate** higher-level (inferred) data based on the raw (sensor) data;
- **Provide** visualization that is useful especially to users that have limited knowledge in mathematics and data science.

Highly qualified teams of professionals create a variety of analytics. They develop, and support the application of analytics in different areas. According to their main role, the analytics are descriptive, predictive and prescriptive. Nevertheless, the abundance of definitions that concern analytics as well as the numerous terms and concepts confuse users. Often different concepts are inconsistent with regard to each other. Inspired by this problem, the current paper:

- Provides a systematic analysis featuring analytics in general and business analytics, in particular, developing this also from the perspective of visualization-related needs;
- Proposes a layered business analytics architecture.

The paper is reporting research in progress and for this reason, further architectural developments and related validations are planned for further research.

The paper is organized as follows: Sect. 2 considers the analytics. Section 3 presents business analytics. Section 4 analyses the visual analytics and visualization aspects. Section 5 proposes a General Architectural Framework for Business Visual Analytics. The section provides briefly analysis of the framework. The paper finishes with related works and a conclusion.

2 Analytics

The analytics represents tools with a defined purpose. Multiple definitions of Analytics are developing and extending thanks to the scientific researchers and practitioners. The definition by Davenport and Haris [1] represents analytics as “the extensive use of data, statistical and quantitative analysis, explanatory and predictive models, and fact-based management to drive decisions and actions”. Another appropriate definition of analytics is the following: “Analytics is the process of developing actionable insights through problem definition and the application of statistical models and analysis against existing and/or simulated future data” [2]. The analytics may involve human decisions or can operate automated decision processes [3]. Shortly, the analytics “refers to a set of techniques and applications to explore, analyze, and visualize data from both internal and external sources” [4]. The analytics uses the unprocessed raw data from different data sources. Then the analytics processes the data to derive valuable insights from that.

After that, the analytics visualizes the information to the users via some kind of data visualization technique. The users use the new insights to help the decision-making processes [5].

The researchers and scientific community are developing new and innovative types of analytics regarding their role to help the business and facilitate the work of the users. The most common classification of analytics divides it into three main groups [4–6]: descriptive, predictive and prescriptive. There are existing variations of taxonomies for analytics adding two or more categories such as Diagnostic Analytics [7] and Cognitive Analytics [8]. The names of these three main groups show their main role and purpose. Briefly [9]:

- Descriptive analytics analyses the available data to provide insights for the past and tries to answer the question - What has happened?
- Predictive analytics presents models of the analyzed data of the past in order to predict the future and provide some explanation of an occurrence of an event. The goal is to answer the question - What could happen?
- Prescriptive analytics are useful tools in defining what is the best solution or decision. It uses optimization and simulation algorithms to advice on possible outcomes - What should happen?

The analytics tools are applied in different areas such as – enterprise ecosystems, business, engineering, education [10], healthcare [11], e-commerce, governments [12], telecommunication, transportation [7], surveillance [13] and cybersecurity [14, 15] and so on.

Many analytic tools combine different types of analytics in their structure. One tool of analytics can consist several types of analytics with the task for describing, analyzing, processing data for solving a specific problem. The analytics can represent a real-time monitoring tool that operates over a desired database or big data and monitors for real-time occurring issues. Everything operates under constraints such as metrics, measurements, and business strategies and so on. The users can monitor and regulate these constraints, add some new ones or delete some of them. This allows the users to adjust the tool with endless possibilities in terms of strategic needs and the desired tasks. The combination of different types of analytics in one fully customizable and working tool is very convenient and powerful for the users and stakeholders. The integration of such tools in the daily enterprise work could increase the overall performance of users and enterprises. Analytics provides innovative analytics capabilities that contribute to the decision-making processes and taking actions based on knowledge gained from processed data.

3 Business Analytics

After having introduced the "analytics" notion, we are going to focus particularly on the enterprises-relevant analytics that is called "business analytics". Unlike healthcare analytics (that concerns capturing vital signs, such as blood pressure, pulse, and so on), traffic analytics (that concerns sensor data featuring vehicles and/or traveling people), and so on, BUSINESS ANALYTICS (BA) is about enterprise data that is often generated by the accounting department of an enterprise, and is about profits, turnovers, money flows, and so on. This data has particular analytical day-to-day importance when reflected in the accounting system of an enterprise, where each transaction is accounted based on a primary accounting document, such as receipt, bank statement, and so on. Nevertheless, at this level, accounting data is of limited use to managers who need inferred higher-level business data, such as “monthly turnover”, for example, to help them in decision-making. Therefore, business analytics would often be about inferring higher-level accounting data based on the low-level (raw) data that corresponds to the accounted transactions.

The main difference between business analytics and analytics is that business analytics requires [16, 17]:

- **A direct relation to business**;
- A **resulting insight** implemented into the **business in the future**;
- **Performance** and **measurement values** based on metrics that will lead to a successful business result.

Nevertheless, based on our scientific research, we have perceived that there is not an established academic definition of Business Analytics [18]. Many enterprises integrate the tools of BA into their business structures and industrial ecosystems, mainly because BA offers many capabilities contributing to improve overall business performance and reactivity in real time. The application of multiple data analysis methods [19], methodologies and technologies [20] for gaining important insights over data, helps the management to “integrate disparate data sources, predict trends, improve performance, see key performance indicators, identify business opportunities, and make better and informed decisions” [21].

The enterprises are striving to achieve certain business goals with the help of BA. For that reason the (business) analytics are constantly developing. Usually, analytics processes data from database or data warehouse [22] and possesses some kind of visual communication between the machine and the user – this is the interface. The interface commonly refers to some kind of communication that enables the user to connect, interact with a computer, machine, tool and so on, and facilitates the work of the user. For that reason, a successful tool of analytics possesses an easy to understand interface - graphical or other type that could increase the effectiveness, performance, and reuse of that tool.

4 Visualization Aspects

Even though an intelligent and well educated manager would be expected to understand data that is presented in reports and/or sheets, it is often even more helpful (especially to managers of no special attitude towards mathematics, data science, and so on) to provide nice visualizations that help the manager understand what inferred data is to show. For this reason, it is not surprising that visualization-related issues concerning (business) analytics are currently receiving increasing attention. Even (discrete-event) simulation is currently also focusing also on visualizations (next to other things). Hence, we argue and the business analytics of the future is to definitely include adequate visualization features.

Establishing a permanent definition regarding analytics and Visualizations is one of the challenges to researchers and practitioners. The problem is complex mainly because users and enterprises use different understandings of analytics, types of visualizations, visualization techniques and metaphors in their business. The visualization of data can be presented as a communication between the user and the analytics through the use of interactive interfaces with three major goals including Presentation, Confirmatory analysis and Exploratory data analysis [23].

The Visual Analytics (VA) is a rapidly growing field of scientific research and development. It is not very easy to perceive a permanent definition of VA, because it comprises several sciences and have a wide multidisciplinary area of application [24]. We perceive the definition of VA proposed by Thomas and Cook [25]. They define that the VA represents "the science of analytical reasoning facilitated by interactive visual interfaces" [25, 26]. VA represents "an integral approach to decision-making processes combining visualization, human factors, and data analysis." [23]. The VA uses the scientific contributions from studies in sciences as Human-Computer Interaction, Cognitive Psychology, and Perception.

The "Visual analytics combines automated analysis techniques with interactive visualizations for an effective understanding, reasoning and decision making on the basis of very large and complex datasets" [24]. These techniques can be "graphics, visualization, interaction, data analysis, and data mining to support reasoning and sense making for complex problem solving" [27]. Combining the analytical and interactive visual techniques contributes to the possibility for knowledge and pattern discoveries, gaining more insights and deeply understanding the processed data and mechanisms. [27, 28].

VA enables people to "synthesize information and derive insight from massive, dynamic, ambiguous, and often conflicting data" [24]. Collaboration may occur at the level of data management, visualization, or analysis" [29]. VA utilizes new and innovative visualization techniques for interactive relationship between VA and user. They facilitate the visual processing of data and encourage decision-making processes in business [30]. VA is based on the interrelationship between (data-information-knowledge) and contributes to a management that will undertake timely strategic actions, and minimize business risk and failure. VA is mainly focused on the utilization of innovative data visualizations that helps the user to see, understand and analyze data visually and take informed decisions.

5 Proposed Architecture

Inspired by the analyses carried out in Sects. 2, 3 and 4, we propose in the current section holistic architectural views (and a corresponding definition) that are essentially focused on business analytics.

We propose a layered business analytics architecture named **General Architectural Framework for Business Visual Analytics** as shown in Fig. 1. The framework presents the main interrelationships between the elements for designing and modeling **Business Visual Analytics** (BVA).

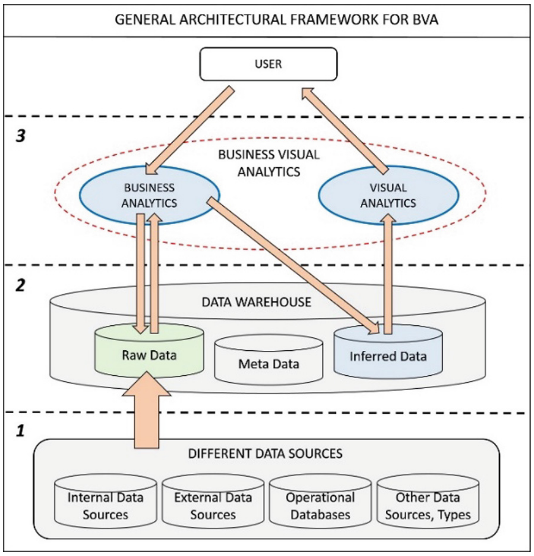


Fig. 1. General Architectural Framework for Business Visual Analytics

As we have presented in previous sections, the multiple recurring and overlapping definitions of analytics, BA and VA often confuse the users. For that reason, we propose a conceptual definition named **Business Visual Analytics** that combines the semantics of those definitions. The BVA is a new conceptual viewpoint mainly focused on the innovative data visualization techniques, BA capabilities and achieving business goals and performance. This paper proposes the definition of BVA as follows:

BVA represents a tool that combines different types of analytics in business, taking advantage of the business analytics capabilities and innovative data visualizations techniques, to gain higher-level inferred data to make timely informed, strategic, business decisions and take actions on solving complex business problems.

The first layer of the framework represents the different sources of data. The sources of data could be internal data, external data, operational databases or other types of data sources and data generated from sensors, machines and so on. All data enters the second layer of the framework as **Raw Data** and it is stored in data warehouse. The Data Warehouse (layer two) comprises all the data for processing and analyzing. The layer includes Raw Data, Meta Data and Inferred Data. The third layer of the framework represents different types of analytics – Business Analytics and Visual Analytics (or **BVA**).

A user has to solve a concrete business problem and needs a higher-level (inferred) data for better understanding the issues and make decisions. For that reason, (s)he uses tools of BA. Firstly, BA use the **raw data** from the Data Warehouse. BA processes the raw data and applies the defined business logic and metrics on data for achieving a concrete business objective. Then, BA generates an output/result as inferred data that is stored again in Data Warehouse. VA use the inferred data to visualize it to the user. VA applies innovative data visualization techniques and allows the user to see, understand and analyze visually the inferred data. This enables the user to make informed, strategic and timely decisions. Based on concrete business problems BA and VA creates an interactive connection between the business logic and the best appropriate visualizations. The BVA could encapsulate the recurring and overlapping definitions of analytics and provide an easy to understand tool that is useful for all users, especially to users that have limited knowledge in mathematics and data science.

6 Related Work

The scientific community does research and develops the field of analytics. In this paper, we have just mentioned some of the publications that we consider as a foundation for our work. We analyse the definitions of analytics and different types of analytics from [1–9]. As a foundation definition of VA we perceive the one from [25]. As addition to the definitions for VA from [23, 24, 27–29]. We adopt the meaning for BA from [16–22]. The analytics tools are applied in different areas such as – enterprise ecosystems, business, engineering, education [10], healthcare [11], e-commerce, governments [12], telecommunication, transportation [7], surveillance [13] and cybersecurity [14, 15] and so on.

The area of application of analytics is constantly growing. In previous paper, we have proposed a classification framework of integration of analytics in business named CSPD Integration of Analytics (Conceptual, Specialized, Platform, and Data Integration of Analytics) [9]. Both papers are related to each other and are planned for further research. The current paper is reporting research in progress and for this reason, further architectural developments and related validations are planned for further research.

7 Conclusion

Acknowledging the increasing importance of (business) analytics for bridging the gap between the abundance of data and the user's needs of data, we have identified many (often mutually contradicting) relevant definitions/concepts. We argue that this is confusing for users and hence, we have carried out a systematic analysis featuring analytics in general and business analytics, in particular (also touching upon visualization issues). Inspired by those analyses, we have proposed holistic general architectural guidelines that concern business analytics. We plan for further research to develop the proposed architectural framework and to validate it by means of case studies.

References

1. Davenport, T.H., Harris, J.G.: Competing on Analytics: the New Science of Winning. Harvard Business School Press Boston, Boston (2007)
2. Cooper, A.: What is analytics? definition and essential characteristics. CETIS Anal. Ser. **1** (5), 1–10 (2012). ISSN 2059214
3. Laursen, G., Thorlund, J.: Business Analytics for Managers: Taking Business Intelligence Beyond Reporting, 2nd edn. Wiley, Hoboken (2017). ISBN 9781119298588
4. Vanthienen, J., Witte, K.: Data Analytics Applications in Education, 1st ed. CRC Press/ Taylor Francis Group/Auerbach Publications, New York (2017). ISBN 9781498769273
5. Somani, A., Deka, G.: Big Data Analytics: Tools and Technology for Effective Planning. CRC Press/Taylor & Francis Group, New York (2018). ISBN 9781138032392
6. IBM Software Group: Descriptive, Predictive, Prescriptive: Transforming Asset and Facilities Management with Analytics. IBM Corporation Software Group, New York (2013)
7. Chowdhury, M., Apon, A., Dey, K.: Data Analytics for Intelligent Transportation Systems, 1st edn. Elsevier Science Publishers, Amsterdam (2017). ISBN 9780128097151
8. Sami, M.: The Evolution of Analytics. https://melsatar.blog/2017/07/30/the-evolution-of-analytics/. Accessed 3 May 2018
9. Dankov, Y., Birov, D.: Extended conceptual framework for business analytics supporting innovations. In: Ketikidis, P., Solomon, A., Thessaloniki, G. (eds.) Proceedings of International Conference for Entrepreneurship, Innovation and Regional Development ICEIRD10, 31 August–1 September 2017 (2017). ISBN 9789609416115, ISSN 24115320
10. Peña-Ayala, A. (ed.): Learning Analytics: Fundaments, Applications, and Trends. SSDC, vol. 94. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52977-6
11. Dodd, J.: Healthcare It Transformation: Bridging Innovation, Integration, Interoperability, and Analytics, 1st ed. CRC Press/Taylor & Francis Group/Productivity Press, New York (2017). ISBN 9781498778442
12. Lowman, M.: A Practical Guide to Analytics for Governments: Using Big Data for Wiley and SAS Business Series. Wiley, Hoboken (2017). ISBN 9781119362821
13. Yan, W.Q.: Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics, 2nd edn. Springer International Publishing AG, Switzerland (2017). https://doi.org/10.1007/978-3-319-60228-8. ISBN 9783319602271
14. Thuraisingham, B., Parveen, P., Masud, M., Khan, L.: Big Data Analytics with Applications in Insider Threat Detection, 1st ed. CRC Press/Taylor & Francis Group/Auerbach Publications, Boston (2017). ISBN 9781498705479
15. Alsmadi, I., Karabatis, G., Aleroud, A.: Information Fusion for Cyber-Security Analytics. Studies in Computational Intelligence, vol. 691, 1st edn. Springer International Publishing, Switzerland (2017). https://doi.org/10.1007/978-3-319-44257-0. ISBN 9783319442563, 9783319442570
16. Schniederjans, M., Schniederjans, D., Christopher, J.: Business Analytics Principles, Concepts and Applications. Pearson Education Inc, USA (2014). ISBN 9780133552188
17. Stubbs, E.: The Value of Business Analytics. Wiley, Hoboken (2011)
18. Bichler, M., Heinzl, A., van der Aalst, W.M.P.: Business analytics and data science: once again? Bus. Inf. Syst. Eng. **59**(2), 77–79 (2017). https://doi.org/10.1007/s12599-016-0461-1
19. Shmueli, G., Bruce, P.: Data Mining for Business Analytics: Concepts, Techniques, and Applications with XLMiner, 3rd edn. Wiley, Hoboken (2016). ISBN 9781118729274
20. Evans, J.: Business Analytics: Methods, Models, and Decisions, 2nd edn. Pearson Education, Boston (2017). ISBN 9781292095448

21. Bayrak, T., A review of business analytics: a business enabler or another passing fad. In: World Conference on Technology, Innovation and Entrepreneurship, Procedia - Social and Behavioral Sciences vol. 195, pp 230 – 239, Elsevier Ltd (2015). ISSN: 18770428
22. Sugumaran, V., Sangaiah, A., Thangavelu, A.: Computational Intelligence Applications in Business and Big Data Analytics, 1st edn. CRC Press/Taylor & Francis Group, FL (2017). ISBN 9781498761017
23. Keim, D.; Mansmann, F.; Stoffel, A., Ziegler, H.: Visual Analytics., Encyclopedia of Database Systems/M. Tamer Özsu et al. (ed.) - Heidelberg: Springer, (2008). http://nbn-resolving.de/urn:nbn:de:bsz:352-opus-68335. Accessed 2 Nov 2018
24. Keim, D., Kohlhammer, J., Ellis, G., Mansmann, F.: Mastering the Information Age – Solving Problems with Visual Analytics. Eurographics Association, Goslar (2010). ISBN 9783905673777
25. Thomas, J., Cook, K.: Illuminating the path: the research and development agenda for visual analytics. In: National Visualization and Analytics Center (2005). ISBN:0769523234
26. Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., Melançon, G.: Visual Analytics: Definition, Process, and Challenges. In: Kerren, A., Stasko, John T., Fekete, J.-D., North, C. (eds.) Information Visualization. LNCS, vol. 4950, pp. 154–175. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70956-5_7
27. Telea, A., Ersoy, O., Voinea, L.: Visual analytics in software maintenance: challenges and opportunities. In: International Symposium on Visual Analytics Science and Technology, Kohlhammer, J., Keim, D. (eds.), the Eurographics Association (2010)
28. Dill, J., Earnshaw, R., Kasik, D., Vince, J., Chung Wong, P.: Expanding the Frontiers of Visual Analytics and Visualization. Springer, London Dordrecht Heidelberg New York (2012). https://doi.org/10.1007/978-1-4471-2804-5. ISBN 9781447128038
29. Heer, J., Agrawala, M.: Design considerations for collaborative visual analytics, information visualization. In: Proceedings of IEEE Symposium on Visual Analytics Science and Technology 2007, Sacramento, CA, USA, vol. 7 (2008). https://doi.org/10.1057/palgrave.ivs.9500167
30. Suh, S., Anthony, T.: Big Data and Visual Analytics, 1st edn. Springer International Publishing AG, Switzerland (2017). ISBN 9783319639154

A Causal Explanatory Model of Bayesian-belief Networks for Analysing the Risks of Opening Data

Ahmad Luthfi[1,2](✉), Marijn Janssen[1], and Joep Crompvoets[3]

[1] Faculty of Technology, Policy and Management, Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands
{a.luthfi,m.f.w.h.a.janssen}@tudelft.nl
[2] Universitas Islam Indonesia, Yogyakarta, Indonesia
ahmad.luthfi@uii.ac.id
[3] Katholieke Universiteit Leuven, Leuven, Belgium
joep.crompvoets@kuleuven.be

Abstract. Open government data initiatives result in the expectation of having open data available. Nevertheless, some potential risks like sensitivity, privacy, ownership, misinterpretation, and misuse of the data result in the reluctance of governments to open their data. At this moment, there is no comprehensive overview nor a model to understand the mechanisms resulting in risk when opening data. This study is aimed at developing a Bayesian-belief Networks (BbN) model to analyse the causal mechanism resulting in risks when opening data. An explanatory approach based on the four main steps is followed to develop a BbN. The model presents a better understanding of the causal relationship between data and risks and can help governments and other stakeholders in their decision to open data. We use the literature review base to quantify the probability of risk variables to give an illustration in the interrogating process. For the further study, we recommend using expert's judgment for quantifying the probability of the risk variables in opening data.

Keywords: Bayesian-belief Networks · Causality · Open data Relationship · Explanatory model · Risks · Sensitivity · Privacy Ownership · Misinterpretation · Misuse

1 Introduction

The expectations to gain access to public data have increased extensively over the last few years [1]. The creation of transparency, accountability, citizen engagement, and to enable business innovation are the main drivers for governments to open more their data [2–4]. The public expects governments to release their dataset for reaping the many benefits of opening data. For example, parents can explore open datasets from the government's portal to find the quality of educational institutes for their children. Foreigners can access open tourism datasets to decide some alternative destinations for their vacations.

Although initiatives to open data can create many benefits, they might also create disadvantages [1, 5]. Risks refer to the chance that these disadvantages come true and

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 289–297, 2018.
https://doi.org/10.1007/978-3-319-94214-8_20

its impact. Potential risks include inaccuracy, sensitivity, privacy, inconsistency, and misuse of data. These risks result in the reluctance of governments to open their data [1]. In addition, two other reasons why governments and data providers are tend not to open their data: (1) opening public and/or private data are a comprehensive insight that may also able to meet risks like inappropriate interpretation of the data [4], and (2) a mistake in translating data or misuse of the data can endanger the reputation of data providers [6]. Governments sometimes also need a huge effort to investigate and analyze the cause and effect in terms of the opening data. Cause refers to an event or action of the risk in opening data that induces something else to occur. Effect means an event or action in releasing a dataset that occurred as a consequence of another event or action. For example, because of the inappropriate visualization of a dataset in government's portal information (as a cause), then the public will tend to misinterpretation of the dataset released (as an effect). Unfortunately, at this moment, the investigation of the risk in opening data and to what disadvantages the opening of data might result is not well-understood yet.

Bayesian-belief Network (BbN) is a method presented in this study because it allows constructing a causality relationship model between risk and effect variables [1, 7]. The use of a BbN in the risks analysis can provide explanations how the disadvantages of the opening of data are created. The BbN is able to capture the strength of causal links to define the cause under uncertainty variables [8, 9], and to visualize the possible consequences of the risks [10]. The consequences of the risks are the disadvantages and the impact of our situation. In this study, we develop an explanatory model as an approach to understand the relationship and influence factors of the risks in opening data. The explanatory processes follow four main steps [11]. To begin with, we require defining the risk variables and its relationships. Next, a network structure of the risks is developed showing how the variables interrelated. In the third step, we interrogate the model to obtain the understand the sensitivity of variables, and finally, the relationship diagrams are developed to enable to communicate the outcomes to stakeholders. The model is illustrated using health patient records dataset to analyze and to understand the relationship of causality factors of the risks in opening data.

The objective of the research presented in this paper is to develop a Bayesian-belief Network model for understanding the causal mechanisms of the risks in opening data. The purpose of the explanatory model is to enhance the understanding of governments and data providers of the potential risks in opening data and why these risks occur. This paper is systematically composed of five sections. In Sect. 1, the research drivers were described, Sect. 2 contains related works for BbN in risk assessment. In Sect. 3 the research approach taken in this study is explained. In Sect. 4, the development of a BbN is step-by-step presented, and finally, conclusions are drawn in Sect. 5.

2 Related Work

The BbN model was introduced for the first time in the field of risk management for terrorism threats [12]. In the second area, a BbN theory was implemented in a research related to the determination and fire mitigating actions [13]. Literature was reviewed to investigate the use of BbN for risk assessment. Table 1 shows seven related domains in which BbN was used to analyze the potential risks. Only one paper is related to the opening data (see paper number 4). They proposed a model to weigh the potential risks and benefits of opening data. However, this paper did not focus on how to construct a BbN causality model to analyze the risks of opening data. The scientific contribution of this paper originates from developing a conceptual model of causality between risk variables in opening data.

Table 1. Related works in risks analysis using Bayesian-belief Networks

	Domain of the study	Description	Source
1	Bank	A case study to develop valid statistical models to measure, and predict such operational risks using Bayesian networks in the basel banking	[14]
2	Health	A presentation of health risks in the area of non-carcinogenic substances for non-critical organs and the additive assumption for multiple hazardous substances affecting the same organ	[15]
3	Smartphones	A discussed related to the information security and risk assessment of smartphone use in Finland	[10]
4	Open data	An iterative process of decision support model to weigh the potential risks and benefits of opening data	[1]

3 Research Approach

In this study, we will develop an explanatory BbN for the opening data. We adopted four main steps to develop a causality and relationship of the risk variables [11], as visualized in Fig. 1.

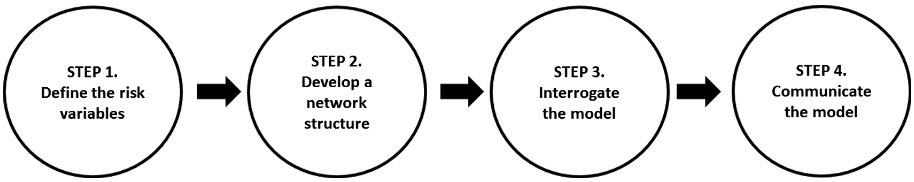


Fig. 1. Step-by-step of the Bayesian-belief Network development model

- Step 1. **Define the risk variables**. First of all, we conducted the literature review to identify the possible types of risk related to the opening data. Then, we classified the risk into several categories to make a singular variable of the risk for classifying the cause and effect elements. To make clear understanding and to avoid misinterpretation of the risk definition, we described the risks elements.

- Step 2. **Develop a network structure**. In this stage, a BbN structure is developed to present a set of risks elements that influence the potential disadvantages of releasing a dataset and provide its relationships. The parent node is a single element of the risk variable that influenced by a single or some causes variable. There are three sub-steps were followed to identify sub-nodes and their relationship. Sub-nodes and its relationships are possible to be repeated several times during the constructing a BbN.
  - Sub-step 2.a: identify the key elements. The key elements will become parent nodes of the top level node. The objective of this sub-step is to construct further sub-elements of the risk variables influencing them.
  - Sub-step 2.b: identify the remaining elements. This sub-step is to describe the causality of the various risk elements until the lowest of levels is generated. For example, Data Misuse (Z) is a parent node that influenced by two sub-nodes namely Data Sensitivity (X) and Data Ownership (Y).
  - Sub-step 2.c: identify the relationship. The objective of this sub-step is to identify the various nodes includes key elements, remaining elements based on their influences on each other. The relationship knowledge, in this work, will be identified from the literature based.
- Step 3. **Interrogate the model**. This step is aimed to interrogate the sensitivity and influence of variables on the risks. Sensitivity means that the responsiveness of each node or variable in the network structure is analyzed using a systematic approach to express the trigger variables [10, 11]. While the influence tends to analyze the frequency of impacts of the parent nodes on their respective child nodes by identifying their influential elements. The expected result of this step is to provide a better understanding of the most significant elements of the risks in opening data.
- Step 4. **Communicate the model**. The final process of the BbN development in this study is how to communicate the resulting network model. In this paper, we utilize the relationships diagram to describe the results visually.

4 An Illustration: Development of a Bayesian-belief Networks

In this section, we give an illustration to show the relationships of influence risks variable in opening data. The four steps explained in the research approach are followed. We illustrated using a Health Patient Record dataset to analyze the possibility of risks. The dataset, for instance, consists of some attributes like name_of_patient, date_of_birth, address, and phone_number. The scenario is government want to analyze the potential risk of the attributes before it released. By constructing a BbN, the government is able to understand the causality and its relationships of the risk element.

4.1 Define the Risk Variables

Open data have been shown to contribute to society through several programs of the governments of many countries in recent years [16]. Nevertheless, along with the benefits

Table 2. Type of risks in opening data

	Category	Description	Source
1	Data inaccuracy	Data inaccuracy can occur when the data providers release their data. Some of the causes of data inaccuracy include: (a) data entry mistake by the users or data operators, (b) flawed data entry process, (c) the null problem with the value of the data, and (d) deliberate error when the users enter a wrong value of the data. This category can affect the quality of the data	[18, 20, 21]
2	Data misuse	Data disclosure can make personal or individual data identifiable by combining several datasets. Some cause misuses of the data are: (a) discredit personal profile, (b) access as unauthorized users, and (c) diminish the government's or company's reputation. This situation influencing the data privacy	[18, 22–24]
3	Data sensitivity	Releasing data can include sensitive attributes. Personal identity elements like full name, date of birth, address, and phone number are possible to be analyzed by the users. This category, influencing the data privacy and data violation	[18, 25]
4	Data incompleteness	Opening incomplete data, create misunderstanding about the meaning of the data. The caused elements of this category are (a) the anonymity of the data source, (b) inappropriate aliases formula, and (c) mismatch of the attribute relationships. This situation is also possible influencing the data quality and data misinterpretation	[18, 22, 23, 25, 26]
5	Data misinterpretation	Publishing data by governments or companies are possible to drive a misinterpretation of the data. The causes factors of this category are: (a) insufficient domain expertise, (b) important variables are omitted, (c) inappropriate data visualization, and (d) error of attribute correlation. The effect of this risk category is influencing the data quality and data incompleteness	[17–19, 23, 27]

of implementing the disclosure of data, potential risks of opening data are emphasized [17–19]. The risks are classified into five categories, as presented in Table 2.

4.2 Develop a Network Structure

In this step, we begin to develop a BbN structure to identify the causes and relationships between risk elements. In step 1, define the risk variables, we have identified five main categories of the risk in opening data. Based on the cause-and-effect for each category, a BbN structure is created. To do so, there are three sub-steps to make a detail action of this process, as follows: (1) Identify the key elements. From the identified of the risk elements, we need to classify which are the parent nodes in particular,

(2) Identify the remaining elements. After the parent and the child nodes have identified, we then make a connection between the parent and child as the one to one or one to many connectivity, and (3) Identify the relationships. The main objective of this process is to make a relationship between parent nodes to other parent nodes become a network a relationship. This step can be processed in several iterations until the lowest sub-node is identified and correlated.

- Sub-step 2.a. **Identify the key elements**. We identified parent nodes of the top level nodes for each of the five categories.
  - Data inaccuracy parent node. Factors like a data entry mistake, flawed data entry process, the null problem, and deliberate error as the influencing factors are sub-elements.
  - Data misuse parent node. The cause elements of this category are discredited personal profile, unauthorized user, and diminish reputation.
  - Data sensitivity parent node having personally identifiable, as a single sub-element.
  - Data incompleteness parent node. The anonymity of data source, inappropriate aliases formula, and the mismatch of attribute relationship are the influencing factors.
  - Data misinterpretation parent node. The insufficient of domain expertise, omitted important variables, inappropriate data visualization, and error of attribute correlation are all sub-elements of data misinterpretation.
- Sub-step 2.b. **Identify the remaining elements**. From the parent node and sub-elements constructed in Sub-step 2a, we made the connection between the variables. The correlation of each node and sub-elements show the relationship of the risks elements until the lowest of levels.

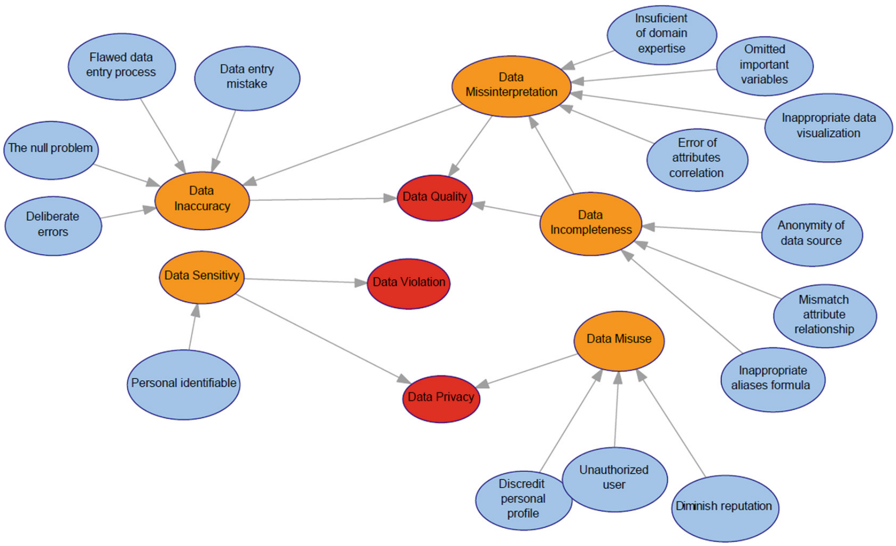


Fig. 2. Causality model of the risks in opening data

- Sub-step 2.c. **Identify the relationships**. In this step, the effect elements of the parent nodes are constructed. There are three attributes influenced by the parent nodes. First, data quality is affected by the data inaccuracy, data misinterpretation, and data incompleteness. Second, data privacy is influenced by the data misuse, and data sensitivity. Finally, single data violation is affected by the data sensitivity. The causal network structure and its relationship is visualized in Fig. 2.

4.3 Interrogate the Model

After the BbN structure is developed, we interrogated the resulting model by distributing the probabilities for each node and sub-elements. The objective of this step is to interrogate the sensitivity level the risk elements and to present the highest and the lowest probabilities of the risks. We designed using the literature review base to quantify the value of each risk element. We divided the value of the probabilities into three options: Yes, Neutral, and No.

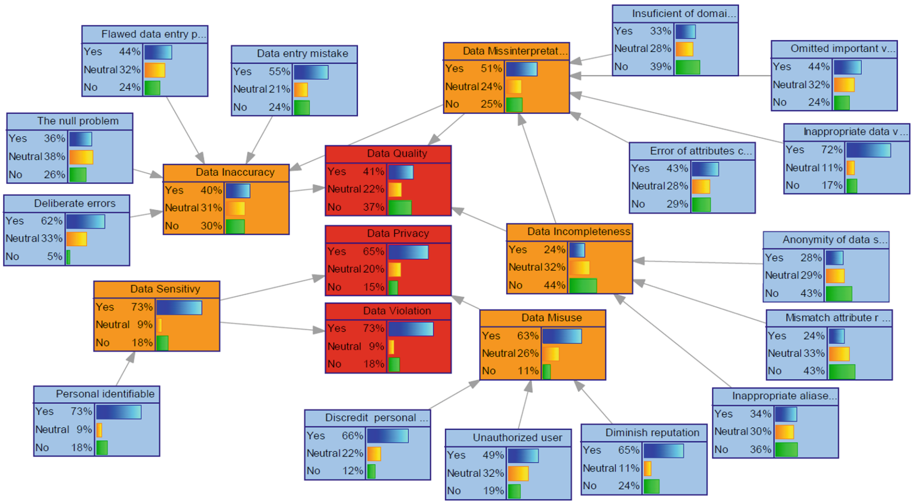


Fig. 3. Causality of the risk elements relationships in opening data

Figure 3 presents the causality and relationships between risk variables in opening data. Based on the literature review, we quantified both cause and effect nodes. To illustrate, we gave values for Data Misuse parent node (Yes = 0.63; Neutral = 0.26, No = 0.11). For cause variables, we identified that the value of sub-nodes as follows: (1) Discredited personal profile (Yes = 0.66; Neutral = 0.22, No = 0.12), (2) Unauthorized user (Yes = 0.49; Neutral = 0.32, No = 0.19), and (3) Diminish reputation (Yes = 0.65; Neutral = 0.11; No = 0.24). Next, we require quantifying the Data Privacy variable as the parent node of Data Misuse with (Yes = 0.65; Neutral = 0.20; No = 0.15). In this case, we can assume that some attributes of health patient records as a given dataset in this illustration has 0.65 or 65% the potential risks when it decided to publish.

4.4 Communicate the Model

The final step of the BbN model is to create a model that can be communicated to stakeholders like decision-makers, policy-makers, and data providers. In further research, we will develop a comprehensive explanation of the model's development in line with the various types of analysis formats [10, 11]. There are some approaches to disseminate the data to the public. For example, data visualization platforms like graphs, charts, histogram, or scatter plot are able to help the stakeholders to use the models in practice [11].

5 Conclusion

The public expects governments and data providers to open their data for reaping the benefits. However, behind the merits of releasing the data, the governments are often reluctant to open their data due to possible disadvantages. Disadvantages like low data quality, individual identification, and opening inaccurate data are the main drivers of government for not opening their data. The risk elements are used for determining the exposure to these disadvantages. At the same time, the causality and relationships between the potential of the risk elements are not investigated yet. In this paper, we derive a Bayesian-belief Network (BbN) to construct a causal model of the risk in opening data. The explanatory model consists of four main steps. First, define the risk variables and its relationship. Second, a network structure of the risk is constructed. In the third step, we interrogate the model to obtain the current sensitive analysis, and finally, we require to communicate the model using relationship diagram to provide the new knowledge to the stakeholders in terms of the risks analysis in opening data. The main purpose of the model is to present the better understanding of the governments and other stakeholders in decision analysis. For example, from the illustration presented in this paper, we noticed that governments need to consider the data privacy issues including its causes and relationships. The limitation of this study is using the literature review base to quantify the risk variables to give an illustration in the interrogating step. For the further research, we recommend using expert's judgment for quantifying the probability of the risks variables in opening data.

References

1. Luthfi, A., Janssen, M.: A conceptual model of decision-making support for opening data. In: Katsikas, Sokratis K., Zorkadis, V. (eds.) e-Democracy 2017. CCIS, vol. 792, pp. 95–105. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71117-1_7
2. Ali-Eldin, A.M.T., Zuiderwijk, A., Janssen, M.: Opening more data: a new privacy scoring model of open data. In: Seventh International Symposium on Business Modelling and Software Design (BMSD 2017), Barcelona, Spain. SCITEPRESS - Science and Technology Publication, Lda (2017)
3. Lourenço, R.P.: An analysis of open government portals: a perspective of transparency for accountability. Gov. Inf. Q. **32**(3), 323–332 (2015)
4. Zuiderwijk, A., Janssen, M.: Open data policies, their implementation and impact: a framework for comparison. Gov. Inf. Q. **31**(1), 17–29 (2013)
5. Zuiderwijk, A., Janssen, M.: Towards decision support for disclosing data: closed or open data? Inf. Polity **20**(2–3), 103–107 (2015)

6. Barry, E., Bannister, F.: Barriers to open data release: a view from the top. Inf. Polity **19**(1–2), 129–152 (2014)
7. Fenton, N., Neil, M.: Risks Assessment and Decision Analysis with Bayesian Networks. CRC Press, Noca Raton (2012)
8. Nadkarni, S., Shenoy, P.P.: A causal mapping approach to constructing Bayesian networks. Decis. Support Syst. **38**(2), 259–281 (2004)
9. Hu, H., Elrafey, A.H., Kerschberg, L.: Using modular ontologies to capture causal knowledge contained in Bayesian networks. In: 2017 IEEE/ACM International Conference. ACM, Sydney (2017)
10. Herland, K., Hämmäinen, H., Kekolahti, P.: Information security risks assessment of smartphones using Bayesian networks. J. Cyber Secur. **4**, 65–85 (2016)
11. Chakraborty, S., et al.: A Bayesian network-based customer satisfaction model: a tool for management decisions in railway transport. J. Decis. Anal. **3**(4), 2–24 (2016)
12. Hudson, L.D., et al.: An Application of Bayesian Networks to Antiterrorism Risk Management for Military Planners (2002)
13. Gulvanessian, H., Holický, M.: Determination of actions due to fire: recent developments in Bayesian risk assessment of structures under fire. Prog. Struct. Eng. Mater. **3**(4), 346–352 (2001)
14. Cornalba, C., Giudici, P.: Statistical models for operational risk management. Phys. A Stat. Mech. Appl. **388**(1–2), 166–172 (2004)
15. Liu, K., et al.: Applying Bayesian belief networks to health risks assessment. Stochast. Environ. Res. Risk Assess. **26**(3), 451–465 (2012)
16. Zuiderwijk, A., Janssen, M., David, C.: Innovation with open data: essential elements of open data ecosystems. Inf. Polity **19**(2–3), 17–33 (2014)
17. Barnickel, N., et al.: Berlin open data strategy: organisational, legal and technical aspects of open data in Berlin. In: Concept, Pilot System and Recommendations for Action (2012)
18. Martin, S., et al.: Risk analysis to overcome barriers to open data. Electron. J. e-Gov. **11**(1), 348–359 (2013)
19. Janssen, M., Charalabidis, Y., Zuiderwijk, A.: Benefits, adoption barriers and myths of open data and open government. Inf. Syst. Manag. **29**(4), 258–268 (2012)
20. Chen, D., Zhao, H.: Data security and privacy protection issues in cloud computing. In: International Conference and Privacy Protection Issues in Cloud Computing, pp. 647–651. IEEE Computer Society, Hangzhou (2012)
21. Dekkers, M., et al.: Data and Metadata Licensing, O.D. Support, Editor. European Comission (2014)
22. Walter, S.: Heterogeneous database integration in biomedicine. J. Biomed. Inf. **34**(4), 285–298 (2001)
23. Amit, S.P., Larson, J.A.: Federated database systems for managing distributed, heterogeneous, and autonomous databases. ACM Comput. Surv. **22**(3), 183–236 (1990)
24. Yannoukakou, A., Araka, I.: Access to government information: right to information and open government data synergy. In: 3rd International Conference on Integrated Information (IC-ININFO), vol. 147, pp. 332–340 (2014)
25. Tran, E., Scholtes, G.: Open Data Literature Review. Barkeley School of Law, University of California (2015)
26. Okediji, R.L.: Government as owner of intellectual property? Considerations for public welfare in the era of big data. Vanderbilt J. Entertain. Technol. Law **18**(8), 331 (2014)
27. Uhlir, P.F.: The Socioeconomic Effects of Public Sector Information on Digital Networks: Toward a Better Understanding of Different Access and Reuse Policies. National Research Council, Washington, DC (2009)

Presence Patterns and Privacy Analysis

Ella Roubtsova[1(✉)], Serguei Roubtsov[2], and Greg Alpár[1,3]

[1] Open University of the Netherlands, Heerlen, The Netherlands
{Ella.Roubtsova,Greg.Alpar}@ou.nl
[2] Technical University Eindhoven, Eindhoven, The Netherlands
s.roubtsov@tue.nl
[3] Radboud University, Nijmegen, The Netherlands
g.alpar@cs.ru.nl

Abstract. Business applications often use such data structures as Presence Patterns for presentation of numbers of customers in service-oriented businesses including education, retailing and media. Presence Patterns contain open data derived from internal data of organizations. In this paper, we investigate different ways of defining Presence Patterns and possible privacy consequences dependent on the chosen definition. The first contribution of the paper is a definition of a family of Presence Patterns. The second contribution is a method for privacy analysis of Presence Patterns.

Keywords: Presence Patterns · Privacy requirements
Method for privacy analysis of presence patterns

1 Introduction

Businesses nowadays provide valuable Internet services to their customers or employees which require registration of customers or employees. The registration data contain information about login and logout that can correspond to the 'presence' of registered users. In this paper, we assume that 'presence' online means actual or physical 'presence'[1]. This information about 'presence' can be used for different business analytics purposes. Business applications often use such data structures as Presence Patterns to analyse participation of customers in service-oriented businesses, including education, retailing and media.

However, businesses are not allowed to use personal data without limitations. The use of personal data is regulated by laws and, in particular, by the General

[1] The login and logout do not always correspond to the physical presence of the logged persons. Nevertheless, the degree of correspondence can be always established by manual control of presence during a chosen time period by sampling. Usually, this degree of correspondence is high, as many people use their mobile phones or PC to connect to the organizational network because this connection provides useful information.

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 298–307, 2018.
https://doi.org/10.1007/978-3-319-94214-8_21

Data Protection Regulation (GDPR)[2]. The major privacy rule, coming from the GDPR, allows businesses to use only the minimum data needed for the given task and does not allow them to relate business indicators to personal data of clients. The way to meet this privacy requirement is to pay attention to new business concepts used by business analytics, carefully design the data structures for all new business concepts and avoid relations with personal data in their definitions.

Because of the GDPR, companies (and organisations in general) have to comply with the privacy-by-design principles [2,7]. Businesses have a hard time to figure out how to do this. They also experience pressure from the technology side. Ever more possibilities and new opportunities emerge every day. However, these new technologies are often coupled with extensive data usage making hard to assess how privacy invasive their deployment can be.

In this paper, we investigate different ways of defining so-called Presence Patterns and possible privacy violations caused by their coupling with the organisation's files and also with, possibly relevant, publicly-available information. Section 2 presents the work related to protection of privacy. Section 3 defines a family of Presence Patterns. Section 4 defines a method for privacy analysis of Presence Patterns. Section 5 illustrates the use of the method with a case study. Section 6 presents conclusions and future work.

2 Related Work

2.1 Privacy, Anonymity and *k*-anonymity

Privacy studies conclude that there are quite a few wrong beliefs about privacy protection. Sweeney [14] reports that it is believed that removing explicit identifiers, such as name, address, telephone number makes data anonymous and protects privacy. However, the studies show that combinations of a few publicly available characteristics "uniquely or nearly uniquely identify some individuals".

The Internet contains many independent data sources. They may contain the same personal information with different extensions and, as a result, may partially release this personal information. In this sense, the Internet is similar to a multi-level database. It is formally proven by Su and Ozsoyoglu [13], that in general, it is impossible to guarantee privacy in a multi-level database due to functional dependencies and multi-level dependencies of data.

The public data can come from registers (phonebook-like databases), social networks (LinkedIn, Facebook, etc.), public ratings like on IMDb, Amazon or other websites and webshops. Moreover, we increasingly rely on infrastructures that collect, store and (possibly) publish data. For example, "smart" driver assisting services are aware of the current vehicle position and can potentially emit lots of data about your vehicle to the outside world. These data items may not be identifying in themselves, but when being aggregated, they can cause almost unforeseeable privacy violations.

[2] General Data Protection Regulation (GDPR)(EU) 2016/679 GDPR official: http://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32016R0679.

Sweeney [14] proposed a model "for understanding, evaluating and constructing computational systems that control inferences" about private information. The model is based on the definition of a quasi-identifier.
"Given a population of entities (a universe) U, an entity-specific table $T(A_1, \ldots, A_n)$, f_c: U → T and f_g: T→ U', where U ⊆ U'. A quasi-identifier of T, written Q_T, is a subset of attributes $\{A_i, \ldots, A_j\} \subseteq \{A_1, \ldots, A_n\}$ where: $\exists p_i \in U$ such that $f_g(f_c(p_i)[Q_T]) = p_i$".

In other words, a quasi-identifier Q_T of a table T is a subset of table's columns named by attributes $(A_1, \ldots A_j)$. A tuple of values of the attributes of the quasi-identifier, that appears in a row of table T, is a value or an occupance of the quasi-identifier. This value of the quasi-identifier is used for re-identification of an entity from the universe U.

Sweeney gives an example, of a voter-specific table (universe U) and a health patient-specific table (universe U'). In the voter-specific table, there are fields {Name, Address, ZIP, Birth date, Sex}. The medical data contain {Visit date, Diagnosis, ZIP, Birth date, Sex}. The quasi-identifier {ZIP, Birth date, Sex} can be used to re-identify the medical information of the voters. He concludes that "the attributes that appear in the private data and also appear in the public data are the candidates for linking: therefore, these attributes constitute the quasi-identifier and the disclosure of these attributes must be controlled". For such a control, he proposes an additional requirement of k-anonymity for any released table (publicly available table) T with an associated quasi-identifier.

The k-anonymity of any table T means that each tuple of attribute-values of the table's quasi-identifier Q_T appears at least in k-rows of this table $T[Q_T]$.

Let it be two tables, a private table PT and a public (released or open) table T. It is proven that, "if the released data T satisfies k-anonymity with respect to the quasi-identifier Q_{PT} of a private table PT, then the combination of the released data T and the external sources on which Q_{PT} was based, cannot link on Q_{PT} or a subset of its attributes to match fewer than k individuals" [14].

2.2 Inference Attacks and Open Data

The k-anonymity of open (and released) data cannot protect from specific data mining techniques, called inference attacks, which use data queries, aggregation of data, sorting, etc. However, the attacks demand resources. The open data should not make the life of an attacker easier.

The most representative sources of open data are the social networks. The attacks on social networks are described in [12]. This work states that only a lack of resources can stop attackers from massive crawling via API or "screen-scraping" [10]. Gross and Acquisti [6] demonstrate that the attributes of the nodes (rows in a entity-specific data table), such as social security numbers and other profile attributes [3], can be predicted with higher accuracy than random guessing.

Besides the nodes (rows in a data table), the social network expose edges, being relations between nodes. Edges provide additional information about nodes (persons) and their behaviour patterns [9].

Social networks provide some protection from attackers. Attackers need to create many dummy nodes helping to re-identify social network members. The online social networks "check the uniqueness of e-mail addresses, and deploy other methods for verifying accuracy of supplied information making creating of dummy nodes relatively difficult" [12].

Nevertheless, on small scale, for defined targets found in the released (open) data, the attacks can be feasible. So, the awareness about quasi-identifiers in any released data should be raised. In this paper, we talk about the released data in form of Presence Patterns.

3 Presence Pattern

In this section we define a family of Presence Patterns: a Simple Presence Pattern and several extensions of it for real-world applications. In the next section we use these definitions to propose a method for privacy analysis of such patterns.

3.1 Simple Presence Pattern

Presence of someone or something is the state of existing of someone or something in a given place in a given time interval. The given place can be described as a class room, a shop, but may also be seen as the state "online" in a media. Presence Patterns are often released for public. For example, a polyclinic may release a presence pattern as patients may choose a less visited day and time interval for their visit.

A Simple Presence Pattern in business usually concerns with numbers of present customers. As businesses are interested in performance and revenues, they are interested in the number of customers who are in the state "In" during a given time interval. They can estimate how many human and other resources are needed for service of customers. We define a Simple Presence Pattern as a mapping of numbers of present customers onto natural time intervals defined during a week.

```
Simple Presence Pattern = (Days, Hours, NumberOfPresent,
            function-times, function-presence)
```

- `Days={Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday}`.
- `Hours={1,2,3,....24}`.
- `NumberOfPresent` is a finite natural number (positive integer number or zero).
- `function-times: Hours` → `Days` divides days of the week into time intervals.
- `function-presence:NumberOfPresent` → `(Hours → Days)` assigns the numbers of customers to the time intervals for each day of the week.

3.2 Presence Pattern Extended with a Schedule

Any real Presence Pattern used for a given business purpose is usually an extension of the Simple Presence Pattern. The most widely used sources of extension are schedules.

```
Schedule = (Days, Times, OpeningStatus, function-times,
                    function-opening)
```

- Times= (Morning work hours 9:00-12:00,
  Evening work hours 13:00-16:00,
  Closing times 16:00-9:00);
- OpeningStatus ={Open,Closed}.
- function-times: Times → Days;
- function-opening = OpeningStatus → (Times → Days).

A schedule is easily included into the Presence Pattern.

```
Presence Pattern = (Days, Times, NumberOfPresent, OpeningStatus,
     function-times, function-opening, function-presence).
```

3.3 Presence Pattern and Schedule with Business Information

In practice, schedules contain also sets and relations [1] with business related information. For instance, in an educational institution, an address with the room number (say, "R206"), and a course name (for example, "Physics") and a teacher name usually correspond to some time interval.

```
Presence Pattern Education =(Days, Times, NumberOfPresent,
Locations, Courses, Teachers, function-times, function-location,
     function-course, function-teacher, function-presence).
```

- function-location = Address → (Times → Days)
- function-course = Courses → (Times → Days)
- function-teacher = Teachers→ (Times → Days)

This Presence Pattern Education contains private information about the service provider: addresses of an educational institute and the names of teachers.

3.4 Presence Pattern, Event Log and User Profile

Another data structure that is closely related to Presence Patterns is an event log. Industrial event logs are emitted by operating systems and applications to record, among other information, online events. The record of any occurrence of a login or logout of a network device always contain a time stamp and may contain private information like the user IP address or the MAC address of a mobile device or a PC, activity (associated or disassociated) with the time stamp. An event record often contains additional information such as the profile (or the access right) of the user (guest, employee, etc), protocol and so on.

A time stamp, an activity (state) and a User Identifier are often considered as the main fields of any industrial log.

Event log = (Time stamp, Activity, User, function-activity-time, function-user-time)

- Time stamp is a set of time stamps, for example, 12.07.2017.10:25.
- Activity = {In, Out}. For example, In means login or check-in; Out means Logout or check-out.
- User is a set of user unique IDs such as names or addresses of personal devices.

Event logs do not belong to any Presence Pattern, but they are used to derive the calculated information for Presence Patterns. For this, the event logs can be automatically processed in order to produce the NumberOfPresent. NumberOfPresent has a one-to-many relation with the users in the log. For every time interval in the schedule, the automated query derives the users logged in within this time interval and logged out within or after this time interval. If for a user there is a time stamp In before or within the given Time Interval i and there is a stamp Out within or after it, then this user is present in the given time interval and the counter NumberOfPresent is increased by 1. By counting, the private information is transformed into the statistical information, i.e. the numbers of customers per a time interval of the schedule.

The internal user (customer, employee) profiles of companies can be used to extend Presence Patterns for particular business purposes. For example, if a user profile contains fields User, Gender, Address, Date of birth, then the event log and the customer profile can be used to derive the NumberOfPresent people of a given gender, or of a given age or registered in a given city. In the extended form, a Privacy Pattern contains information about a subset of entities from a private universe (women, students following a given education, cars, citizens of a chosen city), that are present at a given time at a given place.

For example, the Presence Pattern Gender may be used to indicate the presence of women:

Presence Pattern Gender = (Days, Times, NumberOfPresent, Gender, Locations, Courses, Teachers, function-times, function-location, function-course, function-teacher, function-presence).

- Gender = {Male, Female};
- NumberOfPresentGender is a finite natural number (positive integer number or zero).
- function-presence: NumberOfPresentGender → (Times → Days) assigns the numbers of present woman to the time intervals for each day of the week.

Another example, Presence Pattern City (Fig. 1) shows a Presence Pattern extended with the information about present entities for a chosen home-city.

Presence Pattern City = (Days, Times, City, NumberOfPresentFromCity, function-times, function-presence-from-city)

- City = {Eindhoven, Uden}.
- NumberOfPresentFromCity is a finite natural number (positive integer number or zero).

– `function-presence-from-city: NumberOfPresentFromCity` → `(Times` → `Days)` assigns the numbers of customers to the time intervals for each day of the week.

4 Method for Privacy Analysis of Presence Patterns

We propose a method for privacy analysis of Presence Patterns. It is based on

1. our definition of a family of Presence Patterns given in Sect. 3 and
2. the fact, discussed in Sect. 2, that `k`-anonymity of a quasi-identifier in released data can be validated on internal data. The combination of the released data and the external data cannot link on this quasi-identifier to match fewer than `k` entities.

The value of `k` for `k`-anonymity is defined in advance, depending on the importance of privacy. A general 'rule of thumb' should be that an individual person could be determined from this k-subset with the effort more significant to the perpetrator compared to the privacy violation benefit.

Our method uses the following assumptions about public data:

(1) The Internal data universe contains more than `k` entities.
(2) Publicly available data structures may contain data fields with private information with the same data types as the company Internal data used for design of extended Presence Patterns;
(3) Publicly available data structures do not contain event logs of any other company-related activities of any entity.

Simple Presence Pattern does not have any Privacy Issues. A simple Presence Pattern (`SPP`) consists of the time schedule fields and the field, `NumberOfPresent`. The values of `NumberOfPresent` are derived from the `event log` that is not public. This field, `NumberOfPresent` is always related to more than `k` entities of the Internal data.

Privacy Analysis of Extended Presence Patterns. Let an Analysed Presence Pattern (`APP`) consist of the same fields as the `SPP` and a set `E` of some other fields.

1. Initially, the set of fields for privacy analysis is empty, `SF = ∅`. The set of fileds causing privacy violation is empty, `SV = ∅`.
2. Add a set `E` of fields in `APP` to `SF`, `SF = SF∪E`.
   - for each value `sf` of `SF`,
     - find in the Internal data the maximal value of `NumberOfPresent` and
     - find out if the maximal `NumberOfPresent` for a given value `sf` identifies less than `k` entities.
     - If YES, then add the quasi-identifier `E` to `SV: SV = SV∪E`.
3. If `SV≠ ∅`, then the release of this Presence Pattern can cost privacy violation, otherwise the Presence Pattern respects privacy and can be released.

This procedure gives a systematic way for privacy analysis of Presence Patterns using their specific structure.

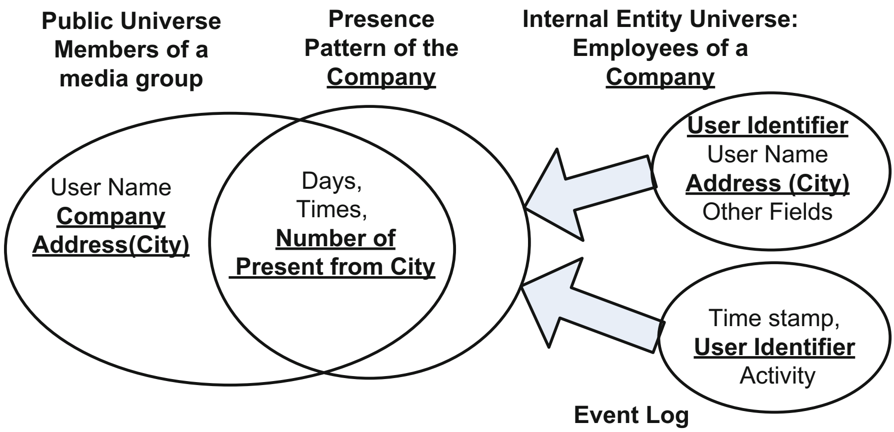


Fig. 1. Quasi-identifier in a presence pattern

5 A Case Study: Usage of Parking Space

Let us consider a case study. A company would like to monitor the use of the parking space in its premises. The entrance to the parking places has an automatic licence plate recognition system. The following internal (not public) data is available:

- an internal log of all cars that arrive to and leave from the parking. The log records include a time stamp and a licence plate number.
- an internal database of cars of employees. Any record in this database presents an employee's name and his (her) car licence plate number.
- the database of all employee's profiles. Among other fields, it contains the address of each employee (Fig. 1).

Further public information may be relevant to this use case:

- the LinkedIn (or other professional social network) profiles of employees of the company;
- The telephone guide, each record of which includes a person name, his (her) phone number (where the city can be identified);
- The public registry of cars containing the licence plate numbers with the corresponding brand and the model of the car, engine type, etc.[3]

In the investigation of the use of the parking space, the company creates new public data in the form of a series of Presence Patterns. The `k` for `k`-anonymity is set to `5` based on expert recommendations.

[3] This kind of data is publicly available in many countries (for example the RDW registry in the Netherlands).

1. First, the company derives and makes public the Simple Presence Pattern with the `Number Of Present Cars at the Parking Space` during the working hours and outside working hours. This pattern does not have any privacy issues as the `Number Of Present Cars at the Parking Space` is related to all employees of the company possessing a car (say to `500` employees).
2. Next, the company desires to analyze how many electric cars are parked daily in order to arrange the necessary amount of charging sockets. The internal database of coming cars has to be extended with the information about the type of the engine in order to be able to identify cars with electric engines. The presence pattern is extended with the indicator (or data type) `ElectricOrNot`. This information can be derived from the public car registry using the license plate number. The `Number Of Present Electric Cars at the Parking Space` during and outside the working hours is presented by the new presence pattern.
   The analysis reveals, that this presence pattern cannot identify less persons than the number of employees having electric cars (say, `120`).
3. The employees who live nearby the company premises, unlike the ones who live far away, may be convinced to avoid using cars. To be able to monitor its anti-pollution effort, the company needs to know how many distantly leaving employees are present during and outside the working hours.
   The pattern of presence is extended accordingly with the `City` taken from the addresses employees. So, the new pattern of presence shows the `Number Of Present Electric Cars at the Parking Space` from each `City`.
   The analysis shows that for at least one city (say, for Uden) the maximal `Number Of Present=2`. It is less than `k=5`. This pattern has a possible privacy violation if just one or a small number of employees (less than a predefined `k`) lives in a particular city.
   The quasi-identifier is `(Company, City)` (Fig. 1). From this pattern, one can derive new information, namely, a pattern of presence of persons working in the given company and living in the given city. This can be done using the combination of the public LinkedIn-like profiles of employees of the company and the public telephone guide.

This case study illustrates how the method identifies possible privacy violations caused by the released Presence Patterns.

6 Conclusion and Future Work

Privacy-by-design is a common goal in the field of information technologies. It is defined in [2] as seven fundamental principles: privacy as proactive, privacy by default, privacy embedded into design, full functionality (respecting other aspects of the system), end-to-end life cycle protection, respect for user privacy. These principles should be refined to methods of privacy analysis during design.

Following the principles ‘privacy as proactive’ and ‘privacy embedded into design’ we have proposed a proactive method for analysis of Presence Patterns released by data analytics. Our method is built on our inductive definition of a family of Presence Patterns and the refinement of privacy requirement by k-anonymity for the specific structure of Presence Patterns.

Presence Patterns are widely used by education institutions [4,11], by the television and radio companies, team sports [5], trends in tourism [8] and voting sites. Observing all these domains, we plan to investigate and evaluate the application of our method for pattern definition and privacy analysis on publicly available Presence Patterns and other released data structures derived from internal data of different organizations.

References

1. Aziz, N.L.A., Aizam, N.A.H.: A survey on the requirements of university course timetabling. World Acad. Sci. Eng. Technol. Int. J. Math. Comput. Phys. Electr. Comput. Eng. **10**, 236–241 (2016)
2. Cavoukian, A.: Privacy by design: the definitive workshop. A foreword by Ann Cavoukian, Ph.D. Identity Inf. Soc. **3**, 247–251 (2010)
3. Chew, M., Balfanz, D., Laurie, B.: (Under) mining privacy in social networks. Citeseer (2008)
4. Ganji, F., Budzisz, Ł., Debele, F.G., Li, N., Meo, M., Ricca, M., Zhang, Y., Wolisz, A.: Greening campus WLANs: energy-relevant usage and mobility patterns. Comput. Netw. **78**, 164–181 (2015)
5. Griffin, P.S.: Girls' participation patterns in a middle school team sports unit. J. Teach. Phys. Educ. **4**, 30–38 (1985)
6. Gross, R., Acquisti, A.: Information revelation and privacy in online social networks. In: Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society, pp. 71–80 (2005)
7. Hoepman, J.-H.: Privacy design strategies. In: Cuppens-Boulahia, N., Cuppens, F., Jajodia, S., Abou El Kalam, A., Sans, T. (eds.) SEC 2014. IAICT, vol. 428, pp. 446–459. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-55415-5_38
8. Kim, H., Cheng, C.-K., O'Leary, J.T.: Understanding participation patterns and trends in tourism cultural attractions. Tour. manage. **28**, 1366–1371 (2007)
9. Korolova, A., Motwani, R., Nabar, S.U., Xu, Y.: Link privacy in social networks. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, pp. 289–298 (2008)
10. Mislove, A., Marcon, M., Gummadi, K.P., Druschel, P., Bhattacharjee, B.: Measurement and analysis of online social networks. In: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, pp. 29–42 (2007)
11. Mulhanga, M.M., Lima, S.R., Carvalho, P.: Characterising university WLANs within eduroam context. In: Balandin, S., Koucheryavy, Y., Hu, H. (eds.) NEW2AN/ruSMART -2011. LNCS, vol. 6869, pp. 382–394. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22875-9_35
12. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: 2009 30th IEEE Symposium on Security and Privacy, pp. 173–187 (2009)
13. Su, T.-A., Ozsoyoglu, G.: Controlling FD and MVD inferences in multilevel relational database systems. IEEE Trans. Knowl. Data Eng. **3**, 474–485 (1991)
14. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**, 557–570 (2002)

Digitization Driven Design – A Guideline to Initialize Digital Business Model Creation

Tobias Greff(✉), Christian Neu, Denis Johann, and Dirk Werth

AWS-Institute for Digitized Products and Processes,
Uni Campus Nord 1, 66123 Saarbrücken, Germany
tobias.greff@aws-institut.de

Abstract. As a megatrend, the so-called Digital Transformation describes the entire potential of new technologies in economy. Thereby new approaches are needed to reveal this potential for companies. Looking at the literature a structured guideline is missing to reveal this potential by developing new digital business models (DBM) from the scratch. Therefore, this paper develops a structured framework to create ideas for DBMs. It contributes to business research by improving the initiation of digital reengineering. The resulting framework is called the Digitization Driven Design model (=D^3 model). The application of this approach is described with a consulting-company use case. Initial evaluations show the first promising expert feedbacks and possibilities for future research.

Keywords: Digitization Driven Design · D^3-model
Digital business model engineering · DBM design · DBM creation

1 Introduction

Today's world is largely changed by the digital transformation. The digital transformation brings fundamental changes in businesses and is primarily driven by new ICT technologies. This concerns changes in single but important value generating processes as well as strategic alignments of whole businesses. Companies are trying to keep up with the speed of digital transformation. More than 70% of organizations believe themselves to lack digital maturity and don't know where to start [1]. Therefore, they are all looking for methods to keep up to date with new ICT and at best be the first to disruptively renew their business. Hereby, the companies are facing a large problem: They are looking for an approach to identify their best way to start with the transformation of their own business and business model, but no tools exist with which they could start. The existing guidelines for "digital transformation of business models" [2–8] and also reviews about business model transformation strategies [9] address the steps beyond the ideation. They all address the value creation and the implementation by new or existing tools (for example with the business model canvas).

Therefore, this paper addresses the problem, how to get to a digital business idea and how to structure the ideation.

The research contribution of the paper consists of the scientific derivation of an extendable framework for ideation. It provides a guideline to decision makers and

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 308–318, 2018.
https://doi.org/10.1007/978-3-319-94214-8_22

officials facing the urging question of how to start DBM engineering and how to develop ideas for digital business models. We aim at developing larger idea sets in advance, evaluating them and select the best-performing idea out of several alternatives [10]. Afterwards the ideas should be developed with traditional business model tools.

To realize this, we follow an action research approach. First, we define our understanding of traditional and digital business models. We then use literature to identify drivers of digitization. These drivers are paradigms applicable to every domain and summarize newest approaches to create business models by state of the art technologies and their proper application. These serve as a basis to the Digitization Driven Design model (=D^3 model), a structured approach to create ideas for digital business models (DBM). The D^3 model is embedded into an easy-to-follow guideline for the design of digital business models. To illustrate the usefulness of our approach, we apply our guideline to a specific, challenging use case, namely digitization of consulting industry. Additionally, we evaluate our approach with a panel of innovation managers in a workshop session, discuss our results and give an outlook to future research.

2 Background

2.1 Business Models

The first question to address when talking about business model transformation is to define, what we mean with the term “business model”, since several definitions and understandings exist [11, 12]. In this work, we restrict ourselves to the definition that a **business model** (=BM) “describes mainly textual on a highly aggregated level—the business logic of an underlying company by a combination of interdependent offering, market, internal as well as economical business model components in a static and dynamic way beyond the company’s borders”. Furthermore, “it is not limited to a certain type of business or industry and is thus generally applicable and intended for internal as well as external addressees” [13]. The advantage of this definition is that it explicitly incorporates the generality of business model concepts as well as the transferability.

2.2 Digital Business Models

Accordingly, we then define our understanding of a **digital business model (DBM)** as a continuously evolving business model, where all essential value-generating and business-operating factors are enhanced or automated using digital technologies and which is steadily adjusted to changes in the economic and technical environment by concerning concepts to gain competitive advantages by deploying those technologies.

Generally, definitions of DBM vary largely in literature, and even reviews of IS literature draw different conclusions of what makes out a DBM [14–16]. By our definition we focus on the core aspect of digital transformation: the idea of disrupting processes by using new technology. The most common example mentioned here, which represents this, is Rolls Royce. The enterprise revolutionized the jet engine business by offering industrial products as services. By using digital technologies, they disrupted the business. Live maintenance and predictive inspection cycles as well as newest tracking systems allow to sell shear force instead of engines. The most important fact thereby was a change in business perspective concerning digital technologies – a DBM.

3 Drivers of Digitization

Drivers of digitization are paradigms applicable to every domain and summarize newest approaches to create business models by state of the art technologies and their proper application. They are concepts to gain competitive advantages by deploying novel information technology [3]. These concepts are applicable to multiple domains and are therefore best fitting for a digital transformation framework. The problem is, that these drivers are not collected and structured so far. Furthermore, they must be derived often from DBMs. This could be done, if similar approaches occur multiple times. Therefore, we had to analyze the existing literature and to reveal drivers of digitization, see Table 1. We divide into categories following the mediation strategies of [3]. Drivers of digitization either are to be positioned between a company and its customers, focus on company-internal processes or are assignable to business partners respectively. The separation between the three areas of concern is not perfectly sharp but helps to distinguish central aspects of how drivers of digitization can affect an individual business model. Furthermore, the separation helps to take a structured view on the drivers by a single perspective. In each perspective multiple drivers have a main area of impact.

Table 1. Summary of drivers

Reference	Drivers
Customers' area of impact	
[4, 5, 17, 18]	**Product to Service:** Offering customer centered IT-based services instead of products. Example: Rolls Royce offers airlines to buy usage time of turbines
[19, 20]	**Digitize Services:** Transfer to scalable partly or fully automatized 24/7 services. Example: Treatwell (digital service process for hairdressers)
[3, 21–23]	**Engage and Excite:** Digitally engage and excite customers by new bidirectional communication channels. Examples: campaigns of baby article resellers (Essentialbaby.com) with free digital contents for parents on Facebook
Internal area of impact	
[9, 24, 25]	**IT to the core:** Transfer value-adding processes to digital knowledge or algorithms. Example: Shazam (music recognition with an algorithm)
[5, 26, 27]	**Value adding data:** Analyze the potential of collectable and storable data and use it. Example: IBM sales optimization by data mining
[28, 29]	**Omni-Channel:** Use digital communication to connect customers and employees on their favorite channel. Example: Backbase – omni-channel banking

(*continued*)

Table 1. (*continued*)

Reference	Drivers
Partners' area of impact	
[30–32]	**Externalize Capabilities:** Realizing scalability with external resources. Examples: Crowd Sourcing Innovation Platform Innocentive
[3, 33–35]	**B2B Networks:** Use digital capabilities to strengthen network. Examples: Twail, the digital gastronomy labour agency
[2, 3, 21, 36]	**Platforms:** Building an eco-system of vendors and customers (paying both). Example: Apple Store – developers pay as well as customers

4 The D³ Model

So far, we identified the drivers and established their necessity in a digitization framework. However, we did not discuss yet how to actually implement them into a usable framework for ideation. To this extent, we introduce the Digitization Driven Design visualization (=D^3 model), see Fig. 1. Each driver identified in Table 1 is inserted into one of the three areas of impact of individual drivers. Additionally, the drivers are accompanied with important aspects that need to be addressed when the influence of single drivers on individual companies is evaluated.

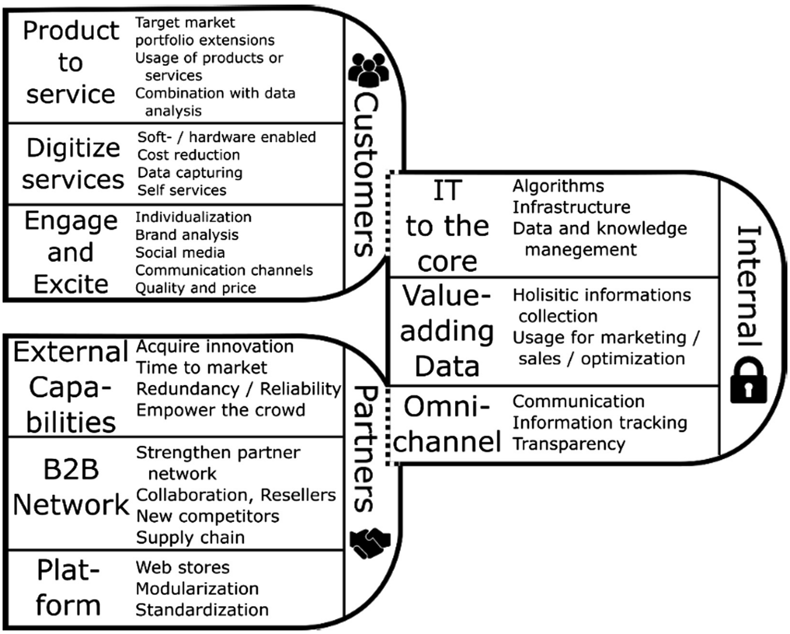


Fig. 1. D^3 model

4.1 Applying the D^3 Model

For the usage of the D^3 model, it is required to embed the model in DBM creation process. Transformation guidelines in the environment of creative processes as idea generation [37] or IS artefact Design [38] are considered as a basis for this guideline. The resulting "D^3-guideline" consists of four steps presented below, see Fig. 2.

Idea Generation: In order to use the D^3 model, the moderator guides through the idea generation. Starting by inspiring the participants when introducing single fields of the D^3 model – first by mentioning typical examples as we present them in our background analysis and second by asking one question for each principal driver by category s/he guides through the three parts of the model. Thereby, the order takes a tangential role. We suggest starting by the customers' area of impact, followed by the partners' area and at last the internal one. During the creativity process, it is the task of the moderator to motivate and empower the participants to create ideas for each topic and to map those ideas to a printed, drawn or projected image of the D^3 model. If the idea generation concerning one driver slows down, the driver should be changed.

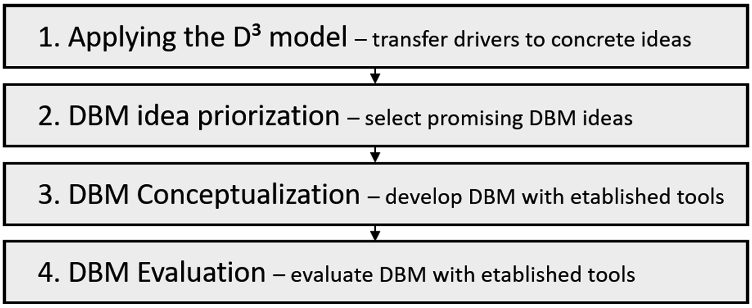


Fig. 2. Usage of the D^3 model in a DBM creation process

Prioritization: A finalized D^3 model consists of a large idea pool. Therefore, the ideas need to get ordered by relevance. It is recommended to have the ideas ranked by the team that developed the ideas and by a group of a comparable number of uninvolved colleagues. Existing sets of idea evaluation methods as the NAF technique or sticking dots could be extracted from idea management literature [39].

DBM Conceptualization: In the next step, the selected ideas are elaborated in business model templates, as for example in the proven business model canvas [8] or the business model framework by [4]. These templates enable more specific drafts of the initial ideas while at the same time considering the key parameters of business models as revenue streams: key partners and activities, communicative channels, used resources etc. A specific idea typically affects several if not all aspects of a business model. Since business model design is a time-consuming process, we recommend transferring only the most promising ideas into business models, based on the idea evaluation method applied to the D^3 model. For this purpose, one may either choose a threshold of a minimum score that an idea has to reach for being implemented as a business model, or a specific number of ideas is selected before the workshop, based on the resources available for business model design.

DBM Evaluation: Every single developed DBM has then to be evaluated. Current tools for evaluating business models are for example optimized for the business components reflected by service blueprints or the business model canvas [40]. As IS artefacts are frequently used, an iterative design and evaluation process is recommended.

5 Use Case – Digitization of IT Consulting

In order to validate our model and illustrate its usage, we applied our guideline to the business model of IT-consultancies. This effort has been done in collaboration with IT-consultants, domain experts and executives. As already discussed in detail by earlier publications [41], consulting is an economic branch that fostered the digitization of all industrial and economical areas, except that the consultants themselves merely use digital technologies. The digital transformation of consulting industry is still at preliminary stage, with the usage of ICT to just support the work of consultants, rather than assist or substitute parts of the consulting process. Figure 3 presents the results of discussions within a workshop with experts (six experts of the company: CTO, CDO, two unit leads and two innovation managers) in the areas of IT consulting and digitization. The aim was to find new ideas for digitization in the own field. Each idea was

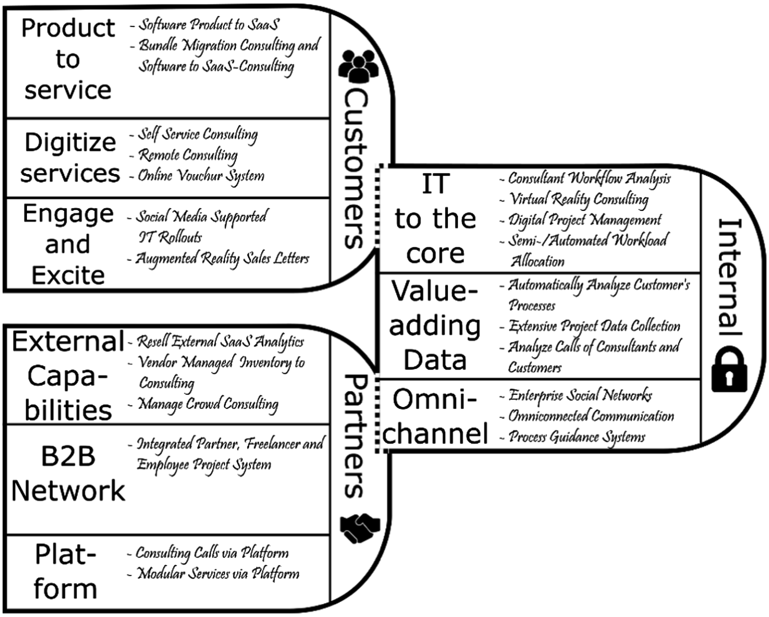


Fig. 3. D^3 model for a consultancy

evaluated by the experts and us regarding its novelty (N), attractiveness (A) and feasibility (F), respectively and the results were tested by uninvolved colleagues (six senior consultants). The results were averaged, and the ideas ordered according to the results, see Table 2. A fact to remark is that a low ranked idea is not generally a bad idea. The prioritization should primarily order the treatment. From the pool of ideas, the suggestion to offer modular consulting services has the largest average score. The novelty has a low grade compared to the other aspects, since consultancies typically already have in-house service catalogues that at least provide guidelines to the sales department during the setup of offerings. The attractiveness to both customers and consultancies is large, since modularization simplifies the preparation of offerings and the process of ordering a consulting service. And feasibility is large as well, since platform solutions to offer particular services already exist and it is not necessary to innovate a completely new platform concept.

The idea of offering modular services was developed to a business model and evaluated already in a previous publication [42]. Additionally, we developed a concept for Self-Service Consulting as a DBM [43].

Table 2. Prioritization of generated digital business ideas

n°	Idea	NAF_internal				NAF_uninvolved				NAF
		N	A	F	ø	N	A	F	ø	ø
1	Modular services via platform	9	9	8	8.7	8	9	6	7.7	8.2
2	Consulting calls via platform	8	8	9	8.3	8	7	8	7.7	8.0
3	Omni-connected communication	5	8	10	7.7	6	9	10	8.3	8.0
4	Self-service consulting	7	8	8	7.7	7	9	8	8.0	7.8
5	Extensive project data collection	6	8	8	7.3	8	8	7	7.7	7.5
…	…	…	…	…	…	…	…	…	…	…
22	Analyze calls of consultants and customers	3	5	3	3.7	2	6	4	3.7	3.7
23	Augmented reality sales letter	7	1	4	3.7	2	2	4	2.7	3.2

6 Initial Evaluation and First Feedbacks from Industry

After several internal iterative improvements, we started using the model in practice, to obtain first evaluated results. First, we used it for DBM ideation in the case of a German IT Consultancy as described in chapter 5. Our application to the use-case revealed both advantages and disadvantages of our guideline. In contrast to existing business model design frameworks and guidelines, we intentionally restrict ourselves to digitized business models and emphasize the initial step of driver identification. Here, the suggested framework simplifies the implementation of and transformation to digital business models by focusing explicitly on the early identification of triggers and their proper transformation into a business model.

Afterwards we used it to develop ideas for DBM in the domain of an energy trading company, a beauty products manufacturer and a print services enterprise. All cases resulted in internal innovation projects. With the aim of evaluating our model in more detail, we applied the D^3 model in a session of a "Digitization Masterclass" in Moscow. The Digitization Masterclass is an event for innovation managers to inspire and to learn about new innovation concepts. In this session six innovation managers of different companies took part in our workshop. They evaluated the model in a follow-up online questionnaire one week after the event as stated in Table 3. After further inquiries, the limited degree of freedom was stated as the main disadvantage of the model. The participants saw a large potential for domain transfer but doubted in parts that disruptive ideas could be structurally developed.

Table 3. Evaluation of the D^3 model in a "Digitization Masterclass Moscow"

Question	Yes	Partially	No	N/A
Do you think that the presented D^3-model is easy to understand?	6	-	-	-
You were presented examples illustrating the principles of the D^3-model. Were they helpful for you?	5	1	-	-
Do you think that the D^3-model is applicable to your industry/business domain?	1	4	-	1
Were you able to generate new ideas by the model?	5	1	-	-
In your opinion, do the generated ideas follow innovative approaches?	3	2	-	1
Are you planning to realize the generated ideas?	2	4	-	-
Would you recommend the D^3-model to others/colleagues?	4	2	-	-
Total	**26**	**14**	**0**	**2**

7 Conclusion

We applied the D^3 model in different industries and evaluated the usage of the model in a larger workshop with multiple innovation managers. The evaluation shows, that new ideas of the digital era can be easily identified in structured manner. Business model design and evaluation in our guideline is moved to separate design cycles, which is seen as a main advantage.

We understand the D^3 model as a generic toolset that can be easily extended by adding new drivers. This shows at the same time the main challenge and probably disadvantage: It is crucial to update the model regularly and to extend it by actual drivers. A solution could be to establish a technology scouting and to use a knowledge bases for drivers. But this raises further questions. Would it be just as easy to understand if we were to expand the number and scope of drivers? How could this be ensured?

This opens further and new research opportunities. First, the drivers could be extended to a maximum by integrating practice reports, web analysis and further literature. Furthermore, it would be necessary to apply the D^3 based digital transformation guideline to further use-cases. Application to new examples will reveal further advantages as well as flaws and drawbacks of our guideline besides those identified so far and can thus enhance the guideline iteratively. As a long-term goal, the success of the D^3 model could be evaluated by the success of generated DBM.

References

1. Kane, G.C., Palmer, D., Philips Nguyen, A., Kiron, D., Buckley, N.: Strategy, not technology, drives digital transformation. MIT Sloan Manag. Rev. **27**, 1–27 (2015)
2. Kalakota, R., Oliva, R.A., Donath, B.: Move over, e-commerce. Mark. Manag. **8**, 22–32 (1999)
3. Andal-Ancion, A., Cartwright, P., Yip, G.S.: The digital transformation of traditional business. MIT Sloan Manag. Rev. **44**, 34–41 (2003)
4. Johnson, M.W., Christensen, C.M., Kagermann, H.: Reinventing your business model. Strategy **86**, 57–68 (2008)
5. Bärenfänger, R., Otto, B.: Proposing a capability perspective on digital business models. In: 2015 IEEE 17th CBI, pp. 17–25. IEEE (2015)
6. Scheer, A.-W.: Thesen zur Digitalisierung. In: Abolhassan, F. (ed.) Was treibt die Digitalisierung?: Warum an der Cloud kein Weg vorbeiführt, pp. 49–61. Springer, Wiesbaden (2016). https://doi.org/10.1007/978-3-658-10640-9_4
7. Zott, C., Amit, R.: Business model design: an activity system perspective. Long Range Plan. **43**, 216–226 (2010)
8. Alexander, O., Yves, P.: Business model generation. Bus. Model Canvas Explan. Concept Bus. Model Canvas 89–107 (2010)
9. Zeng, Q., Li, X.: Evolution of E-business transformation strategy: a four dimension model. In: 2008 International Conference on Service Systems and Service Management, pp. 1–5. IEEE (2008)
10. Im, K., Cho, H.: A systematic approach for developing a new business model using morphological analysis and integrated fuzzy approach. Expert Syst. Appl. **40**, 4463–4477 (2013)
11. El Sawy, O.A., Pereira, F.: Digital business models: review and synthesis. In: Business Modelling in the Dynamic Digital Space. SDS, pp. 13–20. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-31765-1_2
12. DaSilva, C.M., Trkman, P.: Business model: what it is and what it is not. Long Range Plan. **47**, 379–389 (2014)
13. Burkhart, T., Krumeich, J., Werth, D., Loos, P.: Analyzing the business model concept—A comprehensive classification of literature. In: Thirty Second International Conference on Information Systems, pp. 1–19 (2011)
14. Timmers, P.: Business models for electronic markets. Electron. Mark. **8**, 3–8 (1998)
15. Al-Debei, M.M., El-Haddadeh, R., Avison, D.: Defining the business model in the new world of digital business. In: Proceedings of Americas Conference on Information Systems, pp. 1–11 (2008)

16. Veit, D., Clemons, E., Benlian, A., Buxmann, P., Hess, T., Kundisch, D., Leimeister, J.M., Loos, P., Spann, M.: Business models. Bus. Inf. Syst. Eng. **6**, 45–53 (2014)
17. Dubey, A., Wagle, D.: Delivering software as a service. McKinsey Q. **6**, 1–12 (2007)
18. Teece, D.J.: Business models, business strategy, and innovation. http://www.sciencedirect.com/science/article/pii/S002463010900051X
19. Fini, A.: The technological dimension of a massive open online course: the case of the CCK08 course tools. Int. Rev. Res. Open Distrib. Learn. **10**, 1–26 (2009)
20. Berman, S.J.: Digital transformation: opportunities to create new business models. Strateg. Leadersh. **40**, 16–24 (2012)
21. Weill, P., Woerner, S.L.: Optimizing your digital business model. MIT Sloan Manag. Rev. **54**, 71–78 (2013)
22. Herbert, J., Thurman, N.J.: Paid content strategies for news websites: an empirical study of British newspapers' online business models. Journal. Pract. **1**, 37–41 (2007)
23. Scott, D.M.: The New Rules of Marketing and PR: How to Use Social Media, Online Video, Mobile Applications, Blogs, News Releases, and Viral Marketing to Reach Buyers Directly. Wiley, Hoboken (2015)
24. Slywotzky, A.J.: Value Migration: How to Think Several Moves Ahead of the Competition. Harvard Business School Press, Brighton (1996)
25. Slywotzky, A.J., Morrison, D., Weber, K.: How Digital is Your Business?. Crown Business, New York (2001)
26. Baier, M., Carballo, J.E., Chang, A.J., Lu, Y., Mojsilovic, A., Richard, M.J., Singh, M., Squillante, M.S., Varshney, K.R.: Sales-force performance analytics and optimization. IBM J. Res. Dev. **56**, 8:1–8:10 (2012)
27. Lavalle, S., Lesser, E., Shockley, R., Hopkins, M.S., Kruschwitz, N.: Big data, analytics and the path from insights to value. MIT Sloan Manag. Rev. **52**, 21–32 (2011)
28. Wong, C.A., Laschinger, H.K.S.: Authentic leadership, performance, and job satisfaction: the mediating role of empowerment. J. Adv. Nurs. **69**, 947–959 (2013)
29. Brynjolfsson, E., Hu, J.Y., Rahman, M.S.: Competing in the age of omnichannel retailing. MIT Sloan Manag. Rev. **1**, 23–29 (2013)
30. Milovanović, G., Milovanović, S., Spasić, T.: The role of contemporary web technologies in supply chain management. Ekon. J. Econ. Theory Pract. Soc. Issues **62**, 43–58 (2016)
31. Yozgat, U., Demirbağ, O., Şahin, S.: The impact of knowledge sharing and partnership quality on outsourcing success. Int. Proc. Econ. Dev. Res. **63**, 50 (2013)
32. Chesbrough, H., Schwartz, K.: Innovation business models with co-development partnerships. Res. Technol. Manag. **50**, 55–59 (2007)
33. Rai, A., Patnayakuni, R., Seth, N.: Firm performance impacts of digitally enabled supply chain integration capabilities. MIS Q. **30**, 225–246 (2006)
34. Swafford, P.M., Ghosh, S., Murthy, N.: Achieving supply chain agility through IT integration and flexibility. Int. J. Prod. Econ. **116**, 288–297 (2008)
35. Ismail, S.: Exponential Organizations: Why New Organizations are Ten Times Better, Faster, and Cheaper than Yours. Diversion Books, New York (2014)
36. Rochet, J.-C., Tirole, J.: Platform competition in two-sided markets. J. Eur. Econ. Assoc. **1**, 990–1029 (2003)
37. Flynn, M., Dooley, L., O'Sullivan, D., Cormican, K.: Idea generation for organisational innovation. Int. J. Innov. Manag. **7**, 417–442 (2003)
38. Peffers, K., Tuunanen, T., Rothenberger, M., Chatterjee, S.: A design science research methodology for information systems research. J. Manag. Inf. Syst. **24**, 45–77 (2007)
39. El Bassiti, L., Ajhoun, R.: Toward an innovation management framework: a life-cycle model with an idea management focus. Int. J. Innov. **4**, 551–559 (2013)

40. Kayaoglu, N.: A Generic Approach for Dynamic Business Model Evaluation. Technische Universität, Berlin (2013)
41. Werth, D., Greff, T., Scheer, A.-W.: Consulting 4.0 - Die Digitalisierung der Unternehmensberatung. HMD Prax. der Wirtschaftsinformatik **53**, 55–70 (2016)
42. Johann, D., Greff, T., Werth, D.: On the effect of digital frontstores on transforming business models - concept and use-case from the consulting industry. In: Proceedings of the BMSD 2016, pp. 54–63 (2016)
43. Werth, D., Zimmermann, P., Greff, T.: Self-service consulting : conceiving customer-operated digital IT consulting services. In: Twenty-Second Americas Conference on Information Systems, San Diego (2016). Full Paper

Exploring Barriers in Current Inter-enterprise Collaborations: A Survey and Thematic Analysis

Nikolay Kazantsev[1,2(✉)], Grigory Pishchulov[1,3], Nikolay Mehandjiev[1], and Pedro Sampaio[1]

[1] Alliance Manchester Business School, University of Manchester, Manchester M1 3WE, UK
nikolai.kazantsev@postgrad.manchester.ac.uk, {grigory.pishchulov,n.mehandjiev, pedro.sampaio}@manchester.ac.uk

[2] National Research University "Higher School of Economics", 20 Myasnitskaya ul, Moscow 129000, Russia

[3] St. Petersburg State University, 7/9 Universitetskaya nab, St. Petersburg 199034, Russia

Abstract. Original Equipment Manufacturers are increasingly focusing on cooperation with a small number of risk-sharing partners who co-design and deliver key subsystems of the finished product. This trend increases collaboration activities throughout the supply chains involving suppliers of all sizes, including innovative small to medium-sized enterprises. The movement to Industry 4.0 concept such as "lot size of one" and demand-responsive production means these collaborations would be formed "on the fly" to respond to fast changing market needs and ever shorter product lifecycles.

Research has created models and approaches claiming to provide effective software support for such collaborations on demand (also known as instant virtual enterprises); however, these have yet to be implemented and widely applied by suppliers and manufacturers. Research literature, whilst still praising the theoretical advantages and transformative nature of dynamic value chains, is starting to note (as yet undisclosed) economic, managerial and technological concerns that impede uptake of these ideas.

In this paper we analyze exploratory interviews with a number of suppliers in the aerospace industry, and reveal key barriers such as lack of trust, switching costs, information asymmetry and path dependencies that prevent the uptake of short-term collaborations and present them in the sequence they appear forming supplier collaborations on demand.

Keywords: Aerospace supply chains · Collaboration networks · Industry 4.0

1 Introduction

Traditionally Original Equipment Manufacturers (OEMs) maintain links with hundreds and even thousands of suppliers along their value chains that result in multiple direct connections to procure the requested semi-parts [11, 13]. From the early 2000s OEMs

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 319–327, 2018.
https://doi.org/10.1007/978-3-319-94214-8_23

turn their attention to *design-and-build* strategies [3, 14] and invite suppliers to co-design components or even subsystems, coordinating clusters of multiple-tier small and medium-sized enterprises (SMEs) [16]. This shifts coordination and financial costs from the OEM onto networks that are expected to reduce lead-times [2, 10, 12], enable scale economies [13]; facilitate "on-demand" collaborations and make more business opportunities for SMEs available [4, 9]. To bring this vision into reality there are certain collaboration challenges that are considered in this paper in the aerospace industry, characterised by strong and dynamic collaborations spanning complex supply chains.

In this context the leading Multinational Aerospace Corporation (MAC) has initiated streamlining stages of their supply chain avoiding the need to directly link with over 6,000 suppliers, instead collaborating with around 200 trusted tier 1 suppliers that handle transactions with multiple SMEs clusters. One of those clusters, an Association of Aerospace SMEs (AAS), represents a wide spectrum of SMEs mainly supplying MAC. The Association plans to expand its portfolio and to collaborate with new partners for complex products and services in an easy way. Both parties must identify barriers that hamper collaboration and then develop mechanisms to erode them.

Section 2 defines the problem statement for AAS, Sect. 3 explains the methodology of our research, Sect. 4 delivers the main barriers to collaboration on-demand in the supply chains of MAC and the Sect. 5 opens discussion for a number of managerial implications of our findings and reflect on to how the perceived barriers can be addressed by combining technology and managerial solutions.

2 Problem Statement

Literature review suggests that SME participation in modern supply chains is currently facing several challenges, including perceived costs, risks, loss of flexibility and weakening of strategic independence [13], however, neither precisely systemises current barriers to collaboration nor proposes structures to classify them.

We argue that there is a significant gap between theory and practice and a need to assess key requirements from suppliers to facilitate the MAC vision. Although SMEs are already responsible for approximately 25% of patents (and therefore very important for MAC), direct collaboration with them is very costly and must be facilitated to enable further innovations into their aircrafts. Moreover, management of delivery ramp-up in time and establishment of new aircraft programmes require additional efforts for enabling short-term collaborations "on-demand" with supplier clusters, often realised in form of virtual enterprises [9].

In contrast to Glock et al. [5] and Zimmer et al. [15] that review approaches to collaborative supplier relationship management from the buyer's perspective, we address collaborations taking place between suppliers themselves to explain which obstacles exist from the suppliers' perspective, as perceived by SMEs in the aerospace industry, with a special focus on collaborations "on demand" requested by Industry 4.0 [2, 10, 12]. In the context of our work, comprehensive studies that analyse collaborations on demand for aerospace industry supplier networks have not yet been studied in depth. This raises the importance of identifying barriers to collaboration for aerospace SMEs and developing approaches towards addressing perceived barriers.

3 Methodology

The process model of mixed methods methodology comprising literature review, exploratory interviews, thematic analysis [1], information mapping is depicted in Fig. 1 and is explained in Subsects. 3.1–3.3. After we identify collaboration obstacles perceived by aerospace SMEs we map them to the stages of *instant virtual enterprise* formation [6] to identify at which point barriers to demand-driven collaboration occur and open discussion how to address them (further explored in the Subsect. 3.3). Since the dataset underpinning this study consists of input from 17 suppliers in the cluster it was possible to identify certain factors that affect each stage of collaboration on demand.

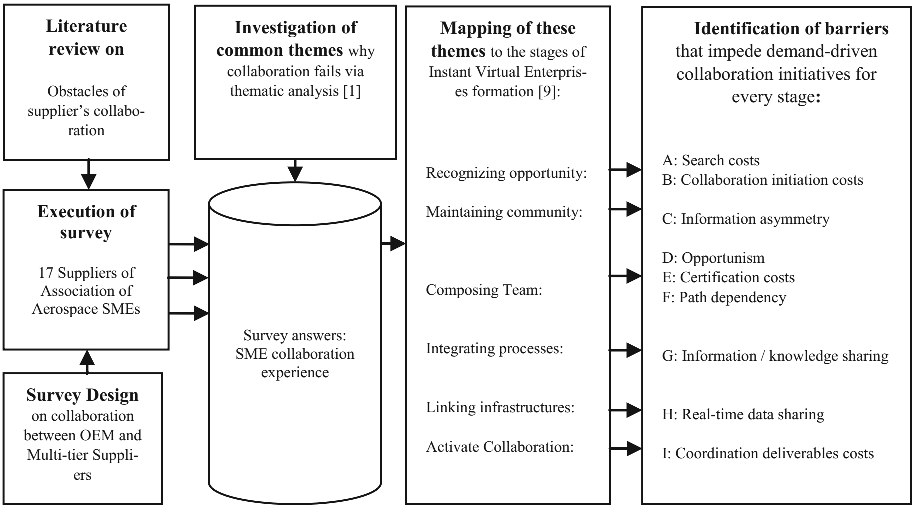


Fig. 1. The process model of research methodology

3.1 Survey

In order to explore the barriers that impede collaboration initiatives between manufacturing suppliers and OEMs we carry out exploratory interviews in an Association of Aerospace SMEs (AAS) introduced earlier in the Sect. 1. In order to construct a representative sample, we have requested AAS to provide access to their suppliers (SMEs) to investigate the main causes that explain why the potential collaborations between them fail. We received a positive response from 17 companies, all headquartered in Germany, which represents approximately 18% of AAS membership. These companies supply various products (e.g. composite aircraft parts) and services (e.g. product testing and environmental simulation, cabin engineering) to MAC.

The survey was held in German language with supplier top-level managers (companies' CEOs) during the time span from October 2016 to February 2017 and later translated into English. Responses were collected by an AAS executive via different channels: e-mail, fax, and phone. The semi-structured questionnaire comprised 20 questions, about company information (sector, size, volume, tier number); priorities and activities in searching for supply chain partners; barriers to vertical supply chain integration; their perceived impacts on business; barriers to horizontal collaborations and IT-related barriers to vertical and horizontal networking. Questions were open-ended in the sense that the respondents were able to indicate any number of items asked for in the respective question; furthermore, the respondents were invited to indicate the strength of the respective item on the Likert scale from 1 to 5, with 1 being least important. Some of the questions were pre-populated with a number of sample items based on initial information obtained from the respondent companies. These were included in the questionnaire to ensure a consistent perception of questions by the respondents and their responses. Examples of questions included in the questionnaire are indicated below:

- What are the main problems for integration of your company in supply chains of your customers (e.g. OEM, 1st tier suppliers)? What are the main impacts of those problems on your business?
- What are the main problems does your company have with collaboration of SME/organisations in networks?
- Which information deficits regarding Supply Chain Management (SCM) does your company have?

After data collection was finished the answers were translated to English for further analysis.

3.2 Thematic Analysis and Identification of Barriers

Amongst approaches to analyse survey data, *thematic analysis* (as a "method for identifying, analysing and reporting patterns (themes)" [1]) appears to be most suitable for exploring the meaning of unstructured data blocks such as the received survey results. King and Horrocks [8] define the word "theme" as follows: "Themes are recurrent and distinctive features of participants' accounts, characterising particular perceptions and/or experiences, which the researcher sees as relevant to the research question" (p. 150). To formalise barriers to collaboration we have followed the process from Braun and Clarke [1]: (1) Transcribe data, noting down initial ideas; (2) Code interesting features of the data in a systematic fashion across the entire data set, collating data relevant to each code; (3) Collate codes into potential themes (barriers), gathering all data relevant to each potential theme (barrier); (4) Check if the themes (barriers) work in relation to the coded extracts (Level 1) and the entire data set (Level 2), generating a thematic 'map' of the analysis; (5) Ongoing analysis to refine the specifics of each theme (barrier), and the overall story the analysis tells, generating clear definitions and names for each theme (barrier); (6) Selection of vivid, compelling extract examples, final analysis of selected extracts, relating back of the analysis to the research question and literature, producing a scholarly report of the analysis.

3.3 Classification of Barriers

To classify the identified themes/barriers we collate them into stages of collaboration on demand that are represented in the model of *instant virtual enterprise* formation [6, 9]. This model explains the behaviour of SMEs when they react to a new business opportunity (call for tender) from manufacturers and cannot fulfil this opportunity alone —in this case suppliers create a virtual entity to collaborate on demand. Once the new entity puts their bid and wins a tender, collaborative SMEs link their processes, infrastructures and further activate an instant virtual enterprise to respond to this business opportunity (Fig. 2).

This representation is very similar to the relationships between MAC and AAS. To open discussion on the erosion of the identified themes/barriers, once they were formed, they were grouped into higher-level factors and mapped to the formation stages from Fig. 2. The results of the mapping are presented in Table 1.

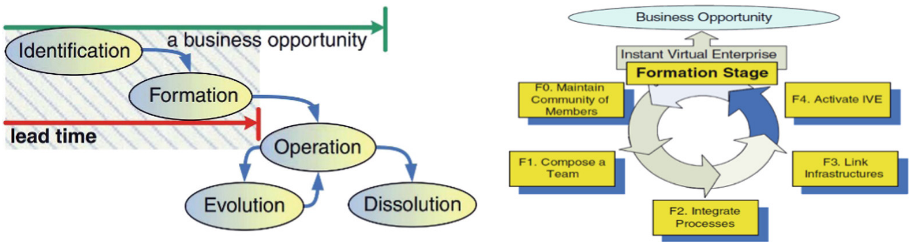


Fig. 2. Formation of *instant virtual enterprise* and its stages [9]

4 The Identified Barriers to Collaboration Between Suppliers and Aerospace Manufacturer

The survey has identified that one of the main intentions of SMEs is to integrate into the value chains of large-scale manufactures. It is caused by the intention to expand business portfolio and to get constant amount of orders, since many of them experience drawbacks due to timely missing, timely under-utilized capacities and constant overhead costs. Additionally, many SMEs are considered unfit for OEMs business opportunities due to company size, scaling capacity towards order volumes, scheduling capabilities, finance requirements and they request collaboration with the similar suppliers to cover gaps in capacities, capabilities and expertise to fulfil the purchase conditions of OEM.

However, in the reality such plans to collaborate often fails due to the issues depicted in Table 1.

Table 1. The identified barriers to collaboration

Formation steps	Main barriers and its manifestations
Identification of a business opportunity (derived from Call for Tenders)	*A: Costs of suppliers for searching, processing and storing information* 1. Lack of Call for Tenders visibility to SME suppliers 2. Information gaps due to vague CfT specifications 3. Long time to find applicable regulations 4. Information loss during communication ("Chinese whispers" effects) *B: Costs of suppliers for initiating collaboration* 1. Offline networking takes time to find new projects 2. SMEs' inability to exert marketing effort for attracting OEMs 3. Differences in legal, tax and patent systems
F0: Maintain community of members	*C: Information asymmetry* 1. Lack of knowledge regarding network collaboration 2. Lack of knowledge regarding to Industry 4.0
F1: Compose a team	*D: Opportunism of suppliers* 1. Extract benefits from collaboration, give nothing in exchange 2. Lack of trust, espionage for innovations 3. Unreliable partnerships (disruptions in deliveries) *E: Certification costs* 1. SMEs' inability to certify its market reputation 2. Expensive certifications: NADCAP, GRAMS, environment (ISO 9100, 16949) 3. Too complex accreditation processes and qualification checks 4. Overprotection of property rights: direct contracts *F: Path dependency of manufacturers* 1. OEMs do not intend to change their existing suppliers 2. OEMs disseminate CfTs to tier-1 suppliers only 3. OEMs dismiss the interests of domestic suppliers 4. OEM's inability to share risks with network partners

(*continued*)

Table 1. (*continued*)

Formation steps	Main barriers and its manifestations
F2: Integrate processes	*G: Challenges with Information and knowledge sharing* 1. Different data protection policies, information privacy 2. Delays in information sharing about demand changes 3. Lack of infrastructure knowledge: time-consuming calibration
F3: Link infrastructures	*H: Challenges with real-time data sharing* 1. Poorly structured data exchange policies 2. Use of proprietary IT without standardised data transfer 3. Variety of IT systems in use
F4: Activate Instant Virtual Enterprise	*I: Costs of leading supplier for coordination deliverables* 1. Missing standards and interfaces in communication 2. Problems signalled by the customer too late, quick fixes 3. OEM requests testing too late, deadline pressure

5 Discussion and Future Work

Our research shows limited collaboration along manufacturer value chains due to the list of factors depicted in Table 1. One of the main reasons (identified as the issue of distrust and competitive pressures) impedes major collaboration initiatives between SMEs operating across and within value chain tiers. To transform and improve competitiveness, companies need to adopt more cooperative interaction patterns that include collaborative governance rules, goal-based team composition and inter-organisational process coordination to lower transaction costs on short-term (demand-driven) collaboration.

Information asymmetry is another obstacle that restricts an access to the context information that is retained by an intermediary instead of propagating it to other players in the chain (e.g. order situation of the customer). Suppliers should be empowered with a digital platform that ensures the integrity of collaboration rules, transparent decomposition of large-scale tender descriptions into well-specified business process tasks and semi-automated supplier matchmaking to business process tasks via semantic technology and recommender systems.

Switching costs and path dependencies represent another area of problems that have to be solved in order to enforce collaborations along value chains. In particular in the area of data sharing, concerns are raised relating to the level of suppliers' access to shop floor data given the potential risk of cyber espionage and access to key corporate secrets. This in turn raises the prospect of a *data sharing paradox* [7] in Industry 4.0 where stewardship of the data is in the hands of companies that may have no interest in sharing information with collaborating partners. This paradox opposes the intention to develop mechanisms for customer order monitoring through highly visible production chains. Notwithstanding, effective data sharing is a precondition to generate "big data" scale datasets supporting efficient production and supply chain management. We are not aware of comprehensive studies that solve this data sharing paradox in Industry 4.0 and we aim to investigate how information privacy concerns between collaborating suppliers and OEMs is correlated with successful business opportunity delivery among other factors in our future works. Additionally, we aim to explain the rationale underpinning the identified collaboration barriers and predict potential supplier behaviour under different set of potential interventions in the value chain from the perspective of the Transaction Costs Theory.

Acknowledgements. This work was supported by the European Commission under the European Union's Horizon 2020 research and innovation programme (grant agreement no. 723336).

References

1. Braun, V., Clarke, V.: Using thematic analysis in psychology. Qual. Res. Psychol. **3**(2), 77–101 (2006)
2. Brettel, M., Friederichsen, N., Keller, M., Rosenberg, M.: How virtualization, decentralization and network building change the manufacturing landscape: an Industry 4.0 perspective. Int. J. Mech. Ind. Sci. Eng. **8**(1), 37–44 (2014)
3. Broft, R., Badi, S.M., Pryke, S.: Towards supply chain maturity in construction. Built Environ. Proj. Asset Manag. **6**(2), 187–204 (2016)
4. Cisneros-Cabrera, S., Ramzan, A., Sampaio, P., Mehandjiev, N.: Digital marketplaces for Industry 4.0: a survey and gap analysis. In: Camarinha-Matos, L., Afsarmanesh, H., Fornasiero, R. (eds.) PRO-VE 2017. IAICT, vol. 506, pp. 18–27. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65151-4_2
5. Glock, C.H., Grosse, E.H., Ries, J.M.: Decision support models for supplier development: systematic literature review and research agenda. Int. J. Prod. Econ. **193**, 798–812 (2017)
6. Grefen, P., Mehandjiev, N., Kouvas, G., Weichhart, G., Eshuis, R.: Dynamic business network process management in instant virtual enterprises. Comput. Ind. **60**(2), 86–103 (2009). https://doi.org/10.1016/j.compind.2008.06.006
7. Hoel, T., Mason, J., Chen, W.: Data sharing for learning analytics–questioning the risks and benefits. In: Proceedings of the 23rd International Conference on Computers in Education. Asia-Pacific Society for Computers in Education (Hangzhou), China (2015)
8. King, N., Horrocks, C.: Interviews in Qualitative Research. Sage, Thousand Oaks (2010)
9. Mehandjiev, N., Haemmerle, A., Grefen, P., Ristol, S.: Towards new frontiers: CrossWork. In: Mehandjiev, N., Grefen, P. (eds.) Dynamic Business Process Formation for Instant Virtual Enterprises. Advanced Information and Knowledge Processing, pp. 13–23. Springer, London (2010). https://doi.org/10.1007/978-1-84882-691-5_2
10. Müller, J., Dotzauer, V., Voigt, K.: Industry 4.0 and its impact on reshoring decisions of German manufacturing enterprises. Supply Management Research. ASSM, pp. 165–179. Springer, Wiesbaden (2017). https://doi.org/10.1007/978-3-658-18632-6_8

11. Pritchard, D., MacPherson, A.: Boeing's diffusion of commercial aircraft design and manufacturing technology to Japan: surrendering the US aircraft industry for foreign financial support. Occasional Paper (30), Canada–United States Trade Center (2005)
12. Smit, J., Kreutzer, S., Moeller, C., Carlberg, M.: Industry 4.0. European Parliament, Document No. IP/A/ITRE/2015-02 (2016)
13. Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E.: Designing and Managing the Supply Chain: Concepts, Strategies, and Cases, 3rd edn. McGraw-Hill, New York (2008)
14. Tiwari, M.: An exploration of supply chain management practices in the aerospace industry and in Rolls-Royce. Doctoral dissertation, Massachusetts Institute of Technology (2005)
15. Zimmer, K., Fröhling, M., Schultmann, F.: Sustainable supplier management – a review of models supporting sustainable supplier selection, monitoring and development. Int. J. Prod. Res. **54**(5), 1412–1442 (2016)
16. Link: European Commission. What is an SME? http://ec.europa.eu/growth/smes/business-friendly-environment/sme-definition_en. Accessed 10 May 2018

Smart Factory Modelling for SME

Modelling the Textile Factory of the Future

Michael Weiß[1(✉)], Meike Tilebein[1], Rainer Gebhardt[2], and Marco Barteld[2]

[1] Deutsche Institute für Textil- und Faserforschung Denkendorf, Denkendorf, Germany
michael.weiss@ditf.de
[2] Sächsisches Textilforschungsinstitut e.V., Chemnitz, Germany

Abstract. The Industrial Internet of Things and Industry 4.0 both promote Smart Factory as the manufacturing of the future. The complexity of production environments consisting of smart entities is far more complex than traditional production scenarios. In particular SMEs have problems with the transition towards Smart Factory. Modelling could support this process. Available modelling frameworks are either too abstract for SMEs and their production networks or far too detailed to be suitable for discussing Smart Factory approaches within the companies and their production networks. A suitable modelling framework has been developed in a German research project on modelling the textile factory of the future, funded by the German Federal Ministry of Education and Research. The smart entity is the core element of the framework, which is detailed by five sub-models. This entity now can be used to describe machines, services as well as complete production networks with the same building blocks. Textile SMEs can engineer their structures towards Smart Factory with a modelling attuned to their needs. The capabilities of the modelling framework are demonstrated by their application to a carbon fibre non-woven fabric production process.

Keywords: Modelling · Smart Factory · Smart entities

1 Introduction

The Smart Factory as manufacturing structure of the future consists of active, autonomous and self-controlling production units, which we refer to as smart entities. In the future, production in a Smart Factory will be realised with ad-hoc networks of smart entities. This approach requires significant changes to the production structures but also to the way of communication between smart entities. The necessary business process redesign is a challenge especially for SME of the textile industry.

Modelling is an appropriate instrument to support business process redesign for Smart Factory. Various modelling frameworks have been developed to support the design and implementation phase. However, the pre-design phase, which is not supported by any framework, is essential for strategic decisions of textile SME. With limited resources available, an early overview about structural changes, future

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 328–337, 2018.
https://doi.org/10.1007/978-3-319-94214-8_24

interaction between internal smart entities and network partners (supplier, service provider but also customer) is necessary.

The German research project "Modellierung Textilfabrik der Zukunft" (modelling the textile factory of the future), funded by the German Federal Ministry of Education and Research, is addressing this demand of a modelling framework for the pre-design phase of business process reengineering. In this project a domain-specific modelling framework and modelling language have been developed that assist the textile industry to change their structures towards the principles of the Smart Factory. The resulting framework has been tested with state-of-the-art textile production processes.

The paper is providing a short overview of the basic concepts for a textile Smart Factory (Sect. 2) followed by a state-of-the-art analysis of modelling frameworks developed within the context of Industrial Internet of Things (IIoT) and "Industrie 4.0" (Sect. 3). The resulting demands for a modelling framework and the requirements are presented in Sect. 4. Section 5 describes the developed modelling framework for the pre-design phase of business process reengineering in the textile industry. The practical application of the framework in a textile production process is described in Sect. 6. Section 7 provides a summary of the results and identifies future work.

2 Flexibility and Smart Factory

The capability to react fast and flexibly on customer requests is a core requisition for enterprises in the next years. In a survey about future production 98.6% of the participating companies labelled this aspect as important or very important [1]. The demand for organizational flexibility has been around for a long time. The design fields and principles for flexible organization identified by Brehm [2] are still relevant today. The principles of organizational slack, modularisation, loose coupling, self-organisation and learning organisations have become even more relevant. The IIoT and "Industrie 4.0" promote these principles and connect them with new technical developments like cyber-physical systems [3, 4].

The German high-tech vision "Industrie 4.0" describes a paradigm shift from rigid production with hierarchical-rigid control structures to flexible structures consisting of active, autonomous and self-controlling production units [5]. These production units form the Smart Factory by autonomously interacting with other production units and the resources required for manufacturing smart products. Core concepts to realise the Smart Factory are [4, 6, 7]:

- horizontal integration through value networks,
- vertical integration and networked manufacturing systems, and
- end-to-end digital integration of engineering across the entire value chain.

These concepts are flanked by further developments of cyber-physical systems and research of new social infrastructures in the workplace. The technical backbone for "Industrie 4.0" is formed by the IIoT. It consists of a complex system of cyber-physical systems. A major challenge is to provide an extensible architecture that simultaneously sup-ports a variety of hardware like sensors, actuators and software systems and retains manageable complexity [8]. One approach is to dissolve the classical automation pyramid and provide a network of services, which is similar to the network of production units and can be described in a similar way.

The transition towards a Smart Factory is a difficult task especially for SMEs of the textile industry. These companies are characterized by flat hierarchies, complex and highly optimized production structures as well as cross-domain knowledge demand. Qualified personnel and financial resources for change processes are very limited. Therefore it is important that supporting methods must be easy to learn and provide a significant complexity reduction. The necessary changes for a transition process involve all areas of the company as well as relations with network partners. Both technical and organisational aspects must be addressed. Therefore, suitable business process reengineering methods such as modelling are required to support the SMEs of the textile industry to evaluate new structures and the necessary transformation process.

3 Modelling Architectures

A variety of modelling frameworks has been developed to support Smart Factories. Weyrich [9] provides a short overview of frameworks that have been developed in close collaboration between research and industry.

The "Referenzarchitekturmodell Industrie 4.0" (RAMI 4.0), promoted by the "Plattform Industrie 4.0", is the best-known among these modelling frameworks. It describes the landscape of smart factories in three dimensions [10]. The first dimension explains the hierarchy of the factory from the product and field device level up to its ecosystem. This approach is contradictory to the future nature of factories with flexible elements linked across all hierarchy levels and communication between all participants. The second dimension visualises the architecture ranging from the real world asset up to functional and business aspects in the virtual world of this asset. The last dimension describes the product life cycle from development to the use phase.

Another approach was developed by the Industrial Internet Consortium. The "Industrial Internet Reference Architecture" (IIRA) comprises the following concepts [11]:

- Business-value-driven and concern-resolution-oriented
- Four viewpoints: business, usage, functional and implementation
- Identification and classification of common architectural concerns into the viewpoints
- Systematic analysis of concerns
- Identification of solutions for concerns

Therefore it shares some aspects with RAMI 4.0 like the product life cycle, assets, function and business viewpoint.

A third reference architecture was developed within the European Research Project "Internet-of-Things Architecture". It consists of a series of sub-models describing the domain, information, functions and communication as well as trust, security and privacy. The architecture now provides views (functional, information, deployment & operation) and perspectives (evolution & interoperability, performance & scalability, trust, security & privacy, availability & resilience) on these sub-models [12].

The fourth approach is derived from the model-driven systems engineering. On the one hand it can alleviate the challenges arising from the number of domains, stakeholders and modelling techniques required. On the other hand it "*... introduces the challenge of managing the multitude of different artifacts, such as configuration, models, templates, transformations, and their relations ...*" [13]. The complexity of the systems engineering process is reduced by using artefacts from the various domains involved describing certain aspects of the Smart Factory. These artefacts can then be connected depending on their nature and the context they are used in. To manage the variety of artefacts Butting [13] proposes to provide an artefact model that precisely specifies the kinds of artefacts, tools, languages and relations present in a model-driven systems engineering project.

In addition to these approaches there are a number of further modelling approaches mainly dedicated to specific aspects of smart factories like the communication between cyber-physical systems or the linking of real world objects with virtual representations.

4 Modelling Demand and Requirements

All reference architectures and modelling approaches support the design and implementation phase of the business process reengineering required for the transition towards a Smart Factory. However, there is a lack of assistance for the preparation phase. Topics [14] like

- the assessment of the potential for reengineering,
- definition of boundaries and the scope of the appropriate process and
- obtaining and maintaining management commitment

are not covered or only rudimentarily supported. Additionally the complexity of the modelling reference architectures is still high for many SMEs, especially for companies of the textile industry with a majority of companies with less than 100 employees. For example RAMI 4.0 forms a of cube 7 hierarchies, 6 layers and 4 life cycle steps resulting in a total of 168 possible sub-models. The other reference architectures share the same issue. The lack of corresponding modelling languages is another important consideration. Many SMEs have modelling capabilities, which are generally restricted to common modelling languages like BPMN, SysML or EEPC.

Therefore, a modelling framework and a corresponding modelling language are required to support SMEs in the first phase of the business process reengineering towards a Smart Factory. The framework should be able to support the modularisation of the factory with an abstraction level suitable for evaluation of reengineering potential, supporting the definition of boundaries and scope of the modularised processes as well as communicating the reengineering vision towards the management.

The German research project "Modellierung Textilfabrik der Zukunft" (modelling the textile factory of the future), funded by the German Federal Ministry of Education and Research, is addressing this methodological gap in business process reengineering. In this project a domain-specific modelling framework and modelling language have been developed that assist the textile industry to change their structures following the principles of Smart Factories. Core element is the smart entity as virtual representation of an intelligent production unit. It encapsulates all tangible and intangible resources, rules, adaptation capabilities and activities necessary to provide a certain service or product in a dynamic environment. A smart entity can be used in various approaches like agent-based, rule-based or object-oriented systems [15] to provide the necessary information required for autonomous production in a smart factory.

The design and development of the modelling framework as well as the corresponding language have to consider a number of requirements. The requirements can be grouped in formal aspects (which are essential for analysis and transformation of models), user-specific aspects (representing the demand of the stakeholders) and, finally, the domain-specific aspects (describing requirements derived from the domain) [16].

The following user-specific and domain-specific requirements had to be considered in the design.

User-Specific Requirements:

- *Stakeholder:* Main stakeholders are the management of the company. Also involved are domain experts from various departments. In addition to this internal perspective on the company, the surround ecosystem also has to be involved in the redesign. Therefore, value chain partners as well as service providers must be incorporated in the first phase of the business process reengineering.
- *Level of abstraction:* The details have to be in line with the topics covered and the stakeholders addressed by the framework. The components available for modelling represent types and not ready to implement structures. For example an information object will be addressed as type order, but no detailed information about the data structure will be provided.
- *Complexity:* Both, the modelling framework and the modelling language must be simple. Thus the numbers of sub-models to describe a smart entity and also the number of modelling blocks have to be limited to reduce the learning curve for stakeholders.

Domain-Specific Requirements:

- *Topics covered:* The first question covered by the modelling framework is the design of organisational and network structures consisting of smart entities. It should be suitable for first discussions about the future structures. Technical details are not important. The second question considers a more detailed view on modules/smart entities. A smart entity comprises structure information about its internal activities, its assets and information pool as well as the rules applied in its activities. Management activities also must be integrated to become smart. A smart entity is not a standalone module, as it must interact with many other entities. Therefore, describing the basic structure of interfacing is essential.
- *System boundary:* The modelling framework concentrates on organisational and network structures. Flexibility measures like work time models, learning and qualification are not in its scope. Also, its focus is on the pre-design phase. Therefore aspects like IT-System design or implementation are not covered.

For the implementation of the modelling framework and modelling language a number of design decisions have been made. The first decision was inspired by the modelling of cyber-physical systems with artefacts [13]. Smart entities are described with an artefact model. This leads to the second decision that the description of smart entities is independent from its instance type. Namely, a smart entity representing a production machine is modelled with the same framework and modelling language as a smart entity representing a value chain. The last one concerns the type of representation of models. In a first step only a graphical representation is realised. In the evaluation of the models only human beings are involved and no automatic model processing is required.

5 Modelling Framework

Based on the identified requirements and the selected design principles, a domain-specific modelling framework for the textile industry has been elaborated. The core element is the smart entity. It describes a closed functional unit with dedicated interfaces and the ability to operate in a smart manner. It contains all tangible and intangible resources required to fulfil its functional purpose that are not provided externally. An instance of a smart entity can be a cyber-physical system, a service or even a network of smart entities.

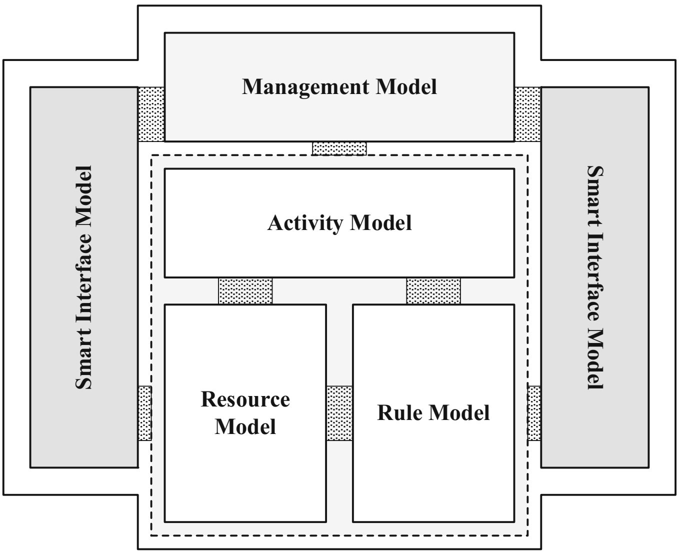


Fig. 1. MTFZ modelling framework for Smart entity

To describe a smart entity three functional aspects must be modelled (see Fig. 1).

One functional aspect represents the internal structure of a smart entity. Three sub-models, grouped with a dotted line, represent the available resources ("resource model"), the activities ("activity model") carried out and the rules ("rule model") applied.

The **"resource model"** provides elements to describe all required tangible and intangible (e.g. machines, information or raw material) resources including personnel. These resources are the starting point for the value creation activities of the Smart Entity.

The **"activity model**" now characterises the value creation. Core element is the transformation, which describes the transition from various tangible and intangible inputs to an output good using the resources of the smart entity. For example the transition from fibre batt and semi-finished textile to a nonwoven product but also the transition of information to provide services. The business logic of the value creation used in the "activity model" is specified by rules.

The **"rule model"** supplies all applicable rules for the value creation process but also for interfacing and smart entity management. There it provides various elements to describe business rules. Conditions, conclusion and events structure the actual flow of activities. Other types of rules like guidelines or calculations support individual activities. A special type of rule is authorisation limiting access to certain activities and resources.

Another functional aspect addresses the smart entity's communication with its environment. The sub-model **"smart interface model"** describes the exchange activities with other smart entities. It consists of elements to address various types of tangible and intangible resources (see also "resource model") exchanged and elements to describe the essential activities (negotiation of exchange rules and the exchange processes) for interfacing. It enables the smart entity to interact with its environment.

The last functional aspect is responsible for providing the smartness of the entity. Management activities changing the internal structure and the interface to the environment are described in the **"management model"**. Core elements are the analysis and the design activity. The analysis activity can evaluate single and group of components of a smart entity and based on the analysis results can trigger a design activity, which is able to restructure the smart entity to cope with changing environmental and business requirements. This capability distinguishes the smart entity from normal modularized production.

All modelling language element can be grouped to provide reusable sets for easy communication and modelling. Each sub-model has been attuned to the complexity of SMEs, to the involved stakeholders and the domain requirements of the textile industry. The sub-models are all interlinked (e.g. via the resource and rule elements) and enable the visualisation of all essential elements for the first phase of business process reengineering. Figure 2 provides an excerpt of the modelling language concept for the "smart interface model".

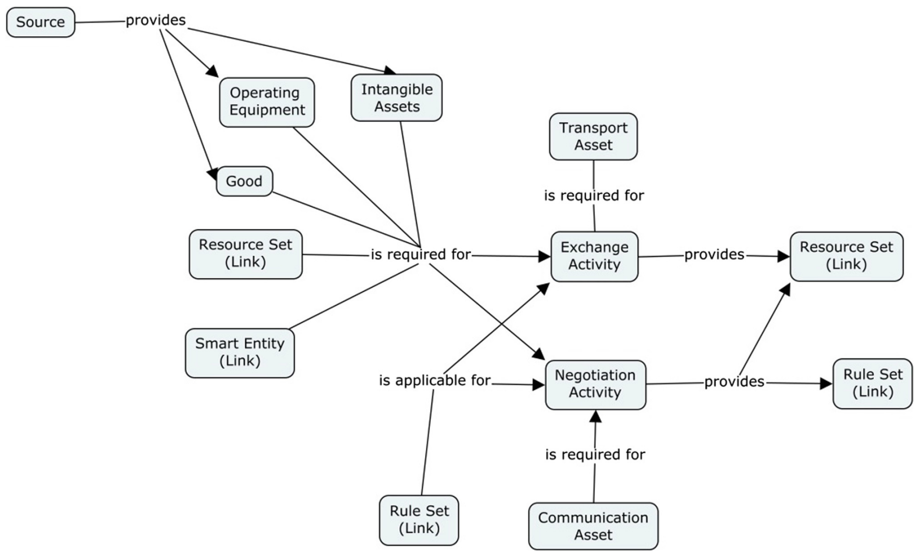


Fig. 2. Excerpt of modelling language concept for “smart interface model”

6 Application Example

The elaborated modelling framework and the associated modelling language have been used to describe complex textile production lines that are already fit for future requirements. One of the application cases was a nonwoven production using carbon fibre waste. For this application, a production line with state of the art cyber-physical and communication systems at STFI’s Center for Textile Lightweight Engineering was used as starting point.

Starting with the fibre opening unit and a fibre mixing plant, the production line offers multiple options to produce nonwovens. One variant is based on a random laid web made by the Airlay unit. Another variant relies on a classical carding machine followed by a cross lapper. Finally, pre-needled nonwovens are produced by means of a needle-punching machine. The production line consists of further steps of that were not considered in the following. The resulting nonwovens are basic materials for the production of test specimens and components in the form of thermoplastic or thermosetting composites based on a wide variety of technologies. The composite formation will be performed in a second step. An integrated testing laboratory, which is designed for the special needs of lightweight textile structures, completes the Center for Textile Lightweight Engineering.

To transform this nonwoven production line to a smart factory the various production steps have to transition to smart entities. That means each production step has to form an autonomous intelligent unit. A concept for the new structure has been elaborated using the modelling framework and modelling languages developed. Activities, rules and resources as well as their dependencies have been identified and documented. Figure 3 shows a section of the resulting “activity model” for the smart factory “nonwoven production”. The capabilities provided by the framework and modelling languages proved to be sufficient for the first phase of the business process reengineering.

For testing and validation the Guidelines of Modelling from Becker [17] have been used. In the testing scenarios the fulfilment of the user-specific and domain-specific requirements was considered.

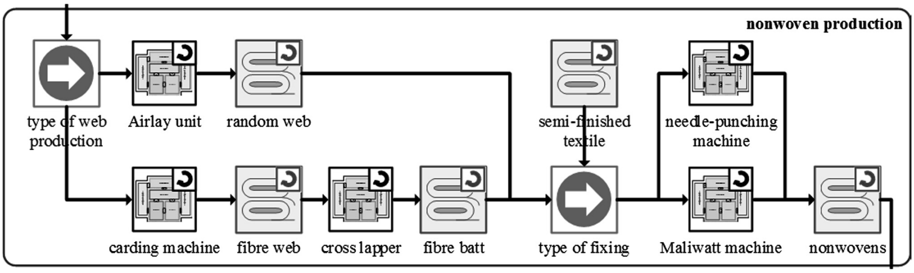


Fig. 3. A section of nonwoven production network consisting of smart entities, choices and the product exchanged

The testing and validation activities have verified that the proposed modelling approach is suitable for describing future textile production structures. The first phase of the business process reengineering can be supported sufficiently by the modelling framework and its modelling language. Discussions of the principle design are supported. Also the requirements of the textile domain have been met. All structures occurring in the textile industry can be visualised. The complexity of the framework is appropriate for the targeted stakeholders.

7 Summary and Conclusion

The demand of a modelling framework for the pre-design phase of business process reengineering has been addressed successfully by the German research project on modelling the textile factory of the future. The domain-specific modelling framework and modelling language developed within the projects assist the textile industry to change their structures towards the principles of the Smart Factory. The resulting framework has been tested successfully with state-of-the-art textile production processes.

In the next step the results of the pre-design phase must be transformed for use in the design and implementation phases of the business process reengineering. A support of various modelling frameworks is possible. Also a transition of the modelling framework to other industrial domains, especially for industries using textile products, is targeted.

References

1. Ganschar, O., Gerlach, S., Hämmerle, M. et al.: Produktionsarbeit der Zukunft: Industrie 4.0. Fraunhofer-Verlag, Stuttgart (2013)
2. Brehm, C.: Organisatorische Flexibilität in Wertschöpfungsnetzwerken. In: Bach, N., Buchholz, W., Eichler, B. (eds.) Geschäftsmodelle für Wertschöpfungsnetzwerke: Wilfried Krüger zum 60. Geburtstag, 2. Auflage. ilmedia, Ilmenau, pp. 80–100 (2010)
3. Lee, J., Kao, H.-A., Yang, S.: Service innovation and smart analytics for industry 4.0 and big data environment. Procedia CIRP **16**, 3–8 (2014)
4. Gilchrist, A.: Industry 4.0: The Industrial Internet of Things. Springer, New York (2016). https://doi.org/10.1007/978-1-4842-2047-4
5. Bischoff, J., Taphorn, C., Wolter, D., et al.: Erschließen der Potenziale der Anwendung von "Industrie 4.0" im Mittelstand: studie im Auftrag des Bundesministeriums für Wirtschaft und Energie (2015). https://www.bmwi.de/Redaktion/DE/Publikationen/Studien/erschliessen-der-potenziale-der-anwendung-von-industrie-4-0-im-mittelstand.html Accessed 28 Feb 2018
6. Kagermann, H., Helbig, J., Hellinger, A., et al.: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0: Deutschlands Zukunft als Produktionsstandort sichern; Abschlussbericht des Arbeitskreises Industrie 4.0 (2013). https://www.bmbf.de/files/Umsetzungsempfehlungen_Industrie4_0.pdf. Accessed 28 Feb 2018
7. Radziwon, A., Bilberg, A., Bogers, M., et al.: The smart factory: exploring adaptive and flexible manufacturing solutions. Procedia Eng. **69**, 1184–1190 (2014)
8. Jeschke, S., Brecher, C., Meisen, T., et al.: Industrial internet of things and cyber manufacturing systems. In: Jeschke, S., Brecher, C., Song, H., et al. (eds.) Industrial Internet of Things: Cybermanufacturing Systems, pp. 3–20. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-42559-7_1
9. Weyrich, M., Ebert, C.: Reference architectures for the internet of things. IEEE Softw. **33**(1), 112–116 (2016)
10. Hankel, M., Rexroth, B.: The reference architectural model industrie 4.0 (rami 4.0) (2015). http://www.control.lth.se/media/Education/EngineeringProgram/FRTN20/2016/ZVEI-Industrie-40-RAMI-40-English.pdf. Accessed 12 Mar 2018
11. Lin, S.-W., Murphy, B., Clauer, E., et al.: Architecture alignment and interoperability: an industrial internet consortium and plattform industrie 4.0 joint whitepaper (2017). http://www.plattform-i40.de/I40/Redaktion/DE/Downloads/Publikation/whitepaper-iic-pi40.html. Accessed 1 Mar 2017
12. Bauer, M., Boussard, M., Bui, N., et al.: Updated reference model for IoT v1.5: deliverable D1.3 (2012). http://www.meet-iot.eu/deliverables-IOTA/D1_3.pdf. Accessed 1 Mar 2018
13. Butting, A., Greifenberg, T., Rumpe, B., et al.: Taming the complexity of model-driven systems engineering projects. Beitrag zu Grand Challenges in Modeling 2017 Marburg (2017). http://www.informatik.rwth-aachen.de/go/id/lolj/file/699244. Accessed 1 Mar 2018
14. Attaran, M.: Exploring the relationship between information technology and business process reengineering. Inf. Manag. **41**(5), 585–596 (2004)
15. Hopgood, A.: Intelligent Systems for Engineers and Scientists, 3rd edn. CRC Press, Boca Raton (2012)
16. Frank, U., van Laak, B.: Anforderungen an Sprachen zur Modellierung von Geschäftsprozessen (2003). https://www.wi-inf.uni-duisburg-essen.de/FGFrank/documents/Arbeitsberichte_Koblenz/Nr34.pdf. Accessed 6 Mar 2018
17. Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of business process modeling. In: van der Aalst, W., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 30–49. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45594-9_3

Configuring Supply Chain Business Processes Using the SCOR Reference Model

Emmanuel Ahoa, Ayalew Kassahun(✉), and Bedir Tekinerdogan

Information Technology Group, Wageningen University and Research, Hollandseweg 1, Wageningen, The Netherlands
emmanueahoa@gmail.com, {ayalew.kassahun,Bedir.tekinerdogan}@wur.nl

Abstract. Supply chains consist of a network of people, activities, resources and organizational systems that coordinate to move a product or services from one point to another. A typical supply chain network contains multiple different actors that have different needs and operate under different business conditions. The complex and lengthy structure of agri-food supply chains for instance makes it difficult for analysts to identify and model the appropriate business processes. Several process reference models have been provided for describing process models, but these tend to omit explicit guidance for configuring supply chain business processes particularly in the agri-food supply chains. Hence, this paper applies and adapt the SCOR model levels to demonstrate an approach for supporting configuration of supply chain business processes dedicated for supply chains. We illustrate the approach for an industrial case of a cocoa supply chain. The approach was also applied to define and model the level 4 processes which is out of scope of the SCOR model and scarce in literature for many sectors including our illustrated case.

Keywords: Supply chain network · Agri-food supply chains
Cocoa supply chain

1 Introduction

The concept of supply chains was born when organizations realized in order to concentrate on their core competencies, they need to source goods and services from companies other than those they own [1]. This led to the development of cross-organizational business processes. This, in turn, has brought about a close relationship between supply chain management and process management [2].

A typical supply chain network of today contains multiple actors who have different needs and operate under different business conditions. However, supply chains are interrelated and interdependent and can be viewed as a network of people, activities, resources and organizational systems that coordinate to move a product or services from one point to another [3]. From the system perspectives, [4] identify three categories of conceptualizations with which supply chains can be defined, namely:

(1) network of cooperating actors, (2) cross-organizational business processes performed by the actors, supply chain business processes and (3) the management of

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 338–351, 2018.
https://doi.org/10.1007/978-3-319-94214-8_25

the business processes performed by actors. To manage supply chain and enhance its practices in the concepts mentioned above, a number of different reference supply chain frameworks have been developed.

The Supply Chain Council (SCC) developed a standard model called the Supply Chain Operations Reference (SCOR) model that can be used to manage, benchmark and advance supply chain business processes and practices. The SCOR model aggregated supply chain management and practices into four levels: top/process level (level 1), configuration/process categories (level 2), process element level/decomposes processes (level 3) and implementation/decompose process element (level 4) [5]. The SCOR model as it stands now is the most established and widely accepted framework used by consultants and industry practitioners in supply chain [6]. However, level 4 is out of scope from the SCOR model since each individual sector will have a different procedure for defining and implementing their business processes [2, 7].

The SCOR model has been applied to the production [8] and construction [9] industries extensively. Despite its relevance in improving and benchmarking supply chain business processes, the application of the SCOR model in the agri-food supply chains is scarce in literature. There is insufficient guidelines from the SCOR model for deriving and modelling agri-food supply chain business processes.

To close and address this gap, this present paper applies and adapt the levels in the SCOR model to propose an approach for configuring supply chain business processes. We illustrate the approach for an industrial case of a cocoa supply chain.

The remainder of the paper is organized as follows: Sect. 2 highlights on the background information of the study. Sect. 3 presents the case and the problem statement. In Sect. 4, we present the proposed approach. The approach was evaluated and applied to an industrial case in Sect. 5. Sects. 6 and 7 focusses on related work and conclusion of the paper respectively.

2 Background

2.1 Supply Chain

Supply chains consists of a network of entities such as raw-material suppliers, manufacturers/producers, distributors and retailers that are interconnected by material, information and financial flows [10]. [1] defined supply chain as the flow of materials, information, money and services from the supplier through a series of a network to the final consumer. Material flow represent the physical products, raw materials, and supplies that flow through a supply chain network. Information flows on the other hand consist of data associated with the various business activities in the chain whilst financial flows involve the monetary transactions such as payments, money transfer or exchange, credit card information and electronic payments.

In order to manage the business processes and performance of these flows, the Supply Chain Council (SCC) developed a standard model called the Supply Chain Operations Reference (SCOR) model. The model further aggregated the supply chain management processes into four levels. The first level defines supply chain management scope and content. This level contains five main management process perspectives: plan (planning), source (procurement), make (production), deliver (delivery), and return (post-delivery support).

The second level defined the process categories of the main process highlighted in level one. This level contains 26 supply chain processes. The choice of this level depends on the production strategy: *Make-to-stock* (MTS), *Make-to-Order* (MTO), *Engineer-to-order* (ETO). The MTS involves the production strategy linked to sales forecasts. The MTO on the other hand concern the production strategy associated with customers' orders. ETO processes is where a particular product is designed specifically to meet customer demand [11, 12].

The third level detailed the activities per process category. It contains 185 steps that are performed to execute the level 2 processes. According to the SCC, level four is out of scope because it concerns the unique focus and characteristics of each sector.

2.2 Business Process Modelling (BPM)

BPM is an accepted method used in many business organizations for structuring business sectors. It is used to provide support to organization processes with the help of different techniques, methods and software tools to control and analyze organizational process as well as activities [13]. [14] defined BPM as an activity of representing elements to produce a model that are used to deliver product or service to a customer or another organization. It has also been seen as the first step in business process modelling and business process management [15]. According to [16], an enterprise can correctly be analyzed and integrated only when the business processes of that enterprise is correctly identified and modelled. They further elaborated that information systems implementation can only be successful only when the business processes of the organization are well understood. This implies that proper identification and modelling of the business processes serve as a key factor in integrating business enterprise.

There are many modelling languages or techniques for business process modelling. The most frequent modelling techniques used include flow chart, data flow diagrams, Business Process Modelling Notation (BPMN), Event- driven Process Chain (EPC) and activity diagram which formed part of Unified Modelling Language (UML). The researchers used the Business Process Modelling Notation (BPMN version 2.0) to present the supply chain business processes presented in this study. The BPMN is the most widely used and prominent visual modelling method to model business processes [17, 18]. BPMN provides notations that is comprehensible by all stakeholders including business owners, process analysts and IT engineers. This helps to bridge the communication gap between these stakeholders.

3 Case Study and Problem Statement

In this section we present the industrial case study that will be used to describe the problem statement and illustrate the proposed approach.

3.1 Case: Cocoa Supply Chain

Ghana forms part of the West African countries bordered by neighboring African countries; to the north is Burkina Faso, the east is Togo, and the west is Cote d' Ivoire. Agriculture and forestry contribute to the bulk of the nation's economy [19].

Cocoa which is the main crop grown in the country accounts for 30% of the total export earnings in the country [20]. In addition, the Ghana cocoa supply chain provides income for six million people, representing 30% of the population in Ghana [21]. Currently, the Ghana cocoa sector is the second largest producer and exporter of cocoa beans in the world and accounts for 20% of the global cocoa production [20].

The cocoa supply chain of Ghana is very complex with a lot of actors with diversified interests, cultural and educational backgrounds [22]. The actors in the supply chain execute the day-to-day activities in the sector. The supply chain has a partially liberalized marketing structure with some privatization elements and strong government presence [23]. For instance, the price of cocoa is regulated under the producer price review committee established by the government of Ghana.

3.2 Problem Statement

In the previous section, the supply chain of the case was described. The description indicates that the cocoa supply chain contains network of entities that oversee the general business processes in the supply chain. These multitude of entities make the structure complex and lengthy and as a result makes it difficult for the actors in the chain to collaborate and coordinate their business activities. It also does not make the supply chain responsive and makes it difficult for the players in the supply chain to plan strategically and share information among themselves. Deriving an accepted business process models and workflows for such a complex supply chain structure tends to be difficult.

Though the SCOR-model provides general guidelines for supply chain management and practices, there is insufficient guidelines for deriving and modelling specific business processes for supply chains such as our running case. It is thus trivial to identify and model the unique processes of the actors in our cocoa supply chain case. We need a systematic approach and guidelines to identify and model business processes for such cases. In the light of this, this paper adapts the levels in the SCOR model to demonstrate an approach for configuring supply chain business processes for the case of cocoa supply chain.

4 Approach

In the following we will present the steps of our approach for deriving business process models dedicated for supply chains. For this we will apply and adapt the SCOR model levels. The overall approach we followed consists of five sequential steps (see Fig. 1). The steps:

1. *Define the goal:*
   In this step, the goal for deriving the supply chain business processes must be defined. From the supply chain perspective, the goal can stem from supply chain business process redesign, product quality, supply chain business process improvement/automation and monitoring.
2. *Identify the stakeholders:*
   This step involves identifying and mapping the stages in the supply chain. A variety of stages such as supplier, manufacturer/producer, distributor, retailer, customer etc. could be identified and mapped at this step. A stakeholder analysis could be used to identify and map the entities at the different stages in the supply chain. In case of unfamiliar supply chain organizations, first-hand information could also be derived from secondary sources.
3. *Identify the supply chain business processes:*
   This step constitutes the main body of the supply chain business process identification phase. It identifies the information, products, and financial business processes perform at each stage of the supply chain. Stakeholder analysis using surveys can be used to identify supply chain business processes at this stage.
4. *Map the business processes to SCOR level 1 to level 3:*
   This step involves mapping the supply chain business processes identified in step 3 to the SCOR model levels (level 1–3).
5. *Define the business processes for SCOR level 4:*
   In contrast to SCOR model, this level is out of scope because the business processes at this stage need to be an industry specific. For the case of this approach, this step involves defining the industry specific business activities for preforming the level 3 processes. The activities detailed in this stage serve as the fundamental building blocks for the business process implementation.

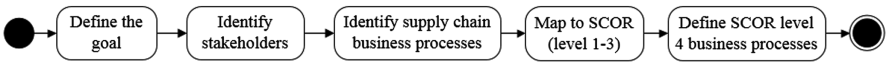


Fig. 1. Approach for deriving supply chain business processes

5 Applying the Approach: Business Processes for the Cocoa Supply Chain

Now that we have adapted and applied the SCOR model to propose a framework for configuring supply chain business processes, it is thus important to evaluate the framework with an industrial case. In this section we illustrated the framework with an industrial case of a cocoa supply chain.

We performed a data collection in the Ghana cocoa supply chain. The data were collected using semi-structured surveys which include face-to-face interviews, office and field visits. Purposive sampling technique was used to select 28 stakeholders in the cocoa supply chain. The stakeholders selected for the study oversee the daily operations in the cocoa supply chain. The field data were digitized onto Microsoft Office

Excel 2016. Secondary sources of data such as industrial reports and published articles were also used to support the research. Both the primary sources of data (field data) and the secondary data were used to identify the stakeholders in the Ghana cocoa supply chain and their business processes.

In this section, the results for applying the proposed approach to the case is presented. The results are presented systematically in accordance with the proposed approach.

5.1 Define the Goal

The current Ghana cocoa supply chain is primarily implicitly defined. This seriously impedes the understanding of the overall process. Likewise, it is harder to control and manage the overall supply chain. In particular, this is necessary for realizing fair and inclusiveness business for all the stakeholders in the supply chain. To this end, the goal of this business process modelling effort is to make the implicit process explicit and understandable in order to pave the way for improvement. In the SCOR model, level 1 describes the high-level configuration; level 2 is used to differentiate the strategies; level 3 describes the steps for executing level 2 processes. As stated before SCOR does not include level 4 which is focused on industry specific aspects. We focus in particular on applying the proposed approach to define and map the cocoa supply chain processes and elaborate on level 4 to describe the specific aspects of the Ghana cocoa supply chain.

5.2 Identify the Stakeholders

The output from the semi-structured surveys reveals three major stakeholders in the Ghana cocoa supply chain: (1) cocoa farmers, (2) cocoa buying companies formerly known as Licensed Buying Companies (LBCs) and (3) COCOBOD. The results confirms with a study conducted by [24].

5.3 Identify the Supply Chain Business Processes

The supply chain business processes performed by the identified stakeholders in the cocoa supply chain have been elaborated in the table below.

Table 1. Supply chain business processes of stakeholders in Ghana cocoa supply chain

Stakeholders	Supply chain business processes	ID
Farmer	• Schedule cocoa production activities	FP1
	• Source inputs (land, cocoa seedlings, fertilizer, insecticides, etc.) for the cocoa production	FP2
	• Grow the cocoa seedlings	FP3
	• Farm maintenance (weeding, pruning, pest and disease control, mulching etc.)	FP4

(*continued*)

Table 1. (*continued*)

Stakeholders	Supply chain business processes	ID
	• Harvest cocoa pods	FP5
	• Post-harvest activities (pod breaking, fermentation, drying etc.)	FP6
	• Bag the dried cocoa beans	FP7
	• Load mini transport vehicles	FP8
	• Transport the dried cocoa beans to buying company	FP9
	• Receive defective cocoa beans where necessary	FP10
	• Receive cash payment	FP11
Licensed Buying Company (LBC)	• Establish cocoa sourcing plans	BCP1
	• Receive dried cocoa beans from farmer	BCP2
	• Check cocoa beans quality	BCP3
	• Weigh cocoa beans if quality is okay or cocoa is well dried. Reject for reconditioning if cocoa beans fail the quality checks	BCP4
	• Transfer cocoa beans to warehouse	BCP5
	• Pay the cocoa farmer	BCP6
	• Plan delivery of bagged cocoa beans to COCOBOD ports	BCP7
	• Load trucks and generate transporting certificates and supporting documents	BCP8
	• Transport bagged cocoa beans to COCOBOD ports	BCP9
	• Receive payment from COCOBOD accounts	BCP10
COCOBOD	• Establish cocoa sourcing plans	CP1
	• Receive cocoa beans from buying company	CP2
	• Check cocoa beans quality	CP3
	• Weigh cocoa beans if quality is okay or cocoa is well dried. Reject for reconditioning if cocoa beans fail the quality checks	CP4
	• Transfer cocoa beans to warehouse	CP5
	• Pay the buying companies	CP6
	• Establish plans (select carrier, call Ghana Customs and Bank of Ghana for advice) for delivery of cocoa beans to local and international customers	CP7
	• Load trucks and container containing cocoa beans	CP8
	• Generate shipping documents	CP9
	• Ship bagged cocoa beans to customers	CP10

Note: The abbreviations ID denote business process identification, FP represents farmer business, BCP signify Licensed Buying Company business process whilst CP denote COCOBOD business process.

5.4 Map the Business Processes to SCOR

In this section, the identified business processes of the stakeholders in the cocoa supply chain are mapped to the SCOR model levels (1–3) as highlighted in the proposed approach. The mapping of the different business processes according to the SCOR model levels are described below using the ID codes in Table 1.

SCOR Level 1 Business Processes of Ghana Cocoa Supply Chain
As indicated in this study, the SCOR model level 1 is aggregated into five management processes (plan, source, make, deliver and return) used to describe supply chains at high level. In this section, these management processes have been defined explicitly for the Ghana cocoa supply chain. The results indicate that the business processes performed by the stakeholders in the cocoa supply chain conform to all the five level 1 processes in the SCOR model.

SCOR Level 2 Business Processes of Ghana Cocoa Supply Chain
From the SCOR model, the level 2 has three important production strategies which include the Make-to-stock (MTS), Make-to-Order (MTO), Engineer-to-order (ETO). The results from the survey study indicates that the actors in the cocoa supply chain operate under the MTS strategy. This implies that the supply chain business processes of the actors are performed based on forecasting measures. Out of the 26 level 2 business processes, 15 are applicable to the cocoa farmers processes outlined in Table 1. The cocoa buying company and COCOBOD uses 14 level 2 processes (see Table 2). For instance, the level 2 processes of the cocoa farmer initiates with *plan supply chain* and ends with *enable return*. The *plan supply chain* process considers the overall planning of the supply chain activities performed by the cocoa farmer and this is also applicable to licensed buying company and COCOBOD. The level 2 business processes not applicable to the cocoa stakeholders supply chain business processes outlined in Table 1 are denoted with N/A.

Table 2. Level 2 processes of Ghana cocoa supply chain

		Cocoa supply chain stakeholder		
Processes	SCOR process (level 2)	Cocoa farmer	Buying company	COCOBOD
1	sPl: Plan Supply Chain	FP1	BCP1	CPI
2	sP2: Plan Source	FP2	BCP1	CPI
3	sP3: Plan Make	FP3	BCP9	CP10
4	sP4: Plan Deliver	FP9	BCP7	CP7
5	sP5: Plan Return	FP10	BCP4	CP4
6	sSl: Source Stocked Product	FP9	BCP2	CP2
7	sS2: Source Market-to-Order Product	N/A	N/A	N/A
8	sS3: Source Engineer-to-Order Product	N/A	N/A	N/A
9	sMl: Make-to-Stock	FP3	N/A	N/A
10	sM2: Make-to-Order	N/A	N/A	N/A

(*continued*)

Table 2. (*continued*)

		Cocoa supply chain stakeholder		
11	sM3: Engineer-to-Order	N/A	N/A	N/A
12	sDl: Deliver Stocked Product	FP9	BCP9	CP10
13	sD2: Deliver Make-to-Order Product	N/A	N/A	N/A
14	sD3: Deliver Engineer-to-Order Product	N/A	N/A	N/A
15	sD4: Deliver Retail Product	N/A	N/A	N/A
16	sSRl: Source Return Defective Product	FP10	BCP2	CP2
17	sSR2: Source Return MRO Product	N/A	N/A	N/A
18	sSR3: Source Return Excess Product	N/A	N/A	N/A
19	sDRl: Deliver Return Defective Product	FP9	BCP5	CP5
20	sDR2: Deliver Return MRO Product	N/A	N/A	N/A
21	sDR3: Deliver Return Excess Product	N/A	N/A	N/A
22	sEP: Enable Plan	FP1	BCP1	CPI
23	sES: Enable Source	FP2	BCP1	CPI
24	sEM: Enable Make	FP3	BCP9	CP10
25	sED: Enable Deliver	FP9	BCP7	CP7
26	sER: Enable Return	FP10	BCP4	CP4

SCOR Level 3 Business Processes of Ghana Cocoa Supply Chain

According to the SCC, the level 3 processes describe the sub-processes performed to execute the level 2 processes. As indicated in the previous section, the level 2 processes centered around the MTS production strategy. This means the level 3 process should contain all the sub-processes or steps for executing the level 2 processes (MTS) described in the previous section. For our running case, some of the sub-processes or steps are applicable and others too are not. The cocoa farmer uses six executing processes in the level 3; cocoa buying company and COCOBOD made use of eight sub-steps of the executing sub-processes in level 3 (see Table 3). N/A in Table 3 was used to represent SCOR level 3 processes not applicable to the supply chain business processes of stakeholders in the Ghana cocoa supply chain.

Table 3. Level 3 processes of Ghana cocoa supply chain

	Cocoa supply chain stakeholders		
SCOR process (level 3)	Cocoa farmer	Buying company	COCOBOD
sP2.4: Establish sourcing plans	N/A	BCPl	CPI
sP3.4: Establish production plans	FP2	N/A	N/A
sP4.4: Establish delivery plans	N/A	BCP7	N/A
sS1.2: Receive product	N/A	BCP2	CP2
sS1.3: Verify product	N/A	BCP3	CP3
sS1.4: Transfer product	N/A	BCP5	N/A
sS1.5: Authorize supplier payment	N/A	BCP6	CP6
sMl.l: Schedule production activities	FP1	N/A	N/A
sM1.3: Produce and test	FP3	N/A	N/A
sM1.4: Package	FP7	N/A	N/A
sD1.5: Build loads	FP8	N/A	CPS
sD1.7: Select carriers and rate shipments	N/A	N/A	CP7
sDl.11: Load vehicle and generate shipping document	N/A	BCP8	CP9
sD1.12: Ship products	FP9	BCP9	CP10

5.5 SCOR Level 4 Business Processes of Ghana Cocoa Supply Chain

This section presents the translation of the level 3 processes into an industry specific business processes. In the case of this study, the level 3 processes derived from the SCOR model are translated to align with the cocoa supply chain business processes. The business processes are presented visually using the BPMN. The supply chain business processes of stakeholders in the cocoa sector is divided into three lanes (cocoa farmer, licensed buying company and COCOBOD).

From the business process diagram, the activities in the cocoa supply chain are initiated by the cocoa farmer. The cocoa farmer activities start with *establish cocoa production plans*. The business process of the cocoa farmer comes to an end when the bagged cocoa beans are transported to the licensed buying company (see Fig. 2). The licensed buying company process is initiated with *establish sourcing plans*. The licensed buying company after receiving the cocoa beans from the farmer also performed a number of activities such as *verify product, transfer product* etc. before transporting the cocoa beans to COCOBOD (see Fig. 2). The domestic business processes of the cocoa supply chain ends when COCOBOD ship the bagged cocoa beans to local and international customers (see Fig. 2).

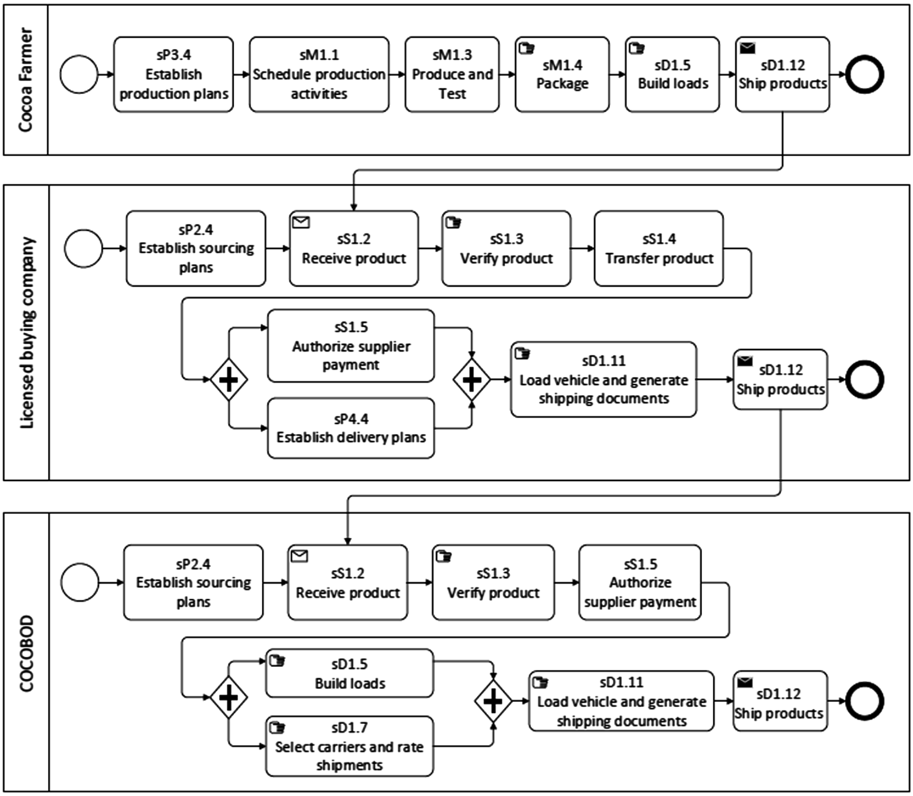


Fig. 2. Business process model of cocoa supply chain stakeholders

6 Related Work

Numerous studies have been conducted in the application of the SCOR model and also proposing frameworks for business processes. However, to the best of our knowledge, none of these studies have dealt into details in proposing systematic guidelines for deriving supply chain business processes.

[15] proposed a framework for business process management. This framework is widely used in the field of business process management for identifying and establishing criteria for prioritizing business processes in an organization. The framework uses two phases to identify business processes: designation and evaluation phase. However, the framework is generic and does not support deriving supply chain business processes. [25] also used process mining concepts for business process discovery and management. The approach uses log files with Process Identification Number (PID) from enterprise systems to create and redesign business processes. This processing mining approach which is widely known in the business process discovery discipline also fails to explicitly provide sufficient guidelines for supporting supply chain business process configuration.

[26] adapted the SCOR model to propose a framework for modelling demand driven supply chains. The proposed framework was applied to the European fruit industry. The SCOR model has also been applied in the construction industry by [27, 28] to model supply chain participants and their business processes.

These related works have some limitations; firstly, they do not focus on providing a step by step guidelines on how to configure the supply chain business processes. Again, these related works and SCOR model literature fails to provide approach for configuring the level 4 industry specific processes.

In the case of the Ghana cocoa supply chain, [24] employed supply chain perspectives to identify the challenges in the cocoa supply chain. The output of the research fails to explicitly indicate the detail supply chain business processes of the stakeholders in the cocoa supply chain.

This paper has addressed these concerns by applying and adapting the well-known SCOR model to propose an approach to configure supply chain business processes.

7 Conclusion

This study has applied and adapted the SCOR model to illustrate an approach that guides deriving supply chain business processes. Our approach consists of the following steps: (1) identify the goal; (2) identify the stakeholders; (3) identify the supply chain business processes; (4) map the business processes to SCOR level 1-3; and (5) defined the business processes for SCOR level 4. The research applied and tested the approach in a descriptive industrial case study. The results from the presented case identify the key actors from the Ghana cocoa supply, their business processes and interactions. The research also elaborated on the business processes on the SCOR model that are applicable to the Ghana cocoa supply chain.

This research which is the first of its kind to use the SCOR model to identify the processes in the cocoa supply chain contributes to a better understanding of the actors in the sector and their interactions. From a practical point of view, the research approach demonstrated in this study could be useful in deriving the supply chains processes in the agri-food sector particularly cash crop supply chains. This is because the research has broken boundary to identify the level 4 processes which is out of scope in the SCOR model for an agri-food supply chain specifically cocoa supply chain.

The output of this present paper has an added value to literature because the research adapts and apply an accepted supply chain reference model to propose an approach to configure supply chain business processes. It builds on existing reference model to illustrate new approach for supply chains business process configuration.

However, more research could be carried out to extend the findings of this study and validate the proposed approach. New research could also be carried out by using the SCOR model to identify the metrics and practices associated with the Ghana cocoa supply chain business processes.

References

1. Rainer Kelly, R., Prince, B., Watson, J.H.: Management Information Systems. Moving Business Forward, 3rd edn. Wiley, New York (2015)
2. Wattky, A., Neubert, G.: Improving supply chain performance through business process reengineering. In: Bernus, P., Fox, M., Goossenaerts, J.B.M. (eds.) Knowledge Sharing in the Integrated Enterprise. ITIFIP, vol. 183, pp. 337–349. Springer, Boston (2005). https://doi.org/10.1007/0-387-29766-9_28
3. Tekinerdogan, B., Sozer, H.: Architectural viewpoint for modelling dynamically configurable software systems. In: Managing Trade-Offs in Adaptable Software Architectures, pp. 79–97 (2017)
4. Lambert, D.M., Cooper, M.C.: Issues in supply chain management. Ind. Mark. Manage. **29**(1), 65–83 (2000)
5. Ling, L., Qin, S., Xu, C.: Ensuring supply chain quality performance through applying the SCOR model. Int. J. Prod. Res. **49**(1), 33–57 (2010). https://doi.org/10.1080/00207543.2010.508934
6. Lambert, D.M., Garcia-Dastugue, S.J., Croxton, K.L.: An evaluation of process-oriented supply chain management frameworks. J. Bus. Logist. **26**(1), 25–51 (2005)
7. Harmon, P.: Second generation business process methodologies. Bus. Process Trends Newsl. **1**(5), 1–12 (2003)
8. Garcia, F.A., Marchetta, M.G., Camargo, L., Morel, L.: A framework for measuring logistics performance in the wine industry. Int. J. Prod. Econ. **135**(1), 284–298 (2012)
9. Cheng, J.C.P., Law, K.H., Bjornsson, H., Jones, A., Sriram, R.D.: Modelling and monitoring of construction supply chains. Adv. Eng. Inform. **24**, 435–455 (2010)
10. Vijay, K.: Systematic assessment of SCOR for modelling supply chains. In: 38th Hawaii International Conference on Systems Sciences. Centre for Process Innovation, Georgia State University. 10 p. (2005)
11. Supply Chain Council, Inc.: SCOR Overview: Supply Chain Council, Version 10 (2010)
12. Lepori, E., Damand, D., Barth, B.: Benefits and limitations of the SCOR model in warehousing. Int. Fed. Autom. Control **7**(1), 424–429 (2013)
13. Weske, M., van der Aalst, W.M.P., Verbeek, H.M.W.: Advances in business process management. Data Knowl. Eng. **50**(1), 1–8 (2004)
14. Davenport, T.H., Short, J.E.: The new industrial engineering: information technology and business redesign. Sloan Manag. Rev. **31**(4), 11–27 (1990)
15. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.: Introduction to business process management. In: Dumas, M., La Rosa, M., Mendling, J., Reijers, H. (eds.) Fundamentals of Business Process Management, pp. 1–31. Springer, Heidelberg (2013)
16. Aguilar- Saven, R.S.: BP modelling: review and framework. Int. J. Prod. Econ. **90**(2), 129–149 (2004)
17. Object Management Group: Business Process Model and Notation 2.0 (BPMN 2.0) (2011)
18. ISO/IEC. Information Technology-Object Management Group Business Process Model and Notation. ISO/IEC 19510 (2013)
19. Asumadu-Sarkodie, S., Owusu, P.A.: The causal nexus between carbon dioxide emissions and agriculture ecosystem-an econometric approach. Environ. Sci. Pollut. Res. **24**(2), 1608–1618 (2017)
20. Monastyrnaya, E., Joerin, J., Dawoe, E., Six, J.: Assessing the Resilience of the Cocoa Value Chain in Ghana. Case study report (2016)
21. Anthonio, D.C., Aikins, E.D.: Reforming Ghana's cocoa sector: an evaluation of private participation in marketing. Master thesis. Lulea University of Technology Master (2009)

22. Collins, O.A., Aboagye, M.O., Justice, O.A., Margaret, O.B.: Information efficiency and the cocoa supply chain in Ghana. Am. Int. J. Soc. Sci. **4**(6) (2015)
23. World Bank: Supply chain risk assessment of cocoa in Ghana (2013)
24. Awuah-Gyawu, M., Brako, S., Adzimah, E.D.: Assessing the challenges facing cocoa production in Ghana. A case of selected licensed buying companies in Ashanti Region-Ghana. J. Supply Chain Manag. (2015)
25. Van der Aalst, W.M.P.: Process discovery. An introduction. In: Van der Aalst, W.M.P. (ed.) Process Mining. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-662-49851-4_6
26. Verdouw, C.N., Beulens, A.J.M., Trienekens, J.H., Van der Vorst, J.G.A.J.: Process modelling in demand-driven supply chains: a reference model for the fruit industry. Comput. Electron. Agric. **73**, 174–187 (2010)
27. Venkataraman, R.: Project supply chain management: optimizing value: the way we manage the total supply chain. In: Morris, P.W., Pinto, J.K. (eds.) The Wiley Guide to Project Technology, Supply Chain & Procurement Management, p. 242. Wiley, Hoboken (2007)
28. Cheng, J.C.: SC Collaborator: a service oriented framework for construction supply chain collaboration and monitoring. Ph.D. thesis, Department of Civil and Environmental Engineering, Stanford University, Standford CA, USA (2009)

Strategy-IT Alignment

Assuring Alignment Using a Relation Algebra Method

Frank Grave, Rogier van de Wetering, and Lloyd Rutledge(✉)

The Open University of the Netherlands,
Valkenburgerweg 177, 6419 AT Heerlen, The Netherlands
Lloyd.Rutledge@ou.nl

Abstract. The main purpose of this paper is to develop a model that assures the alignment between business and IT (BITA) for IT projects, based on the Ampersand method. BITA is essential in gaining value from IT investments to improve technical and human performance, to produce enhanced organizational strategies that yield competitive advantage and to perform better than businesses that do not align their business strategies with their IT strategies. The literature research proposes that Ampersand together with the Business Motivation Model (BMM) can assure BITA in two ways. First, it assures strategic fit between business strategy and business infrastructure and processes. Secondly, it assures functional integration between business infrastructure and processes, and IS infrastructure and processes. The BMM identifies ends (vision, goals, and objectives) and means to achieve the ends (missions, strategies, tactics, and business rules) of an organization. These concepts are relevant to an IT project, especially during the requirements engineering (RE) phase. For an organization to be effective and efficient, the ends and means have to be related to each other in some way. The proposed model records the relations between the ends and means and additionally checks the integrity of these relations using multiplicity constraints and business rules by using the Ampersand method.

Keywords: Business/IT-alignment (BITA)
Strategic Alignment Model (SAM) · Business Motivation Model (BMM)
Ampersand

1 Introduction

Business/IT-alignment (BITA) was first documented in the late 1970's and has been a top IT-management issue ever since [1]. This insight comes as no surprise as BITA realizes value from IT investments, ties the business and IT plan and drives competitive performance [2, 3]. Moreover, Lee and Kim [4] argue that BITA—resulting from specific socio-technical arrangements in organizations' infrastructure—is positively associated with business performance.

According to Luftman [5], BITA refers to applying information technology (IT) in an appropriate and timely way, in harmony with business strategies, goals, and needs. He states that BITA focuses on the activities that management performs to achieve cohesive goals across the IT and other functional organizations (e.g., finance, marketing,

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 352–361, 2018.
https://doi.org/10.1007/978-3-319-94214-8_26

human resources, manufacturing). Additionally, Luftman and Kempaiah [1] state that "*alignment must focus on how IT and the business are aligned with each other; IT can both enable and drive business change*."

Henderson and Venkatraman [3] argue that alignment is the degree of fit (alternative labels for alignment) and integration among the domains of the business strategy, IT strategy, business infrastructure and IT infrastructure. The first two domains focus on the external environment, whereas the last two focus on the internal organization.

Recent studies argue that BITA is optimal when harmony and balance (or equilibrium) exist between organizational and system goals and dimensions [6, 7].

Within the current scope, we use the definition for BITA from Ullah and Lai [8]: BITA is the degree of fit between business and IT activities such as business strategy, business infrastructure, IT strategy and IT infrastructure, etc. This definition is consistent with the Strategic Alignment Model (SAM) [3], which is a widely recognized model for describing different perspectives of alignment. However, current alignment conceptualizations and models are abstract, hard to operationalize and therefore challenging to apply in practice. Another alignment problem is the complexity of organizations [9], which impacts decisions on IS development and alignment.

Given the above, this paper proposes the use of the Ampersand method [10]—a method that applies relation algebra as a requirements language to functional specifications—as a means to achieve and assure BITA. The Ampersand approach makes requirements within the requirements engineering phase of the software development process explicit and supports stakeholders to agree upon these requirements. Consequently, Ampersand employs business rules to formulate a solid foundation for information systems (IS) design and, ultimately, the design that meets business expectations.

Hence, we define the following research questions that drive this current paper:

1. *How can the Ampersand method support the assurance of consistent and complete alignment of business and IT?* And
2. *Can we derive a model for BITA assurance based on the Ampersand method?*

This paper is structured as follow. Section Two introduced the theoretical background on the Ampersand method, the Business Motivation Model, and the Strategic Alignment Model. The evaluation and demonstration of our artifact follow these sections. We end this paper with a discussion of the results, conclusions and some limitations of our current work.

2 Theoretical Background

2.1 Ampersand: A Relation Algebra Method

Ampersand is a simple requirements specification language with relational semantics [10]. It is a relatively simple version of relation algebra. Scholars developed Ampersand for students and practitioners with a minimal mathematical background, who use it for designing business processes. The Ampersand approach employs business rules to formulate a sound basis for subsequent IS design and to define the business processes. Within this purely declarative syntax, actions are not specified but derived.

Ampersand features rules, relations, and concepts [10]. In essence, a specification in Ampersand is a set of rules and a set of relation symbol declarations with a concept-based type. Each rule is an expression (a relation term) in relation algebra that must be kept 'true' throughout time. Thus, each relation that is a rule represents an invariant requirement of the business. Ampersand populates the concepts in the business rules with atoms. An atom refers to an individual object in the real world. Another important aspect of the Ampersand method is the need for a language that is shared by all stakeholders [11]. Part of this method is to create a shared understanding of particular sentences that are input for the Ampersand application.

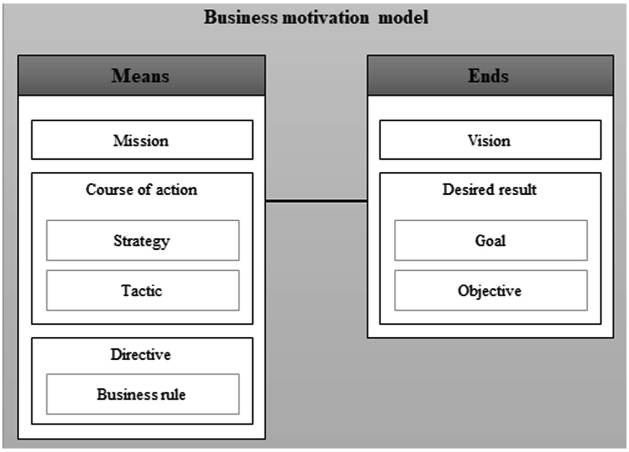


Fig. 1. BMM adapted from OMG [18]

2.2 Business Motivation Model (BMM)

The SAM offers strategies for achieving BITA. The SAM implies that organizations need to develop and align the business and IT strategy, as well as the organizational infrastructure, business processes, and the IS infrastructure. However, organizations strive to achieve a particular vision. This vision can be decomposed into clear and specific goals, creating a cohesive framework of interrelated goals [12]. Refined goals support organizations to harmonize its mission(s) and vision across the different levels of the organization. Organizations typically have a business and IT strategy (and plan) to achieve these goals.

Within this particular context, we use the Business Motivation Model (BMM); a joint effort of the Object Management Group and the Business Rules Group, shows in Fig. 1. The BMM is as a scheme or structure for developing, communicating, and managing business plans in an organized manner. This model supports the process of breaking down the business vision and mission into less abstract objects and eventually connecting the vision and mission to business rules and processes [13].

We argue that this particular model can be used together with Ampersand to assure BITA in practice. Hence, (i) the BMM contains the concepts and relations for the strategic fit of the business, and (ii) the Ampersand method helps formulate and check BITA using the business rules that are part of the BMM. Moreover, Ampersand can verify the business rules for consistency (not containing any logical contradictions) and completeness (having all necessary or appropriate elements).

The BMM consists of 'ends' and 'means', among the ends, are things the enterprise wishes to achieve (e.g., goals, and objectives). Among the means are things the enterprise employs to achieve those ends (e.g., strategies, tactics, business policies, and business rules).

SAM consists of four components, or quadrants. The business quadrants of the SAM are (1) business strategy and (2) the organizational infrastructure and processes. BITA aims to align the business quadrants of the SAM with its IT quadrants, i.e., (3) IT strategy and (4) IT infrastructure and processes. By keeping ownership of the requirements responsibility of the business and linking these requirements explicitly to business goals, we can assure the link with the business strategy. Organizations can synchronize the organizational infrastructure with the business strategy by deriving the process entirely from the requirements. Hence, alignment between information systems and the business can now is achieved through the generation of software straight from the requirements. IT representatives are always one of the stakeholders within the requirements engineering process.

The process of assessing BITA works as follows. A stakeholder within an organization has a 'purpose.' That purpose can be an executive-level business purpose, e.g., achieving a vision; it could also be a formal business rule from a project or anything in between. A purpose is motivated. Hence, its meaning is obtained by relating it to other atoms from different concepts within the same conceptual model.

2.3 Overall Method and Research Design

Research rigor is the driving goal for method selection [14]. The method helps in producing and presenting high-quality research. The goal of the research is to iteratively design a model that can assure the alignment between business and IT. Hence, we develop and evaluate an artifact that solves the identified organizational problem of misalignment between business and IT. In doing so, we follow the Design Science Research Methodology (DSRM) model by Peffers, Tuunanen [15]. Through the DSRM process iterations (from problem motivation to design and knowledge dissemination) a robust technology-based solution is obtained.

To ensure the quality and validity our artifact, we followed foundational guidelines for useful design science in information systems research [16]. These include, e.g., (1) the production of a viable artifact in the form of a model, (2) use of technology-based solutions for a business problem, (3) design evaluation and (4) research rigor.

3 Artifact Description

We created an artifact to demonstrate the feasibility of the designed product. The 'vision' concept of the BMM is part of the business strategy quadrant of the SAM. To achieve the declaration of objectives and collective goals, an organization (both public and private) employs core business activities and processes. These particular aspects are part of the organizational infrastructure and processes quadrant of the SAM. Business rules control the processes. Ampersand generates the design based on business rules, through which software can be generated [17]. The business rules are the basis for IS processes and infrastructure and also a quadrant of Henderson and Venkatraman's SAM. Following the BMM, and its relationships we now instantiate this, as Fig. 2 shows.

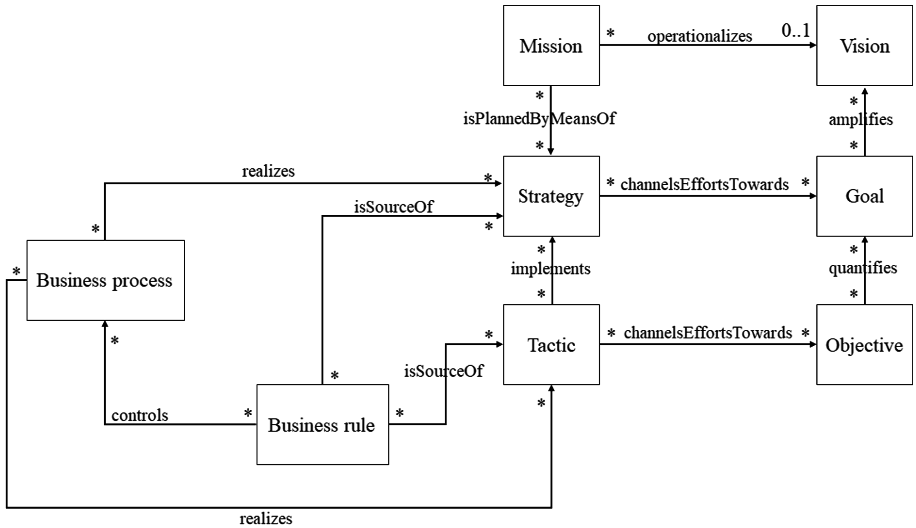


Fig. 2. Instantiated BITA alignment assurance model

Following the formal Ampersand requirements, this instance still needs the invariant business rules to work. Hence, we formulate six fine-grained (invariant) business rules—based on cycles, or 'closed loop,' that check whether we have all the relevant rules—that relate the eight concepts and their relations from the assurance. The six rules collectively determine each business rule atom contribute to the vision atom.

We defined the following invariant business rules:

i. For each strategy that is a means to plan a mission that operationalizes the vision, that strategy must channel efforts towards goals that amplify the vision. The formal notation in relation algebra is: *isPlannedByMeansOf∼; operationalizes* ⊢ *channelsEffortsTowards1; amplifies.*
ii. Each tactic that implements a strategy that channels efforts toward a goal, must channel efforts toward objectives that quantify that goal. The formal notation in relation algebra is: *implements; channelsEffortsTowards1* ⊢ *channelsEffortsTowards2; quantifies.*
iii. For each business process that realizes a strategy, there is a business rule—that is the source for that strategy—that controls that business process. The formal notation in relation algebra is: *controls; realizes1* ⊢ *isSourceOf1.*

iv. For each business process that realizes a tactic, which implements a strategy, that business process also realizes that strategy. The formal notation is: *realizes2; implements* ⊢ *realizes1*.
v. For each business process that realizes a tactic, that business process is controlled by a business rule that is the source for that tactic. Its formal notation is: *controls; realizes2* ⊢ *isSourceOf2*.
vi. For each business rule that is the source of a tactic that implements a strategy, that business rule is also the source of that strategy. Hence, the formal notation in relation algebra is: *isSourceOf2; implements* ⊢ *isSourceOf1*.

4 Evaluation of the Artifact

4.1 Project Criteria

We demonstrate and evaluate the model at the Dutch Tax and Customs Administration (DT&CA). The purpose of the demonstration is to unfold how an IT project that develops a new process and accompanying IS fits within the vision of the DT&CA. In doing so, we identified and selected projects that are managed and governed by the business unit Central Administrative Processes (CAP). To achieve our primary aim, we choose this single business unit, so that the independent project concepts are kept stable. Hence, we now can assess the contribution of projects along the lines of the project business rules, via business unit goals, toward the vision of the organization. We did seek for maximum variation by selecting projects with differentiation based on:

1. The delegated business owner: the responsible delegated business owner differs per project.
2. IT components: the projects make use of IT systems that are entirely separately managed and developed from each other.
3. Department: the projects are for three different departments.
4. Functionality: each project realizes a different kind of functionality.

4.2 Atom Formulation and Model Population

To populate our model, we need to derive atoms for vision, mission, strategy, goal, tactic, and objective from available documentation. Based on relevant project and policy documents within DT&CA and the definitions of the concepts in the model, we generate the required population for these concepts.

An essential part of the Ampersand method is the fact that business processes are designed based on requirements. We assume that each requirement translates to one business rule, as suggested by Joosten [17]. Also, we assume that user stories mention business rule atoms. Hence, atoms were deduced from these user stories thereby using RuleSpeak®[1] as a way of formulating the business rule atoms.

[1] RuleSpeak® is a set of guidelines for expressing business rules in concise, business-friendly fashion as much as possible.

4.3 Stage-Based Model Validation

We collected project data through the following steps. In the first step, we collected all available user stories from three projects, i.e., (1) Collecting on declarations (COD), (2) Payment Factory (PFA), and (3) Mini-One-Stop-Shop-Member State of Identification (MOSS-MSID). COD realize a process and IS that supports the collection of motor vehicle taxes. PFA assures that 99.11% of all payment processes are automated. MOSS-MSID develop the process and ICT for the Dutch Tax and Customs Administration to be able to comply with the European laws and regulations concerning the MOSS with the Dutch Tax and Customs Administration in the role of MSID.

Scaled Agile Inc. [18] argues that 'user stories' are short descriptions of a small piece of desired functionality, written in the user's language. User stories are therefore the primary artifact used to define system behavior. Hence, these stories provide just enough information for our research. In the second step, the user stories were used to extract business rules and business processes. Next, we validated these outcomes with the involved business architects and business analysts. We interviewed three subject-matter experts with in-depth knowledge of the IT projects from DT&CA. We interviewed these subject-matter experts independently using a semi-structured interview guide. The first expert is a senior business consultant with broad and extensive experience in enterprise-wide IT implementations. Also, this expert has much valuable knowledge on the particular challenges associated with IT project within DT&CA. The second expert is a business and change consultant. We selected this consultant based on his experience in shaping and managing organizational change. The third expert, a mid-level manager, was selected to determine the understandability of the model for the (senior) management. We then incrementally processed the review comments from these validation sessions into the business rules and business processes.

5 Model Demonstration

For each project, we loaded an Ampersand script (using a.txt file) into the Repository for Ampersand Projects (RAP) to complete the script. RAP is a cloud-based solution that stores Ampersand-scripts that users can specify, analyze and ultimately use to build information systems[2]. Once all functional requirements in RAP are free of errors, the consistency of these requirements is a mathematically guaranteed property. A script within the RAP environment that passes these (consistency) tests on syntax and typing is called 'accepted.' Full details concerning Ampersand and RAP are somewhat too technical, and beyond the scope of the current paper.

An Ampersand script contains all atoms of all concepts from the model for a project. The repository checks for consistency between atoms in the defined relations and specified business rules. After loading the script, RAP checks for rule violations, with dedicated functionality to report these violations. Table 1 presents the RAP outcomes per project. Table 2 shows the number of violations per business rule for each project.

[2] https://github.com/AmpersandTarski/RAP.

Table 1. Overview per project

	PFA	COD	MOSS-MSID	Total
Number of user stories	33	63	129	225
Number of business rules	23	36	24	83
Number of processes	7	39	28	74
Number of rule violations	181	107	87	375

Table 2. Number of violations per business rule per project

	PFA	COD	MOSS-MSID	Total
Business Rule 1	0	0	0	0
Business Rule 2	4	0	0	12
Business Rule 3	16	23	12	51
Business Rule 4	8	20	30	58
Business Rule 5	134	41	27	202
Business Rule 6	19	23	18	60
Total number of violations	**181**	**107**	**87**	**375**

In summary, Table 2 shows a total of 375 inconsistencies based on the assessment in RAP. While we see minor errors and inconsistencies, in Business Rules 1 and 2, we see many inconsistencies in Business Rule 5 for each project, specifically for PFA. Rule 5 specifies that for each business process that realizes a tactic (i.e., a course of action that is a device or that it is expedient to employ as part of a strategy.), that a business process is controlled by a business rule that is the source for that tactic. It seems that assuring this rule in practice is an unprecedented challenge. Controlling the consistency that is checked by this rule for example, is not a known process for the DT&CA. These violations should be used by consultants and analysts in practice to trigger actions. This principle follows Shewhart's Plan-Do-Check-Act cycle [19]. Analysis of the inconsistency should point out if the process actually does realize the tactic and if the business rule really is not the source of that tactic. The business rule to business process relationship is less likely to be incorrect. A stakeholder meeting can help to clear up this inconsistency. These results demonstrate how to maintain and assure alignment.

6 Discussion, Conclusions, and Limitations

Motivated by what in theory and practice appears to be a complicated process, this research developed a model to assess the alignment between business and IT following the Ampersand method. Using the SAM as described in [3] as a reference for measuring BITA, the model assures alignment between three of four quadrants of the SAM. The method checks alignment between business strategy and business organizational infrastructure and processes, alignment between business organizational infrastructure and processes and IS infrastructure and processes. Alignment with the IT strategy quadrant is currently not in scope within the proposed alignment method.

Our evaluation and demonstration show that BITA can be assessed and assured based on signaled violations requiring action. Violations of business rules and multiplicities of relations from the conceptual model form signals for an architect or analyst that the project design is inconsistent or incomplete. Based on these 'triggers' identified inconsistencies can be investigated and, if needed, restored to assure alignment.

From a managerial point of view, we apply the Ampersand method in assuring BITA within the organization. It is imperative—from both a theoretical and practical perspective—that business requirement should contribute to the firm's vision. Our proposed artifact explicitly uncovers often overlooked relationships by capturing requirements through business rule atoms and relating the business rule atoms to the vision atoms. The complimentary, holistic view of the incorporated BMM and the (in)consistency checks made by the RAP assure consistency in the use of requirements.

The current study uses only three validation cases. This amount of cases might inhibit the generalizability of our results. However, our restricting the scope enabled us to get an in-depth view of these projects and their contribution toward alignment. Second, many organizations do not deploy and implement projects following the BMM vision. BMM provides a generalized scheme or structure for the development, the process of communicating and managing business plans systematically. We, therefore, have the conviction that our artifact can be used situationally as a useful artifact and checklist to identify the alignment improvement areas systematically.

Acknowledgements. This work is based on the Master's Thesis of the first author, Grave [20].

References

1. Luftman, J., Kempaiah, R.: An update on business-IT alignment: "a line" has been drawn. MIS Q. Exec. **6**(3), 165–177 (2007)
2. Chan, E.Y., Reich, H.B.: IT alignment: what have we learned? J. Inf. Technol. **22**(4), 297–315 (2007)
3. Henderson, J.C., Venkatraman, N.: Strategic alignment: leveraging information technology for transforming organisations. IBM Syst. J. **32**(1), 4–16 (1993)
4. Lee, S.M., et al.: Developing a socio-technical framework for business-IT alignment. Ind. Manag. Data Syst. **108**(9), 1167–1181 (2008)
5. Luftman, J.: Assessing business-IT alignment maturity. Strat. Inf. Technol. Gov. **4**, 99 (2004)
6. Singh, S.N., Woo, C.: Investigating business-IT alignment through multi-disciplinary goal concepts. Requir. Eng. **14**(3), 177–207 (2009)
7. Wetering, R.: Modeling alignment as a higher order nomological framework. In: Abramowicz, W., Alt, R., Franczyk, B. (eds.) BIS 2016. LNBIP, vol. 263, pp. 111–122. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-52464-1_11
8. Ullah, A., Lai, R.: Modeling business goal for business it alignment. J. Comput. Inf. Syst. **51** (3), 21–28 (2011)
9. Bush, M., et al.: The alignment of information systems with organizational objectives and strategies in health care. Int. J. Med. Inform. **78**(7), 446–456 (2009)

10. Michels, G., Joosten, S., van der Woude, J., Joosten, S.: Ampersand. In: de Swart, H. (ed.) RAMICS 2011. LNCS, vol. 6663, pp. 280–293. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21070-9_21
11. Wedemeijer, L.: A relation-algebra language to specify declarative business rules. In: 4th International Symposium on Business Modeling and System Design, BMSD 2014 (2014). Revised Selected Papers: Shiskov, B. (ed.) BMSD 2014. LNBIP, vol. 220, pp. 63–73. Springer
12. Sandkuhl, K., Stirna, J., Persson, A., Wißotzki, M.: Enterprise Modeling. Tackling Business Challenges with the 4EM Method, vol. 309. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-43725-4
13. OMG: The Business Motivation Model Version 1.3 (2015)
14. Gregor, S., Hevner, A.R.: Positioning and presenting design science research for maximum impact. MIS Q. **37**(2), 337 (2013)
15. Peffers, K., et al.: A design science research methodology for information systems research. J. Manag. Inf. Syst. **24**(3), 45–77 (2007)
16. Hevner, A.R., et al.: Design science in information systems research. MIS Q. **28**(1), 75–105 (2004)
17. Joosten, S.: Deriving Functional Specifications from Business Requirements with Ampersand. Open University of the Netherlands (2007)
18. Scaled Agile Inc.: Story - Scaled Agile Framework (2017). http://www.scaledagileframework.com/story/. 9 Dec 2017
19. Shewhart, W.A., Deming, W.E.: Statistical Method from the Viewpoint of Quality Control. Courier Corporation, Chelmsford (1939)
20. Grave, F.: Assuring strategy and IT alignment: a relation algebra method using relation algebra to assure alignment between business and IT. Master's thesis, Open University in the Netherlands, March 2018

An Ontology-Based Expert System to Detect Service Level Agreement Violations

Alper Karamanlioglu(✉) and Ferda Nur Alpaslan

Department of Computer Engineering, Middle East Technical University, Universiteler Mh., 06800 Cankaya, Ankara, Turkey
{alperk, alpaslan}@ceng.metu.edu.tr

Abstract. In this paper, an expert system developed with an ontology-based approach to detect Service Level Agreement (SLA) violations is presented. The widespread use of SLAs in various areas complicates SLA management and in particular the detection of violations. Although it is necessary to automatically detect SLA violations, developing a different solution for each domain is quite costly. Several domains were investigated, and many common concepts have been identified in terms of SLAs. Nevertheless, it has been determined that each domain has its own distinct metrics and criteria. By combining familiar and changeable concepts, we have acquired the idea of creating a generic SLA ontology for different SLA domains. After generic SLA ontology was created, an expert system called SLAVIDES was developed using this ontology. The developed expert system is designed to detect SLA violations, check constraints, and make inferences. The developed system has been tested on the SLA data of the telecommunication domain. The results show that the proposed system can correctly detect SLA violations.

Keywords: Ontology-based system · Service Level Agreement
Quality of service

1 Introduction

A Service Level Agreement (SLA) can be defined as an agreement between two or more parties, one of which is a customer and the others are service providers. SLAs are contracts that involve separate organizations or different teams within an organization. Nowadays, SLAs have become frequently used agreements to increase accountability and quality as business volume increases in sectors where outsourcing is widespread. These agreements are most often employed in telecommunications, information technology, and healthcare sectors.

Ontologies are constructs that are used to gather information about specific fields of knowledge [1]. Concepts of knowledge domains, and the relations between these concepts can be defined using ontology. They provide analysis and reuse of domain knowledge. Therefore, it is possible to develop specialized systems in specific areas. Expert Systems are the type of specialized software systems that aim to solve the problems that human experts are creating better solutions. Ontologies and expert systems are often used together [2, 3].

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 362–371, 2018.
https://doi.org/10.1007/978-3-319-94214-8_27

The ability to manage SLAs as they become widespread in various sectors has become a major challenge. SLA violations should be detected shortly after the occurrence and the necessary sanctions should be applied quickly. Manual detection of the violation involves many problems. Some of these problems are the high error rate, the excess of reaction time, and loss of work power. Although manual detection has many challenges, automatic detection has its own difficulties. Because SLAs are used in many different domains, it is costly to develop separate software for each domain. For this reason, it is expected to develop a software that can be applied to various domains as much as possible. Database systems are not specific to the domain, so they will not meet this need.

In this study, an expert system called SLAVIDES is developed to detect violations of SLAs in various areas. The expert system has been established with an ontology-based approach. A generic SLA ontology is designed to provide a framework for creating SLAs. While creating this framework, concepts defined in SLA field and their relations are taken into consideration. Since the ontology is designed as generic, it is intended to be used in many different knowledge domains. To achieve this, we have defined separate classes for metrics and criteria. The main reason for doing this is that there are different metrics and criteria in each SLA domain. Thus, the system can adapt the metrics and criteria of the relevant domain.

SLA data is recorded as RDF triples so that semantic operations can be performed on it. In addition to SLA data, Shapes Constraint Language rules and constraints are also recorded in RDF format. These data, together with SLA ontology, are all recorded in Jena TDB [4]. Thus, constraint checking and rule inference can be performed. With the expert system developed by using the proposed ontology, it is desired to determine whether the violation of the SLA occurs. Two different SPARQL queries have been created to monitor and infer SLA violations. The violation monitoring result is sent back as a service message, and the violation inference result is stored in TDB. The developed expert system has been tested on telecommunication data.

1.1 Contributions

A generic SLA ontology is proposed, which can be used directly in many areas and expanded in some specialized areas. Thus, it is intended to increase reusability and flexibility. Storage of SLA data is provided in triple stores. In this way, many semantic operations can be performed on this data. An expert system has been developed to detect SLA violations using the proposed ontology. The system provides important contributions to identify violations accurately. Constraint checking and inferencing on SLA data is performed using new technologies. So, it is possible to comply with specified constraints and to infer hidden information. Generally, no actual SLA data was used in past studies. We have used real data as well as synthetic data.

2 Background

The definition of Service Level Agreement (SLA) is given as "the official commitment between a client and a service provider" [5]. Initially, SLAs were started to be used by the fixed line telecom operators in the late 1980s. In today's world, the usage of SLAs is so common that a company can have more than one SLA in itself, as the company is the service provider and the customers are as clients. In this way, SLAs can be very useful for a company to be able to offer the same quality of service among the different units. SLAs are also useful to assess the difference in a service that is provided by itself and a service gathered by an outside source [6]. The main focus of an SLA is the outputs received by customers based on the provided services.

One of the main purposes of using SLAs by enterprises is to offer a better Quality of Experience (QoE) to the clients domestically and externally [7]. QoE is a term that was defined to introduce some sort of measurement for the quality of a service or a product with respect to their performance, customer satisfaction, overall sales and delivery of these products or services. Thus, QoE allows an enterprise to balance the quality level of various products according to their costs and the expectations of the customers. Because of that, it is crucial to offer the necessary distinctions among the various products or services.

Since the quality of a product or service is standardized with QoE and the definition, measured objectives of a product or service are determined via SLA, it can be said that the QoE and the SLA are related. SLA should meet the quality measures determined by QoE. Mapping is required to match the measurements from the QoE to the objective measurements of the SLA. In order to achieve the expected quality level by the QoE, the Key Quality Indicators (KQI) must be included in the SLA. The KQI is achieved through defining, measuring and agreeing on some Key Performance Indicators (KPI). The metrics can be defined as KQI, which is a meaningful to the customer, or KPI, which is meaningful low-level metric type for the service provider [8].

The metrics used to measure and manage the performance characteristics of the service objects are important factors that make the agreement successful. These metrics, called SLA metrics, provide the ability to manage and measure performance compliance to SLA commitments. SLA metrics provide business continuity because of its contribution to customer satisfaction and confidence.

3 Related Work

The idea of creating generic ontologies has been addressed in several studies [9, 10]. While these ontologies can provide appropriate solutions for many areas, they can not adequately address specific areas such as SLAs. Apart from that, there are many studies do not propose a general solution enough or focuses on specific areas. Xie et al. have proposed an expert system [11] to detect SLA violations in virtual machines. This expert system has been developed ontology-based but it is not intended to find a suitable solution for all domains. The SLA ontology described by Di Modica et al. [12] is aimed to support service discovery in future cloud markets. Dobson et al. have proposed an ontology for Quality of Service called QoSOn [13] which is particularly focused on the field of service-centric systems.

In the study conducted by Green [14], ontology-based SLA formalization is defined. Charging, time unit, temporal, currency, network metrics, violation, entity, and SLS are defined as separate ontologies under the SLA ontology.

In [15], the ontology-based SLA Management (OSLAM) proposed by Seo et al. aimed to manage and guarantee SLAs for IPTV services using SWRLs together with ontologies. The authors analyzed many IPTV PIs from a variety of standard organizations. As a result of this analysis, they extended the DEN-ng model to suggest an IPTV PI hierarchy. The SWRL rules are used to detect SLA violations, to infer hidden relationships between an SLA and a PI, and to find PI value from other PIs.

Hamadache and Rizou offered an SLA ontology [16] covering the entire service lifecycle based on a QoS ontology representing the QoS model to ensure and improve the evaluation of the services. In the proposed ontology, SLA is the central concept and is directly linked to QoS requirements, Actor, Role, Service, and Feedback. With the ontology-based SLA management approach proposed in [17], it was intended to improve the SLA by taking into account the semantic meaning of SLA concepts and contextual information from the consumers of cloud services.

4 Generic SLA Ontology

While the previous studies in SLA field are investigated, it is seen that in almost every study the fundamental concepts are SLA, Service Level Specification (SLS), and Service Level Objective (SLO). SLS is a frame definition that contains all of the metrics, thresholds, and calculation formulas required for SLA metrics. SLO is a definition that contains metrics and their threshold values. It is revealed that there are other common concepts related to SLA, SLS, and SLO. Most of these concepts are not included in the constructed ontology. The reason for this is that most of these concepts are specific to the field. According to the definition of SLA, there must be a product and parties to the agreement. For this reason, “Product” and “Party” classes are added to the ontology. In addition to those, we have added Criteria and Metric classes to ensure the genericity of our ontology. We have identified that there are different metrics and criteria for each domain, so we are convinced that we need to create these classes. For example, although there are metrics such as “Order Completion Time KQI”, “Fault Completion Time KQI”, and “Service Availability” in the network area, these metrics do not have to be in every domain. In addition, concepts such as “region”, “job type”, and “type of access” that need to be defined in the network domain may not have to be in other areas.

Depending on the defined metric and criteria, there can be different threshold values. “Threshold” is defined as a separate class and this class determines the expected target value for metric measurements to reach. When determining this, it is critical to define whether the violation occurs below or above or equal to the threshold value. We achieve this with the help of the “Comparator” class we have defined. “MetricMeasure” is the class for which the measurement value is calculated based on the metric type. The calculated value is compared with the threshold value based on the comparator and it is determined whether there is a violation. Parties and SLSs have types and these types may sometimes be the same. Apart from that, criteria and metrics have also types. In case of an SLA violation, a penalty may be required [18]. Penalty class is created for a penalty to be determined in case of violation.

SLA ontology is created using Protégé tool [19] according to the classes, object properties, and data properties we have defined. The data properties shown here are just examples and may vary from field to field. SLA ontology was exported in OWL format. In Fig. 1, the constructed ontology is visualized with OntoGraf, a component of Protégé. Since there are no is-a relationships among the concepts, all the classes defined in the ontology are at the same level in the class hierarchy.

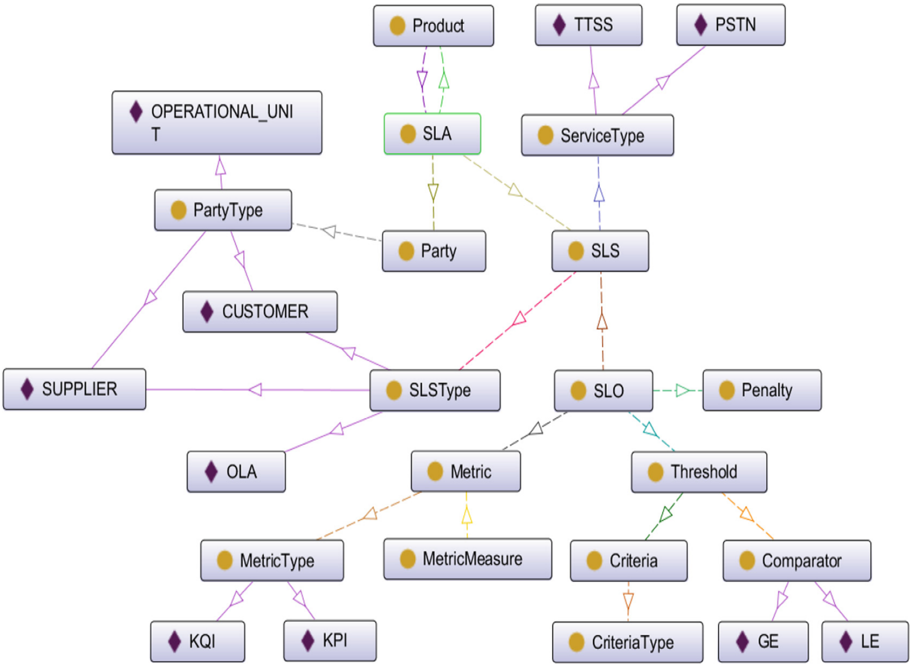


Fig. 1. The constructed SLA ontology.

5 SLA Violation Detection System

5.1 Storing SLA Values in Triple Stores

SLA values should be stored in triple stores as RDF triples so that semantic querying, inferencing or constraint checking can be performed. Most of the triple stores can usually be queried with SPARQL. With graph databases, data can be recorded as linked, but these databases do not have a standard query language. Thus, RDF triple store is preferred when recording data. TDB is a component of Apache Jena and is used for RDF storage and querying. It supports all of the Jena APIs. TDB can be used as a high-performance RDF storage in a single machine. The triple data employed in this study are stored in TDB. We recorded SLA values in TDB in three phases. The first phase is to record the metric-related data. Then SLS-related data is recorded. Finally, SLA-related data is included in TDB. Metric class has not related to the classes other than MetricType. For this reason adding, updating or removing metrics will not cause major problems. The most challenging task when recording SLA value is to store SLS–related data. All classes except Metric, SLA, Product and Party are considered as SLS-linked. The reason for this is that the SLS has all the specifications of the service part of SLA. Deleting or updating SLS-related data may cause several problems. For this reason, these operations should be carried out considering the classes to which they are connected. When adding SLA-related data, data related to Product and Party are added as well as SLA.

5.2 Architecture of the Developed Expert System

SLAVIDES processes the incoming service message and generates SPARQL queries for this message to monitor and infer the SLA violation. A new class named "Breached" has been added, which is not defined in the constructed ontology, to semantically store information related to the violation. One of the queries generated by SLAVIDES is to create a new relationship between SLA and Breached classes. The other is to monitor the SLA violation in the desired format. The generated queries are executed on SLA data which is stored as triples in Jena TDB. The violation monitoring query response is returned as a JSON message, and the violation inference query result is recorded in TDB. Besides, constraint checking and inferencing are performed according to SHACL [20] rules and constraints. These rules and constraints are also recorded in TDB. The data to be inferred is sent to the inference engine, and the result is also stored in TDB in RDF format. The system has components that perform the specified tasks. The system architecture in which these components are included is shown in Fig. 2.

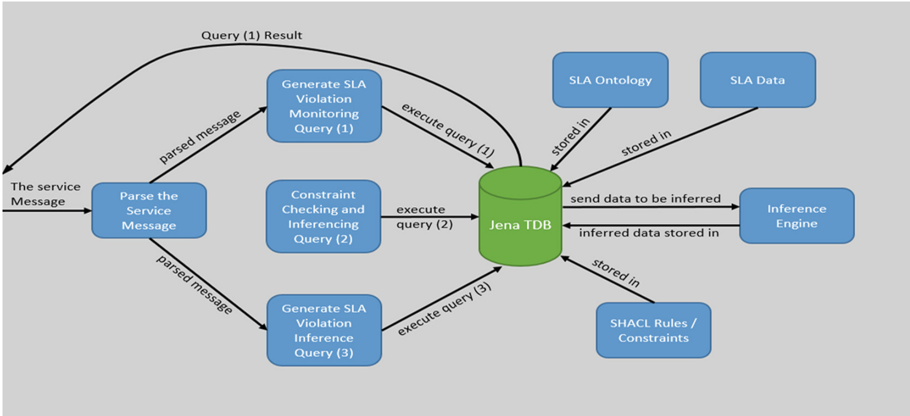


Fig. 2. The architecture of the developed expert system.

5.3 SLA Violation Monitoring

The information describing the product to be detected whether SLA violation has occurred is sent to SLAVIDES as service messages. In the content of these messages sent to the system, the incoming message ID, the product ID, the measurement values of the metrics and the information of the criteria are included. SLAVIDES obtains this information and ensures that a SPARQL query is generated to detect the violation. The system executes the generated query on the triple data which is stored in TDB. The query result in the form of JSON that occurs after executing this query is shown in Fig. 3. The query result is returned as a service message along with SLA violation and other required information. "+" value of the "Breached" class means that a violation has occurred. "Comparator" value "LE" means that the metric measurement value is compared with the "low and equal" comparator.


```
{
"head": {
"vars": [ "MID" , "KPI" , "SERIAL_NUMBER" , "FIRSTNAME" ,
"SURNAME" , "MEASURE" , "THRESHOLD" , "COMPARATOR" ,
"BREACHED" ]
} ,
"results": {
"bindings": [
{
"MID": { "type": "literal" ,
"datatype": "http://www.w3.org/2001/XMLSchema#integer" , "value": "1111" } ,
"KPI": { "type": "literal" , "value": "Order Completion Time KQI" } ,
"SERIAL_NUMBER": { "type": "literal" , "value": "2123122222" } ,
"FIRSTNAME": { "type": "literal" , "value": "Metin" } ,
"SURNAME": { "type": "literal" , "value": "Takak" } ,
"MEASURE": { "type": "literal" ,
"datatype": "http://www.w3.org/2001/XMLSchema#float" , "value": "3.0" } ,
"THRESHOLD": { "type": "literal" ,
"datatype": "http://www.w3.org/2001/XMLSchema#float" , "value": "5.0" } ,
"COMPARATOR": { "type": "literal" , "value": "LE" } ,
"BREACHED": { "type": "literal" , "value": "+" }
}]}}
```


Fig. 3. Returned SLA violation message.

Each service message may contain multiple metric measurements. Therefore, it is possible that more than one violation may occur for each message. SLAVIDES can process a single message as well as have the ability to process multiple messages. Initially, incoming service message parsed and the list of messages is identified. The results are then combined by performing separate queries for each message. In addition, many criteria can be defined within each message. The violation is detected by the query generated by considering these criteria and metrics. The measured values and threshold values are compared according to the comparator and it is determined whether or not the violation occurs.

The SPARQL query that monitors the SLA Violation is produced as follows:

- The prefixes to be used in the query are determined. Many of the classes defined in the SLA ontology are used when constructing queries. So, each of these classes must be defined as a prefix when starting to construct the query. Thus, it is possible to use abbreviated forms of URIs.
- "SELECT" command defined in SPARQL is used to monitor whether an SLA violation has occurred. Parameters to be sent back as service messages should be defined here.
- Multiple metrics are included in the query using the "VALUES" command. The number of metrics is not specific and can be any value.
- Criteria are produced dynamically according to the incoming message. The number or order of the criteria in the message may be different.
- By using the "OPTIONAL"/"FILTER" commands, it is possible to compare measured values by considering the comparison type.

5.4 Experiments of SLA Violation Detection System

The SLA Violation Detection System, developed using the proposed ontology, has been tested on 1036 actual triple data from SLAs in the field of telecommunication. Accordingly, the success of the system was measured by considering 75 different scenarios with different inputs. A group of three people determined that 62 of these scenarios did not constitute violations and 13 of them had one or more violations. It is stated that there is more than one violation in 3 of the violation cases. The developed system achieves the same results as the results of the group in all of the mentioned scenarios.

Other than that, we also wanted to measure the success of the system in a larger dataset, so we produced synthetic data. Synthetic data set consisting of 10000 triple data we produced was recorded in TDB. Again, 75 different service messages have been created. While creating these messages, we did not have the goal of keeping the percentage of actual violations occurring. The same group of people investigated whether these messages constituted violations. Nearly half of the messages have been constituted violations. More than one violation occurred in about half of the messages that have been constituted violations. Thereafter, the violation detection queries were executed on the synthetic data set. It has been determined that much of the violations found by the group of people and the violations that the system identifies are the same. Differences were again reviewed. It has been seen that the system correctly detected the violations, and the differences were caused by human error.

5.5 Constraint Checking and Rule Inference

Constraint checking and rule inference are frequently used features of ontology-based systems. We noticed that there are no common constraints and inference rules in all areas where SLAs are used. Every area has its own set of rules and constraints. Therefore, we have shown how sample constraints and inference rules can be defined. To achieve this, SHACL, a new technology approved by the W3C, is used. We employed SHACL's open-source and Jena-based implementation, TopBraid SHACL API, to perform constraint checking and rule inference. TopBraid SHACL API has been developed fully in accordance with SHACL specifications. Constraints and rules can be easily defined in TopBraid Composer. We then convert it to SHACL format via TopBraid SHACL API. Therefore, they can be stored as RDF triples in TDB. There are many different constraints and rules in different SLA areas. Frequent violations of SLAs in the same product may be due to incorrectly defined KQI values. For this reason, it is very important that these rules and constraints are correctly identified. Such examples may vary for each area and even for each company in the same area.

6 Conclusion and Future Work

In this study, an expert system developed to detect the violations of Service Level Agreement is introduced. This system, called SLAVIDES, has been developed ontology-based and therefore has the advantages of using ontologies. The most important of these benefits is to enable reuse of domain knowledge. SLAVIDES has also ability to perform semantic queries, infer new information from existing information, and check constraints. The developed expert system provides significant contributions in accurately detecting SLA violations. Furthermore, the flexibility of the proposed ontology leads to the possibility of direct use in many areas. In some areas, new ontologies can be created by extending SLA ontology when necessary.

Since the proposed ontology is generically designed, it can be used satisfactorily in many different areas, but there may be different needs and expectations in Service Level Agreements in some domains. Therefore, it will be useful to test the validity of the ontology and the developed SLA violation detection system by applying it to areas outside the telecommunications sector. Apart from these, another system is planned to be developed using machine learning techniques in the future to automatically predict SLA violations. The system to be developed will contribute to the prevention of SLA violations and improve the service quality. The system is planned to be developed as Java-based to facilitate integration into SLAVIDES.

Acknowledgements. The authors thank INNOVA IT Solutions Inc for all opportunities they have provided at their company. This work was partially supported by TUBITAK-TEYDEB Technology and Innovation Funding Programs (Project No: 3150860).

References

1. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing? Int. J. Hum. Comput. Stud. **43**(5–6), 907–928 (1995)
2. Lin, F.T., Lin, Y.C.: An ontology-based expert system for representing cultural meanings: an example of an Art Museum. In: Pacific Neighborhood Consortium Annual Conference and Joint Meetings (PNC), pp. 116–121. IEEE (2017)
3. Mekruksavanich, S.: Medical expert system based ontology for diabetes disease diagnosis. In: Software Engineering and Service Science (ICSESS), pp. 383–389. IEEE (2016)

4. Apache Jena - A free and open source Java framework for building semantic web and linked data applications. https//jena.apache.org. Accessed Dec 2017
5. Wieder, P., Butler, J.M., Theilmann, W., Yahyapour, R.: Service Level Agreements for Cloud Computing. Springer, Heidelberg (2011). https://doi.org/10.1007/978-1-4614-1614-2
6. Ding, J.: Advances in Network Management. CRC Press, Boca Raton (2016)
7. Bain, G., Dia, J.: SLA management handbook-volume 4: enterprise perspective. In: TM Forum (2004)
8. Guide ETSI: User Group; Quality of telecom services; Part 1: Methodology for identification of indicators relevant to the Users (2014)
9. Razmerita, L., Angehrn, A., Maedche, A.: Ontology-based user modeling for knowledge management systems. In: Brusilovsky, P., Corbett, A., de Rosis, F. (eds.) UM 2003. LNCS (LNAI), vol. 2702, pp. 213–217. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-44963-9_29
10. Sebastian, A., Noy, N.F., Tudorache, T., Musen, M.A.: A generic ontology for collaborative ontology-development workflows. In: Gangemi, A., Euzenat, J. (eds.) EKAW 2008. LNCS (LNAI), vol. 5268, pp. 318–328. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-87696-0_28
11. Xie, X., Wang, W., Qin, T.: Detection of service level agreement (SLA) violation in memory management in virtual machines. In: Computer Communication and Networks (ICCCN), pp. 1–8. IEEE (2015)
12. Di Modica, G., Petralia, G., Tomarchio, O.: An SLA ontology to support service discovery in future cloud markets. In: Advanced Information Networking and Applications Workshops (WAINA), pp. 1161–1166. IEEE (2013)
13. Dobson, G., Lock, R., Sommerville, I.: QoSOnt: a QoS ontology for service-centric systems. In: 31st EUROMICRO Conference on Software Engineering and Advanced Applications, pp. 80–87. IEEE (2005)
14. Green, L.: Service level agreements: an ontological approach. In: Proceedings of the 8th International Conference on Electronic Commerce: The New Ecommerce: Innovations for Conquering Current Barriers, Obstacles and Limitations to Conducting Successful Business on the Internet, pp. 185–194. ACM (2006)
15. Seo, S., Kwon, A., Kang, J.-M., Hong, J.W.-K.: OSLAM: towards ontology-based SLA management for IPTV services. In: IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 1228–1234. IEEE (2011)
16. Hamadache, K., Rizou, S.: Holistic SLA ontology for cloud service evaluation. In: International Conference on Advanced Cloud and Big Data (CBD), pp. 32–39. IEEE (2013)
17. Labidi, T., Mtibaa, A., Gargouri, F.: Ontology-based context-aware SLA management for cloud computing. In: Ait Ameur, Y., Bellatreche, L., Papadopoulos, George A. (eds.) MEDI 2014. LNCS, vol. 8748, pp. 193–208. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11587-0_19
18. Rana, O.F., Warnier, M., Quillinan, T.B., Brazier, F., Cojocarasu, D.: Managing violations in service level agreements. In: Rana, O.F., Warnier, M., Quillinan, T.B., Brazier, F., Cojocarasu, D. (eds.) Grid Middleware and Services, pp. 349–358. Springer, Heidelberg (2008). https://doi.org/10.1007/978-0-387-78446-5_23
19. Musen, M.A.: The protégé project: a look back and a look forward. AI Matters **1**(4), 4–12 (2015)
20. Knublauch, H., Ryman, A.: Shapes Constraint Language (SHACL). Working Draft (work in progress), W3C (2016)

Multi-sided Platforms for the Internet of Things

Thibault Degrande(✉), Frederic Vannieuwenborg, Sofie Verbrugge, and Didier Colle

IDLab, imec, Ghent University, iGent Tower, Tech Lane 15, 9052 Ghent, Belgium
thibault.degrande@ugent.be

Abstract. This paper combines the concept of multi-sided platforms and the emerging market of the Internet of Things (IoT). At present, the IoT market is in an early stage, with an abundance of fragmented solutions targeting specific domains and/or specific types of applications. In this paper, we defend our position that software back-end IoT platforms offer considerable potential to become leading multi-sided platforms as the core of the complex IoT ecosystem and hence, a race for domination of the IoT platform market will appoint these platform leaders. Furthermore, some important trade-offs are discussed for platform providing companies that aspire to become leading multi-sided platforms in the Internet of Things industry.

Keywords: Multi-sided platforms · IoT · Business models

1 Introduction

At the end of the previous century, companies that had taken leadership positions in their industries had become champions in either operational excellence, customer intimacy, or product leadership [1]. Since then, an extension of those three value disciplines has become imminent, since this categorization would not be able to fit in some of the most influential, largest and fast-growing companies of the past decade; the so-called ‘platform leaders’ [2]. Multi-sided platforms are “technologies, products or services that create value primarily by enabling direct interactions between two or more customer or participant groups” [3, p. 1]. The multi-sided platform (MSP) model has been gaining considerable attention, largely driven by the success of eBay and its Asian counterparts. Other prominent household name examples include Airbnb; enabling the lease or rent of short-term lodging of dwelling and Uber; enabling the interaction between drivers and passengers.

This positioning paper aims to initiate the research question if the multi-sided business model will also dominate an emerging and developing market like the IoT market. Could the biggest IoT company turn out to be a multi-sided platform, enabling the interaction between two or more distinct groups?

The remainder of this article is organized as follows: Sect. 2 presents a literature review on multi-sided platforms, followed by a brief overview of the IoT building blocks. In Sect. 3, multi-sided platform opportunities in the IoT value chain are discussed. Section 4 concludes.

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 372–381, 2018.
https://doi.org/10.1007/978-3-319-94214-8_28

2 Previous Literature

2.1 Multi-sided Platforms

The definition from [3] implies that a MSP is a physical or virtual place that enables participants to interact. Indeed, physical MSPs have been around for centuries. One can see that city square markets and shopping malls are early physical versions of two-sided markets, enabling the sale between individual vendors and customers and monetizing through ‘booth fees’. Recently, marketplace-firms have successfully leveraged the rise of the information technologies to become increasingly important agents in the digital economy [4].

Two key features make multi-sided platforms different from other forms of intermediary or regular firms, being: (1) affiliation of each side with the platform and (2) enabling direct interactions between two or more distinct sides [5]. By ‘affiliation’, the authors mean that both sides have to make some kind of platform-specific investments (e.g. access fee, resource, opportunity cost) in order to be able to interact directly with the other side. By ‘direct interaction’, the authors mean that the different sides retains control over the key terms of the interaction. Pure two-sided platforms simply provide the common marketplace, thereby determining buyer and seller affiliation. The control rights over non-contractible decisions (e.g. advertising, customer service) rest with the independent suppliers [6]. This is opposed to an intermediary, who takes over full control from the seller over the sale and the strategic variables, economic risk and ‘ownership’ of buyers [4]. This distinction between merchant and a two-sided platform is depicted in Fig. 1.

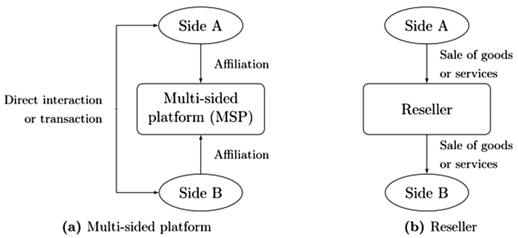


Fig. 1. MSPs vs reseller business model, based on [5]

Earlier academic work suggested definitions that rely on the pricing structure chosen by the intermediary [7], or emphasized the presence of indirect network effects between the two sides participating on the platform (among others, [8]), rather than on the division of control [5]. Network effects can either be same-side or cross-side, and positive or negative. According to that literature, MSPs are characterized by positive cross-side externalities, meaning that an increase in buyers on the platform makes the platform more valuable for the sellers, leading to a greater number of higher quality offered services/products, which consequently incentivizes more buyers to use the platform. This positive feedback loop leads to a self-growing user network, intrinsically making the platform more and more valuable. However, those earlier definitions struggled to distinguish multisided platforms from resellers [6].

Additionally, [6] provide a further extension to the earlier academic work by considering moving towards a marketplace mode as a strategic choice, rather than as a given characteristic of a firm. Previous work from the authors, being [4, 5, 9] had already stressed the importance of the strategic choice between the marketplace and reseller model. The choice is not a binary one, however. Amazon, for instance, has successfully adopted a hybrid mode, offering products under each of the two different modes [6]. Evaluating whether a company-of any kind- should pursue the MSP or reseller model, or where to position them on the continuum in between them, involves consideration of four fundamental trade-offs, as discussed in [4, 6, 9].

First, a multi-side platform model is preferable for products for which suppliers have a significant information advantage, relative to the intermediary. Amazon, for example, only sells a small portion of its electronic products as a reseller, since products in that category change constantly, leading to an information disadvantage for Amazon. Secondly, only if products have limited price- and marketing spillovers on other products, a marketplace business model is suitable. Resellers can exploit complementary relationships between products for their marketing and pricing strategies, which independent, uncoordinated suppliers on a platform cannot do. The third kind of products for which the multi-sided platform model is best suited are the long-tail products. The marginal cost advantage of a reseller having economies of scale, does not apply for low-demand products. Amazon, having a hybrid model, offers its high-demand products as a reseller, while acting as a marketplace for low-demand products. Finally, products from late-stage companies are best suited for the MSP model. The chicken-egg or catch-22 problem, as discussed in [9, 10], is known as the phenomenon that buyers will not join a firm's marketplace if it does not provide enough sellers, neither will sellers if it does not provide enough buyers. This makes that the early-stage marketplace faces unfavourable expectations. Besides considering the trade-offs between a MSP or reseller, [5] discusses the economic trade-offs between a MSP and other alternative business models, being vertically integrated firms and input suppliers.

As stated, an important feature of MSPs is the presence of cross-side or indirect network effects. Those network effects create high barriers to entry for aspiring platform providers. The biggest challenge is the aforementioned chicken-egg or catch-22 problem. The minimum set of participants, necessary for the platform to have self-sustaining growth, is called the 'critical mass'. Several strategies to secure critical mass are discussed in [11]. Ways to deal with the catch-22 problem include subsidizing a certain user group, by allowing temporary discounts for early adopters [10]. Challenges here are to determine which side to subsidize and for how long [12]. Furthermore, a familiar strategy could be to bribe 'must have' transactions partners to affiliate exclusively with your platform [10].

However, both of these network mobilization strategies require high investments and thus imply great risks. Therefore, [10] discusses alternative strategies to side-step the challenges that come with the network effect, by resorting to a different business model for an interim period. When applying the 'vendor to two-sided platform' strategy, the potential MSP starts selling products or services to customers on one side of the network. Once the critical mass on the first side is secured, the firm can launch mediation services between both sides of the network. Google used this strategy for its search engine, as it only incorporated paid-listing advertisements after attracting a huge end-user base. Secondly, the 'merchant to two-sided platform' strategies suggests that after building infrastructure and relationships, the merchant firm gives its suppliers more responsibility for managing inventory, pricing, and merchandising, hence shifting back the risks. Amazon successfully transformed into a two-sided platform following this strategy. Different flavours of the strategies, the pitfalls and guidelines on the transition are discussed thoroughly in [10].

Other typical risks and strategic challenges that come with the development of a MSP are discussed in [3, 10, 12], among others. They include challenges inherent in launching an MSP business, like how many sides to bring on board, the pricing structures, which functionalities and features to provide in its design, which governance rules to impose in order to protect itself against potential sources of market failure, how to satisfy conflicting interests, the treat of envelopment by an adjacent platform provider and winner-take-all dynamics. Furthermore, [13] provides meaningful insights in design and governance of platforms. Important to note is that, as the previous section described, the difficulties of an MSP business does not imply that building a solid non-MSP business is not possible [3].

2.2 Internet of Things

The Internet of Things (henceforth: IoT) is a paradigm that represent the interconnection of physical objects or 'Things', enabling the objects to connect and exchange data for various purposes. The preceding definition is not the generally accepted definition, as there is no common convention on what the IoT actually encompasses. The definition of IoT keeps expanding and has been extended far beyond the scope of the original focus of machine-to-machine communication without human intervention. A discussion on the multitude of alternative definition is given by [14].

IoT has enjoyed a tremendous amount of interest, fed by forecasts of leading advisory firms, such as Gartner, on the enormous potential of the industry, and prominent IoT-related takeovers like Nest by Google for $3.2 billion. The high IoT expectations are said to have created a hype, as it becomes clear that IoT implementations represent the challenging task of creating highly complex systems and coordinating technology, investment and talent across both space and time [15]. However, the Internet of Things does vastly expand the reach of information technology, offering enormous opportunities across a wide range of domains.

From a technological point of view, connected products require a new technology infrastructure [16]. The IoT technology stack is composed of multiple layers, usually presented by a four-layered model, as shown in Fig. 2. The first layer of the model includes the information generation by hardware, consisting of the 'Thing', equipped with IoT components such as sensors, actuators, processors and communication modules. Secondly, the communication or connectivity layer represents the plethora of communication standards and technologies to choose from when connecting 'things' to the cloud to transport information. Examples include link layer protocols (e.g. ZigBee, BLE), transport layer protocols (e.g. 6LoWPAN) and application layer protocols (e.g.

CoAP, MQTT). Thirdly, the software back-end layer consists of the software running on the manufacturer's or third party servers that processes and manages the device data, allowing service provision by IoT applications, i.e. the fourth layer. Finally, cutting across all layers, security must be vertically integrated in all layers, therefore being the fifth layer of the model [17].

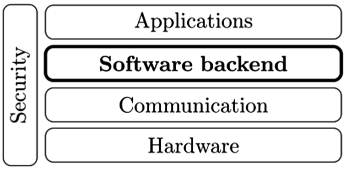


Fig. 2. IoT building blocks, based on [17]

Most of the first IoT offerings to come to market, mostly one specific application for one kind of thing or device, consisted of all building blocks provided and controlled by one single company. There was yet little existing infrastructure in place to support such offerings and in addition, it has the advantages for the end-users that (1) there are no compatibility issues, and (2) it provides a single point of contact [18]. However, a number of issues arise when following this vertical model. The latter advantage, indeed, also implies the risk of vendor-lock in, i.e. that the end-user is entirely dependent on the vendor for improvements, enhancements, or upgrades of its mostly proprietary technology. This leads to a low pace of innovation due to long time to market and slow cost declines [19]. Moreover, when a user intends to achieve a spectrum of tasks, several different systems are needed in parallel. Furthermore, those traditional vertical models often imply isolated data silos, preventing cross-vertical use of the data. Vertically oriented and mostly closed systems in a highly fragmented fashion hamper the growth of the IoT industry. Missing interoperability is considered to be one of the biggest threat to the valorisation of the potential benefits of IoT [20].

Development of horizontal IoT platforms has allowed that vertical value chains are evolving towards a value network comprising multiple stakeholders in an ecosystem [20]. Horizontal platforms can be defined as "cloud-based and on premise software packages and related services that enable and support sophisticated IoT services" [19, p. 13]. Indeed, IoT applications share a substantial part of their core functionalities. Incorporating those functionalities in a horizontal IoT platform therefore enables IoT application developers to concentrate on the aspects that differentiates their application and prevent unnecessary redundancy and duplication of transformation processing [21]. More importantly, horizontal IoT platforms (henceforth: IoT platform) reduce the complexity in developing, deploying, and managing IoT applications over the application lifecycle by providing a comprehensive set of application-independent functionalities. However, there is no standard configuration for an IoT platform, as it might consist of a variety of important building blocks like connectivity & normalization, device management, database, processing and action management, analytics, visualization, external interfaces and additional tools [17]. Hence, a vast amount of IoT platforms have been established, each addressing specific needs and areas of application [22]. [23] provides a list of 450 IoT platform companies across various geographies and industries.

2.3 Problem Statement

Current vertical IoT solutions are simple for users to acquire and apply. However, as users have become aware of the drawbacks of this 'one-stop shop' vertical approach when they seek to update or enhance their systems, there is has been a need for a horizontal approach. This approach allows rapid proliferation of new applications and businesses, but it needs to gain considerable traction in order to reach its promising potential. A prerequisite for adaption is a substantial number of open gateways to be in place before third-party developers can count on having an adequate market to serve. This reminds us of the catch-22 problem in multi-sided markets. Therefore, this paper attempts to investigate if the characteristics of a multi-sided platform model are applicable for horizontal IoT platforms. More specifically, in analogy with other industries, will the winner-takes-all dynamics set up the race for domination of the IoT market?

3 MSP Opportunities in the IoT Ecosystem

3.1 IoT Ecosystems

The famous and successful household brands in the introduction provided different authors with textbook examples for the theory of multi-sided markets. However, heterogeneous, multi-layered IoT markets exhibit a far more complex structure than B2C markets [24]. Hence, the trade-offs discussed in Sect. 2 do not necessarily apply for IoT market actors, especially since these reside in complex ecosystems.

IoT ecosystems can include various types of firms, including the telecom operators, application and service providers, as well as the platform providers and integrators. According to [25], business ecosystems are formed around a specific core, i.e. representing some assets commonly used by the ecosystem members, allowing positive network effects. Hence, the technology stack in Fig. 2 offers three potential assets around which an IoT ecosystem might be build, being (1) hardware platforms based on specific hardware, (2) connectivity standards based in the communication layer and (3) services provided on IoT software platforms. The Xbox gaming marketplace is created with the hardware gaming console. Apple with its superior hardware experience created massive multi-sided platforms based on hardware devices as iPod (iTunes) and iPhone (App Store). In the IoT market, firms like Geotab [26] have successfully adopted that strategy to create an ecosystem around its hardware device, with an app marketplace where data from the device is available for Geotab or any third party to develop new applications for that data. Ecosystems can also grow around the connectivity standards provided by alliances such as the IPSO Alliance and ZigBee Alliance. For instance, Proximus EnCo is building a marketplace around its LoRa standard with APIs from both Proximus and third party API providers. However, in what follows, the main focus will lay on the third category, IoT software back-end platforms, as indicated in Fig. 2 by the bold frame and text.

3.2 Advanced IoT Platforms

As stated in Sect. 2.2, there exist a vast array of IoT platforms, providing solutions for specific types of applications of specific vertical domains. Therefore, in making the analogy with a MSP model, it is important that by 'IoT platform', an 'Advanced' or mature (horizontal) IoT platform is intended, as depicted in Fig. 3. As advanced IoT platforms contain all of the important building blocks discussed in Sect. 2.2, they allow complementary third parties to interact with users by offering additional services, and therefore can be considered multi-sided. This definition of an IoT platform is in line with the definition of multi-sided platforms discussed in Sect. 2 and with the definition of an external (industry) platform as "products, services, or technologies developed by one or more firms, and which serve as foundations upon which a larger number of firms can build further complementary innovations and potentially generate network effects" [27, p. 4]. Therefore, they are also referred to as IoT Application Enablement Platforms (AEPs). Among IoT platforms, platforms like e.g. ThingWorx have dedicated application stores. Others allow the applications to be publicly shared (e.g. IFTTT), or will be enabling the charging of the end users of these applications [28]. Indeed, standard IoT platforms does not comprise such marketplace [24].

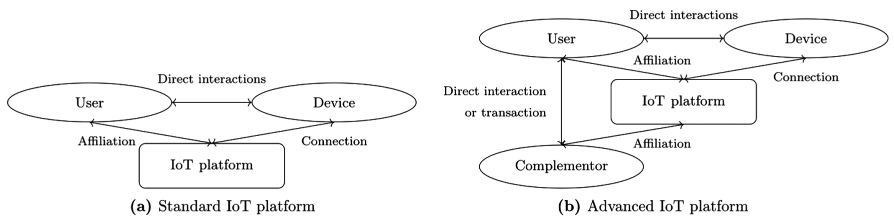


Fig. 3. Mature IoT platforms as multi-sided platforms [24]

3.3 Discussion

Besides compliance with the MSP definition and the presence of network effects in IoT ecosystems, analogous with the cross-side positive network effects, some other IoT market findings are in line with in the literature on multi-sided platforms. First, in accordance with [3], who state that high switching costs are necessary to keep all sides to the MSP, the network effects created by a technical standard in the IoT market, would make using multiple platforms ('multihoming') or switching from one platform to another difficult or costly [27]. An analogy is also found with [10] on staging MSPs, as vertical, 'standard' IoT platforms are currently being used as an interim model and leveraged to kickstart activity on a horizontal, 'advanced' offering (Fig. 3), analogous to the 'vendor to two-sided' platform strategy. Because of these elements, it is plausible to think that one or two leading multi-sided platforms are expected to become the core of the IoT ecosystem, corresponding to the winner-take-all dynamics, as discussed in [12]. Also, because of the highly fragmentated and complex nature of the IoT industry, multiple leading platforms are likely to coexists. Although large companies like Google, Amazon, IBM, Microsoft and SAP attempt, through their offerings, to become the IoT standard platform, such complex ecosystems are currently in the formation stage, and no single IoT platform has been identified to play the role of the core of the ecosystem. This is mainly, in line with the theory of [4, 6] on early-stage ventures, because IoT platforms are still in the phase of foster an user base, before opening their platforms to third-party supply side by means of a marketplace [24], as shown in Fig. 3. Another reason is that no single dominating set of standard protocols and interfaces has emerged yet [25]. The lack of such a generally accepted dominant design eventually prevent wider adoption of the IoT technologies, since the growth of the IoT market depends on the emergence of IoT ecosystem around common/dominant standards, platforms, and interfaces [25].

In what follows, four closely related trade-offs and challenges are discussed for IoT platforms aspiring to become a future leading multi-sided IoT platform, based on [24]; (1) platform orientation, (2) platform openness, (3) platform compatibility and (4) platform ecosystem. The first trade-off IoT platforms have to make is whether they focus on vertical, industry-specific use cases with highly specialized functionality, creating several unconnected and heterogenic Intranets of Things [21], or on use cases across different industries, hence providing more generic functionality. The latter offers the ultimate prize for leading platforms as it allows a software platform across all vertical application domains in the Internet of Things. Secondly, an IoT platform provider must position itself in the degree of openness of the platform to third-party contributions. Since the present market is vertically oriented with a high degree of fragmentation, one question is how vertical platforms will translate one to another. [24] suggest that due to the fragmentation, the positive network effects are not as strong. Therefore, some companies keep their application programming interfaces (APIs) private, hoping to lock-in customers into a proprietary technology and make money on license fees and a monopoly-like position in a later stage [17]. However, open platforms are considered more promising compared with the proprietary alternatives, since it enables faster integration of new IoT solutions across the application domains and speeds up the adoption of the software platform [28]. Due to APIs, the massive amount of data can be put to use in the most advantageous ways. In the example of Nest from the introduction of Sect. 2.2, it has built applications that can communicate with its proprietary products, like the smart thermostat, but also offers an API that third-party applications can use to communicate with the Nest devices. Thirdly, an IoT platform provider should make a choice on the technology. As a result of its ongoing fast evolution, the IoT technology architecture contains a multitude of connection protocols and low-level software languages and is currently far from being standardized [17]. Interoperability consists on three levels; (1) services and semantics, (2) networking and (3) communication. Common, interoperable standards are needed as a basis for IoT platforms to act as an AEP, breaking out of the vertical silos [24]. Without interoperability standards, such horizontal platforms would have to guarantee interoperability by trying to integrate all technologies through interfaces and the support of all possible communication protocols, an impossible task [24]. Therefore, companies are bundling their forces in consortia such as the AllSeen Alliance or the Industrial Inter-net Consortium in order to develop interoperable standards [17]. An important initiative is the BIG IoT [20, 29], a cross-domain ecosystem on top of IoT platforms, unlocking the potential of cross-platform and even cross-domain application developments and marketplaces to share and monetize IoT resources. Interoperability has been identified as one of the main challenges in the IoT industry, and in cloud applications in general [30]. As a result, enabling interoperability by integrating different standards is one of the most important functionalities of the IoT platform, allowing the IoT platform at the core of the IoT ecosystem [17]. Finally, IoT platforms need to realize the current IoT play is an ecosystem play in which they form the core and hence needs to deal with various stakeholders in it.

4 Conclusion and Outlook

Due to the promising potential of IoT, a large number of IoT solution in different application domains developed by start-ups, SMEs, large corporations and research institutions make their way into the market. A first important challenge for those companies lies in the choice of the IoT platform: at present, the market is in an early stage, with an abundance of fragmented solutions targeting specific vertical domains and/or specific types of applications. As discussed in this paper, whether or not IoT platforms will become a leading IoT platform will largely depend on the extent to which the IoT platforms are horizontally oriented, have a sufficient degree of openness, support compatibility and have managed to build an IoT ecosystem around them. However, currently the IoT market is not mature enough, and will only over time decide who the winners are in the IoT platform derby.

There are many promising directions in which this type of analysis can be extended. An update on the body of literature on MSPs might include the alternative roles, challenges and trade-offs for a company that aspires to become a leading MSP in complex (IoT) ecosystems. Furthermore, this paper solely focused on software back-end IoT platforms. Interesting extensions would include a discussion on hardware IoT platforms and platforms based upon connectivity standards. Finally, as the IoT market matures, a reflection on the market dynamics that will establish the winning platforms would provide meaningful insights.

References

1. Treacy, M., Wiersema, F.: Customer intimacy and other value disciplines. Harv. Bus. Rev. **71**, 84–93 (1993)
2. Hidding, G.J., Williams, J., Sviokla, J.J.: How platform leaders win. J. Bus. Strateg. **32**(2), 29–37 (2011)
3. Hagiu, A.: Strategic decisions for multisided platforms. MIT Sloan Manag. Rev. **55**(2), 71–80 (2014)
4. Hagiu, A.: Merchant or two-sided platform? Rev. Netw. Econ. **6**(2), 115–133 (2007)
5. Hagiu, A., Wright, J.: Multi-sided platforms. Int. J. Ind. Organ. **43**, 162–174 (2015)
6. Hagiu, A., Wright, J.: Marketplace or reseller? Manag. Sci. **61**(1), 184–203 (2015)

7. Rochet, J.: Two-sided markets: a progress report*. RAND J. Econ. **37**(3), 1–42 (2005)
8. Armstrong, M.: Competition in two-sided markets. RAND J. Econ. **37**(3), 668–691 (2006)
9. Hagiu, A., Wright, J.: Do you really want to be an eBay? Harv. Bus. Rev. **91**(3), 102–108 (2013)
10. Eisenmann, T.R., Hagiu, A.: Staging Two-Sided Platforms, pp. 1–12 (2018)
11. Evans, D.S., Schmalensee, R.: Matchmakers: The New Economics of Multisided Platforms. Harvard Business Review Press, Brighton (2016)
12. Eisenmann, T., Parker, G., Van Alstyne, M.W.: Strategies for two- sided markets. Harv. Bus. Rev. **84**(10), 12 (2006)
13. Schreieck, M., Wiesche, M., Krcmar, H.: Design and governance of platform ecosystems – key concepts and issues for future research. In: ECIS Proceedings (2016)
14. Atzori, L., Iera, A., Morabito, G.: The Internet of Things: a survey. Comput. Netw. **54**, 2787–2805 (2010)
15. Manyika, D., Chui, J., Bisson, M., Woetzel, P., Dobbs, J., Bughin, R., Aharon, J.: The Internet of Things: mapping the value beyond the hype. McKinsey Global Institute (2015)
16. Porter, M.E., Heppelmann, J.E.: How smart, connected products are transforming competition. Harv. Bus. Rev. **92**, 64–88 (2014)
17. Lasse Lueth, K., Kotzorek, J.: The central backbone for the Internet of Things (2015)
18. Nedeltcheva, G.N., Shoikova, E.: Models for innovative IoT ecosystems. In: Proceedings of the International Conference on Big Data and Internet of Thing, pp. 164–168 (2017)
19. Lucero, S.: IoT platforms: enabling the Internet of Things. IHS Technology Whitepaper, pp. 1–19 (2016)
20. Schladofsky, W., Mitic, J., Megner, A.P., Simonato, C., Gioppo, L., Leonardos, D., Bröring, A.: Business models for interoperable IoT ecosystems. In: Podnar Žarko, I., Broering, A., Soursos, S., Serrano, M. (eds.) InterOSS-IoT 2016. LNCS, vol. 10218, pp. 91–106. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56877-5_6
21. Berkers, F., Roelands, M., Bomhof, F., Bachet, T., Van Rijn, M., Koers, W.: Constructing a multi-sided business model for a smart horizontal IoT service platform. In: 2013 17th International Conference on Intelligence in Next Generation Network, ICIN 2013, pp. 126–132, October 2013
22. Wortmann, F., Flüchter, K.: Internet of Things: technology and value added. Bus. Inf. Syst. Eng. **57**(3), 221–224 (2015)
23. IoT analytics: IoT Platform Company List 2017 (2017). https://iot-analytics.com/product/list-of-450-iot-platform-companies/. Accessed 16 Apr 2018
24. Schreieck, M., Hakes, C., Wiesche, M., Krcmar, H.: Governing platforms in the Internet of Things BT - Software business, pp. 32–46 (2017)
25. Mazhelis, O., Luoma, E., Warma, H.: Defining an Internet-of-Things ecosystem. In: Andreev, S., Balandin, S., Koucheryavy, Y. (eds.) NEW2AN/ruSMART -2012. LNCS, vol. 7469, pp. 1–14. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32686-8_1
26. GPS Fleet Tracking and Management | Geotab. https://www.geotab.com/. Accessed 11 Apr 2018
27. Gawer, A., Cusumano, M.A.: Industry platforms and ecosystem innovation. J. Prod. Innov. Manag. **31**(3), 417–433 (2014)
28. Mineraud, J., Mazhelis, O., Su, X., Tarkoma, S.: A gap analysis of Internet-of-Things platforms. Comput. Commun. **89–90**, 5–16 (2016)
29. Bröring, A., et al.: Enabling IoT ecosystems through platform interoperability. IEEE Softw. **34**(1), 54–61 (2017)
30. Loulloudes, N., Sofokleous, C., Trihinas, D., Dikaiakos, M.D., Pallis, G.: Enabling interoperable cloud application management through an open source ecosystem. IEEE Comput. Soc. **19**, 54–59 (2015)

Towards Context-Aware Vehicle Navigation in Urban Environments: Modeling Challenges

Ivan Garvanov[1(✉)], Christo Kabakchiev[2], Boris Shishkov[3,4], and Magdalena Garvanova[1]

[1] Faculty of Information Sciences, University of Library Studies and Information Technologies, Sofia, Bulgaria
{i.garvanov,m.garvanova}@unibit.bg

[2] Faculty of Mathematics and Informatics, Sofia University "St. Kliment Ohridski", Sofia, Bulgaria
ckabakchiev@yahoo.com

[3] Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Sofia, Bulgaria

[4] Institute IICREST, Sofia, Bulgaria
b.b.shishkov@iicrest.org

Abstract. Currently, urban vehicle navigation is considered important because smoothness of traffic is needed for the sake of safety and cleanliness of air; in this regard, people essentially count on GPS navigation. Nevertheless, due to signal building shadows, GPS signal disturbances may occur in an urban environment; then alternative navigation options should be considered. It is therefore a matter of context-awareness to establish whether or not GPS navigation is available and adjust the delivered navigation services accordingly. In this paper we propose a top-down approach for modeling context-aware vehicle navigation systems; firstly, a high-level technology-independent conceptual model is derived (featuring context-awareness in general) and then this model is mapped towards corresponding lower-level technical solutions. Even though research materials, touching upon context-awareness or vehicle navigation, can be found in literature, we miss holistic approaches that align high-level (conceptual) goals with navigation-specific (technical) solutions; hence, our proposal is considered a step forward in this regard. We also propose an innovation concerning the technical aspects of navigation. Our proposed approach is partially validated by means of experiments.

Keywords: IT service · GPS navigation · Context-awareness

1 Introduction

The **`real-time acquisition of relevant data`** is important with regard to the current **`Internet-of-Things`** (IoT) developments that concern some IT (Information Technology) services ("services", for short) [7]. Such data acquisition is facilitated by **`GPS`** (Global Positioning System) **`technologies`** [8] and **`sensor technologies`** [10]. This inspires **`IoT-connected car solutions`** that enhance driving experiences [1]. Those developments are currently receiving much

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 382–389, 2018.
https://doi.org/10.1007/978-3-319-94214-8_29

attention not only among drivers but also within some leading (software) development companies who are challenged by the goal of delivering advanced vehicle navigation platforms, for example: the *Microsoft Connected Vehicle Platform* [3]. Such efforts are justified by the **increasing needs for better road traffic management** in its broadest sense, addressing concerns such as *travel time*, *congestion*, *safety*, and *pollution* [4].

Urban vehicle navigation is hence considered important because *smoothness of traffic* is needed for the sake of *safety* and *cleanliness of air*; in this regard, people essentially count on **GPS navigation** [2]. Nevertheless, due to **signal building shadows**, GPS signal **disturbances** may occur in an urban environment; then alternative navigation options should be considered [5]. This would mean using GPS navigation when it is possible and "switching" to an alternative service when GPS navigation is unavailable. It is therefore a matter of context-awareness to establish what is available and adjust the delivered navigation services accordingly.

We thus argue that **context-awareness** [11] and **service-orientation** [12] are relevant with regard to that challenge because *context-aware service solutions* assume the *system* capability of **adapting its service provisioning to the situation at hand**.

As it is seen from the above discussion and cited references, research materials, touching upon context-awareness or vehicle navigation, can be found in literature. What in our view is missing nevertheless are **holistic approaches** that align *high-level (conceptual) goals* with *navigation-specific (technical) solutions*.

We claim that *a service-orientation-solution perspective would be adequate in this regard because services that are composed at high-level have underlying low-level technical components* [14].

In this paper we propose a **top-down approach for modeling context-aware vehicle navigation systems**; firstly, a high-level technology-independent conceptual model is derived (featuring *context-awareness* in general) and then this model is mapped towards corresponding lower-level technical solutions; in this, we narrow the scope to consider in particular the challenge of resolving GPS signal disturbances in an urban environment and we propose an *innovation concerning the technical aspects of navigation*.

Hence, the contribution of the current paper is two-fold: (i) A proposed *holistic approach spanning over* ***technology-independent*** *and* ***navigation-specific*** *issues*; (ii) A proposed *innovation* (related to the approach) that concerns *the alternative* (*non-GPS-based*) *navigation*.

Our proposed approach is partially validated by means of **experiments**.

The remaining of the current paper is organized as follows: In Sect. 2, we introduce context-aware systems and on that basis, we present a technology-independent bi-service model that we consider relevant to the urban vehicle navigation. Then, staying consistent with that model, we present our proposed vehicle-navigation-related approach in Sect. 3 (featuring two abstraction levels). In Sect. 4, we partially validate our proposed approach, considering experimental results. Finally, in Sect. 5, we present the conclusions.

2 Context-Aware Systems

Traditional (software) systems used to assume a “static” user whose needs are not changing over time, this in turn meaning that the services delivered to the user are “fixed”. In contrast, **User-Driven Context-Aware Systems** (“context-aware systems”, for short) consider the **user in his or her context** – this meaning that the user may be in different situations (for example: “*user walking*”, “*user driving*”, “*user sleeping*”) that ask for different service variants. As it concerns context-aware systems, we refer to **another paper** (particularly its Subsect. 2.2) from the *current proceedings book* where they are exhaustively *introduced* and *discussed* [9].

With regard to urban vehicle navigation (see Sect. 1), we have **exactly two context states**, namely: (a) *GPS navigation is available for the user*; (b) *GPS navigation is unavailable for the user*. Hence, if we “generalize” this, we arrive at a **bi-service context-aware behavior**, as illustrated in Fig. 1, using the notations of UML – Activity Diagram [13].

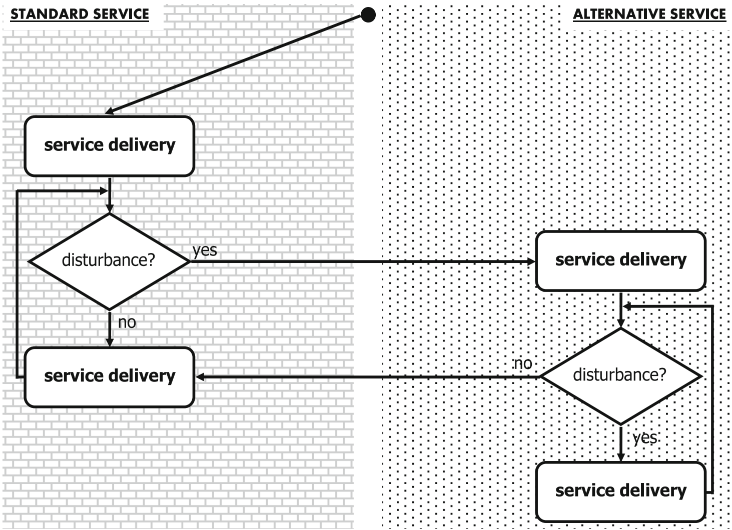


Fig. 1. A technology-independent model of a bi-service context-aware system

As suggested by the figure, we assume that at a “start moment”, standard service delivery is available. We assume as well that the system periodically checks the service delivery situation. Then, if there are no disturbances, the standard service delivery would continue. All this concerns the left “swim lane” (background: bricks) from the two swim lanes that are arranged vertically on the diagram. In case of disturbances nevertheless (when it would be no longer possible to use the standard service of full value), alternative service delivery should start – see the right “swim lane” (dotted background). Then this service delivery would only continue as long as the mentioned disturbances are present (periodical checks would be continuing). At the moment, the disturbances are gone, standard service delivery is resumed.

This technology-independent model will be mapped towards lower-level navigation-specific technical solutions. This will be done in the following section.

3 Vehicle Navigation Approach

We consider the context-aware system model presented in the previous section, a suitable basis for approaching urban vehicle navigation: the “standard” service would be the **`GPS navigation`** and the “alternative” service would be the so called **`smart navigation`**, as exhibited in Fig. 2. Further:

- We assume that the vehicle under consideration is “currently” moving along streets /boulevards in an urban environment, being GPS-navigated, and this is seen from the start sign (black disk) in the figure, pointing to the activity: “GPS navigation”. Such a navigation is possible if at least 4 satellites (out of the 24 GPS satellites that are circling the Earth) are “visible” from the GPS receiver installed in the vehicle. Periodically the signal strength is checked and if at least 4 satellites are “visible”, the GPS navigation would continue, as suggested by the figure (the left “swim lane”).

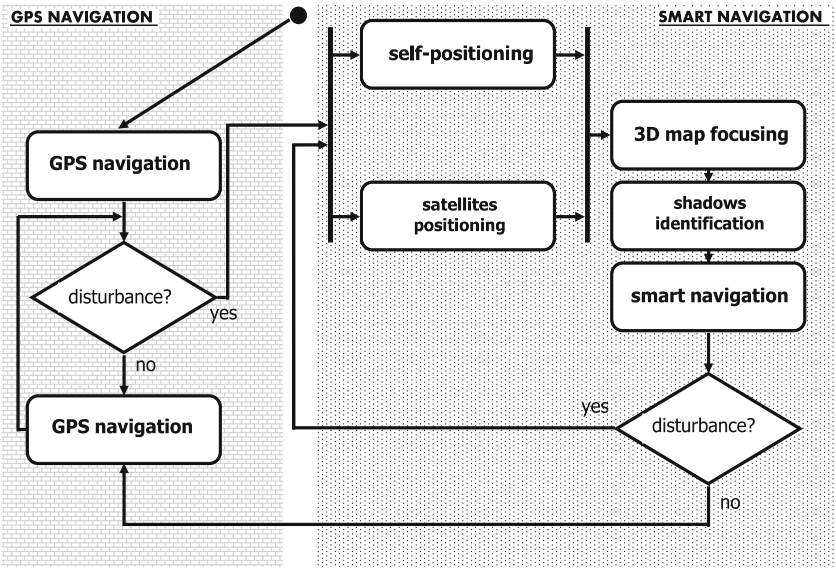


Fig. 2. Navigation approach represented at a high abstraction level

- Nevertheless, caused by (tall) buildings, signal “shadows” may occur that would substantially lower the signal strength of particular satellites; such a disturbance may lead to a situation when less than 4 satellites are “visible” and hence, a full-value GPS navigation would no longer be possible; then, as suggested by the

figure, the *smart navigation service* would be triggered (the right "swim lane" in the diagram):
- In parallel, the vehicle positioning and the positioning of the visible satellites are established, pointing to the last moment when full-value GPS signal (at least 4 visible satellites) was available (this data is maintained by means of logging);
- Assuming the availability of 3D maps, a relevant 3D map is considered with establishing the right focus, based on the above-mentioned positioning data;
- Then the signal shadows in the corresponding area are identified (this is possible because of the availability of the vehicle positioning data, the satellites positioning data, and an appropriately focused relevant 3D map);
- All this is sufficient for enabling a smart navigation (it will be elaborated further in this section);
- All this is maintained while full-value GPS signal is unavailable.

- When the vehicle would again have visibility, the GPS navigation service is resumed.

Further, even though the SMART NAGIVATION activity is modeled as one activity at the high abstraction level – see Fig. 2, it should be possible to map it towards lower-level elaborations; this is needed for the sake of presenting the proposed navigation-specific innovation that is supported by experimental results (see Sect. 4).

We propose establishing the smart navigation as a 2-CHANNEL PROCESS where:

- The shadows identification (see Fig. 2) help in computing the GPS building shadows while
- Real-time measurements help in their run-time detection and parameters estimation.

This is illustrated in Fig. 3:

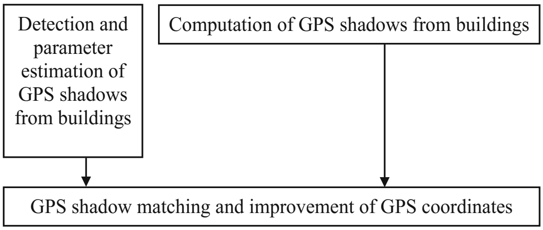


Fig. 3. Representing the "smart navigation" activity at a lower abstraction level

Those two inputs are complementary to each other because the **`computation`** is important for grasping the shadows and the corresponding shadow "entrance" point while the **`measurements`** are important for capturing the "current" position of the vehicle.

In the following section, we will partially validate our proposed approach (with a particular stress on the measurements part), by means of conducted experiments.

4 Experiments

As above mentioned, in the current section, we provide partial experiments-driven validation of the proposed navigation approach.

4.1 Briefing

We consider a moving *car equipped with a GPS receiver* and an *antenna* installed on top of the car. The car's average speed is *30* km/h and the car is in an *urban environment*, as it is shown in Fig. 4 (left). The *movement trajectory* goes in proximity to *two 19 story buildings*. The *positioning of the satellites* during the experiments is shown in Fig. 4 (right); the satellites' position with regard to the GPS receiver is shown in Fig. 4 (right). Those figures show that the signals coming from the 2nd, the 6th, the 12th, and the 24th satellites are highest on the horizon and their signals are to essentially contribute with regard to the determination of the coordinates of the GPS receiver. Further, as it is seen in Fig. 4 (left), the first three of the mentioned satellites (2, 6, and 12) get "shadowed" by high buildings while the 24th satellite is "visible" during the whole experiment.

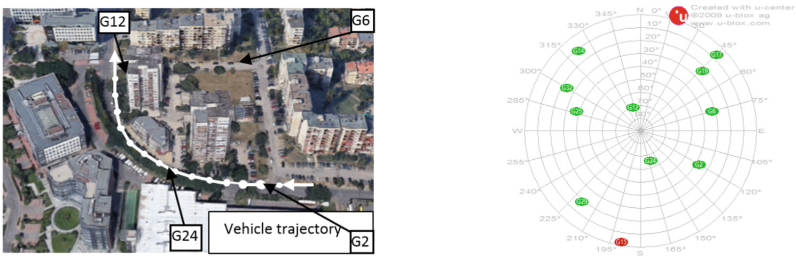


Fig. 4. Car trajectory (left); Satellites position (right)

4.2 Results

The positions of the *GPS shadows* concerning the 2nd, the 6th, and the 12th satellites are to look as shown in Fig. 5 (left). The positioning of those shadows should be calculated using an algorithm related to the 3D model of the buildings in the neighborhood.

The GPS signals received from all *four satellites* (*2*, *6*, *12*, and *24*) have been processed using an *algorithm* (proposed in [6]) featuring the *derivation of parameters' estimation that concerns the GPS shadows*; the results are shown in Fig. 5 (right) – they have been achieved, by applying an integration procedure assuming *integration time of 200* ms. As it is seen from the figure, the signal strength of the 24th satellite is more or less constant and is comparatively high throughout the whole experiment – GPS shadows are not noticed. Further, the signal of the 12th satellite points to a GPS shadow in the *8* s–*10* s. *measurement interval* while the signal of the 6th satellite – to *two GSP shadows* as follows: The first shadow is in the *12* s–*17* s. *measurement interval* and the second shadow is in the *19* s–*22* s. *measurement interval*. Finally, the signal of the 2^{nd} satellite points to a GPS shadow in the *11* s–*30* s. *measurement interval*. The GPS shadows of the *2^nd^* and the *6^th^* satellites are *overlapping* for a period of time, and this is seen both on Figs. 5 (left) and (right).

Inspired by those experimental results, we claim that the availability of *a-priori* information featuring the *location of the GPS shadows* (reflected in the 3D area model) would make it possible to know at a time the coordinates of the *GPS receiver*, with high degree of precision, if in the course of the vehicle movement, we identify the point of intersection between a shadow of consideration and a street, and we bring together computation-driven information and measurements-driven information, for achieving a reliable smart positioning.

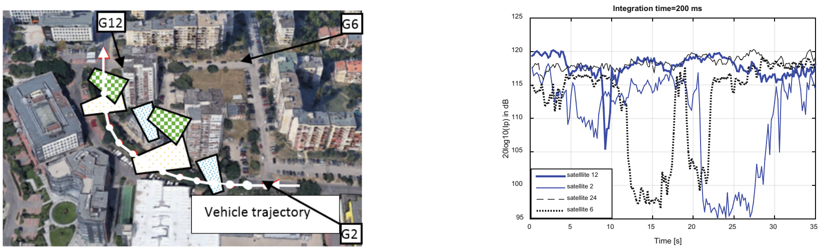


Fig. 5. The GPS shadows

5 Conclusion

In the current paper, we brought together context-aware systems design and navigation technologies, proposing an innovative design of IoT-driven IT services that concern road auto-traffic. In particular, we have considered two valid context states alternative to each other, namely: "GPS navigation is possible" and "GPS navigation is unavailable", proposing accordingly a service adaptation mechanism; when GPS navigation is unavailable, an alternative smart navigation is considered that is supported by both a-priori computation and run-time measurements. Further, we have brought forward our proposed approach top-down – starting from a high-level technology-independent conceptual model and reflecting it to a lower (navigation-specific) technical level. This is considered useful because urban vehicle navigation is driven by navigation-specific technical solutions but its usability concerns high-level real-life business processes. Finally, we have partially validated our proposed approach by means of conducted experiments. Still, this is research in progress and we plan to develop further the proposed approach (aiming at a broader applicability that goes beyond urban vehicle navigation) in the future, and to also carry out more experiments in order to make the validation more solid.

Acknowledgement. This work is supported by Project DFNI-T 02/3/2014.

References

1. Adjrad, M., Groves, P.: Intelligent urban positioning: integration of shadow matching with 3D-mapping aided GNSS ranging. J. Navig. **71**(1), 1–20 (2018)
2. Chang, I.C., Tai, H.T., Hsieh, D.L., Yeh, F.H., Chang, S.H.: Design and implementation of the travelling time- and energy-efficient android GPS navigation app with the VANET-based A* route planning algorithm. In: Proceedings of the International Symposium on Biometrics and Security Technologies, Chengdu, pp. 85–92 (2013)
3. Connected vehicles the website of microsoft connected vehicle platform (2018). https://www.microsoft.com/enus/internet-of-things/connected-vehicles
4. Costea, I.M., Nemtanu, F.C., Dumitrescu, C., Banu, C.V., Banu, G.S.: Monitoring system with applications in road transport. In: Proceedings 20th International IEEE Symposium for Design and Technology in Electronic Packaging (SIITME), Bucharest, 2014, pp. 145–148 (2014)
5. Garvanov, I., Dimitrov, K., Behar, V., Kabakchiev, Ch.: Comparative analysis of object shadows obtained by GPS and sound signals. In: Proceedings Signal Processing Symposium (SPSympo), Jachranka, 2017, pp. 1–4 (2017)
6. Garvanov, I., Kabakchiev, Ch., Behar, V., Garvanova, M.: Target detection using a GPS forward-scattering radar. In: IEEE Proceedings of the Second International Conference Engineering and Telecommunications – EnT, Moscow-Dolgoprudny, Russia (2015)
7. IoTDI, 2nd International Conference on Internet-of-Things Design and Implementation. ACM/IEEE (2017)
8. Kabakchiev Ch. et al.: Data mining classification of cars based on GPS shadows in forward scatter radar systems. In: Proceedings of the 18th International Radar Symposium (IRS), Prague, pp. 1–9 (2017)
9. Shishkov, B., Larsen, J.B., Warnier, M., Janssen, M.: Three categories of context-aware systems. In: Shishkov, B. (ed.) Proceedings of BMSD 2018 - 8th International Symposium on Business Modeling and Software Design. Lecture Notes in Business Information Processing, vol. 319. Springer, Cham (2018)
10. Shishkov, B.B., Mitrakos, D.: Towards context-aware border security control. In: Proceedings of the BMSD 2016 - 6th International Symposium on Business Modeling and Software Design. Scitepress (2016)
11. Shishkov, B., van Sinderen, M.: From User Context States to Context-Aware Applications. In: Filipe, J., Cordeiro, J., Cardoso, J. (eds.) ICEIS 2007. LNBIP, vol. 12, pp. 225–239. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88710-2_18
12. Shishkov, B., van Sinderen, M.J., Quartel, D.: SOA-driven business-software alignment. In: Proceedings of the ICEBE 2006, IEEE International Conference on e-Business Engineering. IEEE (2006)
13. UML, The Unified Modeling Language (2018). http://www.uml.org
14. Zhang, L.J., Zhang, J., Cai, H.: Service-oriented architecture. In: Zhang, L.J., Zhang, J., Cai, H. (eds.) Services Computing. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-38284-3_5

Design Options of Store-Oriented Software Ecosystems: An Investigation of Business Decisions

Bahar Jazayeri[1(✉)], Olaf Zimmermann[2], Gregor Engels[1], Jochen Küster[3], Dennis Kundisch[1], and Daniel Szopinski[1]

[1] Paderborn University, Paderborn, Germany
{bahar.jazayeri,gregor.engels,dennis.kundisch,daniel.szopinski}@upb.de
[2] University of Applied Sciences of Eastern Switzerland, Rapperswil, Switzerland
ozimmerm@hsr.ch
[3] Bielefeld University of Applied Sciences, Bielefeld, Germany
jochen.kuester@fh-bielefeld.de

Abstract. Nowadays companies like Apple create ecosystems of third-party providers and users around their software platforms. Often online stores like Apple App Store are created to directly market third-party solutions. We call such ecosystems *store-oriented software ecosystems*. While the architecture of these ecosystems is mainly derived from business decisions of their owners, ecosystems with greatly different architectural designs have been created. This diversity makes it challenging for future ecosystem providers to understand which architectural design is suitable to fulfill certain business decisions. In turn, opening a platform becomes risky while endangering intellectual property or scarifying quality of services. In this paper, we identify three main design options of store-oriented software ecosystems by classifying existing ecosystems based on similarities in their business decisions. We elaborate on the design options, discuss their main contributions, and provide exemplary ecosystems. Our work provides aspiring ecosystem providers with the reusable knowledge of existing ecosystems and helps them to take more informed architectural decisions and reduce risks in future.

Keywords: Software ecosystems · Reusable designs · Variabilities

1 Introduction

Software ecosystems have become an emerging architectural approach for many companies to grow. The term *software ecosystem* is inspired from ecological ecosystems that are the result of an interplay between organisms as well as interactions with a physical environment [1]. Comparably, a software ecosystem

This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Center "On-The-Fly Computing" (CRC 901).

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 390–400, 2018.
https://doi.org/10.1007/978-3-319-94214-8_30

consists of a software platform, a set of third-party providers in service to a community of users [2]. For instance, Apple Inc. created an ecosystem of third-party developers in service to the users of mobile Apps around the iOS and MacOS platforms. In practice, many of software ecosystems include online stores such as Apple App Store to make third-party solutions directly available to their users. We call them *store-oriented software ecosystems*.

Recently, such ecosystems have widely been created in different application domains. For instance, while mobile App ecosystems (like Apple and Google) are among the most popular ones with millions of users, ecosystems are created around open source platforms (e.g., Eclipse and Mozilla Firefox[1]) and enterprise applications (e.g., Salesforce[2] and Cloud Foundry[3]). Furthermore, existing ecosystems use different terminologies. For example, third-party providers are known under different terms such as independent developers, committers, and partners. Third-party solutions are also called by plug-ins, Apps, add-ons, etc. In this paper, we refer to third-party providers as *extenders* and third-party solutions as *extensions*. Moreover, in support of different business decisions, ecosystems with diverse architectural designs have been created whereas different ecosystems include very different software features [3]. For instance, in commercial ecosystems like Apple and Salesforce, revenue streams are channeled to the extenders using billing features. However, billing is a negligible feature in open source software ecosystems like Eclipse and Mozilla Firefox.

This diversity makes it challenging for future ecosystem providers to understand which architectural design is suitable to fulfill certain business decisions. In turn, opening a platform becomes risky while endangering the intellectual property or scarifying quality of products and services [4]. To tackle the diversity, some works [5–7] study variabilities of business decisions and software features in software ecosystems. However, the relation between these two is hardly known. Therefore, it is not clear how to design a store-oriented software ecosystem that fits to an ecosystem provider's needs. This hinders systematic development of customized ecosystems in the future.

In this paper, we investigate 111 existing store-oriented software ecosystems and classify them based on similarities in their business decisions. To do so, we use a variability model identified by Jazayeri et al. [6,8] for variable design decisions of store-oriented software ecosystems. Modeling recurring architectural decisions for knowledge reuse has been proposed by Zimmermann et al. in [9]. We notice that similar business decisions result in similar software features. Our work abstracts from this knowledge and suggests an architectural landscape for each group of ecosystems. Such an architectural landscape resembles a macro view to the system and its interaction with an environment [10, p. 254]. The contribution of our work is twofold: (a) Our results show that three design options of store-oriented software ecosystems are frequently applied in practice, which we call *resale software ecosystem*, *partner-based software ecosystem*, and

[1] www.mozilla.org/. Accessed April 2018.
[2] www.salesforce.com/. Accessed April 2018.
[3] www.cloudfoundry.org/. Accessed April 2018.

open source software-based ecosystem (shortly *OSS-based software ecosystem*). We discuss each design option in detail, elaborate on its main contribution in terms of a business goal, and provide real-world examples. (b) Our work provides a practical mean to aspiring ecosystem providers by sharing the knowledge on the reusable architectural designs to perform more informed decision-making while creating their own ecosystems. Specifically, each design option contributes to a main business goal. This knowledge can also be used by existing ecosystem provider to assess their own architectural designs with respect to the business goals discussed in this paper. In the following, we investigate and classify the existing ecosystems in Sect. 2. Section 3 presents the frequently applied design options whereas Sect. 4 elaborates on the overall contributions of the design options. The paper is concluded in Sect. 5 and future directions are addressed.

2 Classifying Store-Oriented Software Ecosystems Based on Variabilities of Business Decisions

We aim at identifying the main design options of store-oriented software ecosystems. We take real-world ecosystems as the basis of our investigation. Initially, we collect a list of ecosystems by first defining our search terms and then performing web searches. To capture as many ecosystems in as diverse application domains as possible, we derive the search terms by using a taxonomy for software ecosystem introduced by Bosch [1]. The author classifies open software platforms into three types, i.e., desktop, web, and mobile. Whereas, each type appears in three dimensions, i.e., operating system, application, and end-user programming. Examples of our search terms are “top operating systems”, “top end-user programming software”, and “top cloud computing platforms”.

Afterwards, we inspect the ecosystems with respect to their business decisions based on a variability model in [6,8]. We use the variability model, because it provides a concrete set of variabilities that we can use to compare and group the ecosystems. Decisions on such variabilities derive the main architectural landscape of an ecosystem. Table 1 shows the variabilities in form of *variation points* and *variants*. A variation point is the subject of a variability whereas a variant is the object of the variability, i.e., a concrete business decision. Accordingly, **extender** defines who extends a platform’s functionality. **Openness** specifies whether a platform is open source. **Fee** defines the main costs of participating in an ecosystem. **Feedback Loop** determines software features that enable a positive feedback loop between users and extenders. In the context of markets, this happens when more users use the platform, and thereby, the number of extenders increases [11]. Finally, **Knowledge Sharing** defines software features that enable knowledge-sharing among users or extenders to communicate and generate common values [12].

We consider whether and how the ecosystems in our list realize every variation point in Table 1. Zimmermann et al. [9] propose an approach to capture and model recurring architectural design decisions for the purpose of knowledge reuse. This allows us to detect three groups of ecosystems that provide similar

Table 1. Variabilities of business decisions and real-world instances from each group of ecosystems ("–" means that a variant is not realized in the architecture)

	Variation Point	Variant	Resale Software Ecosystems	Partner-Based Software Ecosystems	OSS-Based Software Ecosystems
Determinant Design Decisions of Business Architecture	**Extender** Who extends the platform's functionality?	**Trusted Partner**	Hardware / software suppliers, strategic partners, system integrator, & resellers	Hardware /software suppliers, strategic partners, system integrator, & resellers	–
		Independent Developer	High number of independent developers	–	High number of independent developers
	Openness Is the source code open?	**Open**	Fully or partially open source libraries	Access to the source code and API Reference for partners	Source code on a public repository, e.g., GitHub or SourceForge
		Closed	Fully or partially closed source libraries	No access for independent developers	–
	Fee What are the main costs of participating in the ecosystem?	**Entrance Fee**	Same payment for everyone / One time payment / periodic payment	Membership in partner programs / different payments for different partners	–
		Platform Fee	Usually the same for everyone	Different payments for different platform editions, e.g., standard & enterprise	–
	Feedback Loop Facilitator Which software features enable a positive feedback loop between the users and extenders?	**Rating & Reviewing**	Number of downloads, likes. Star system, reviewing	–	Number of forks / sharing / likes (bookmarks, favourites)
		Ranking	List of featured / popular / new extensions	–	–
		Market Analytics	–	Statistics on user visits (No. of clicks, time spent on a webpage). Consulting services on market trends. Mail server to keep users in contact with partners	–
		Version Control Management	–	–	E.g., SVN, Git, Mercurial, Perforce, TortoiseCVS, TeamForge, etc.
		Ticket and Issue Tracking System	–	–	Test planing, bug tracking, notification interfaces, e.g., Jira, Bugzilla, Apache Bloodhound, and Roundup
	Knowledge Sharing Which software features enable knowledge sharing inside the communities of users or extenders?	**Documentation Framework**	Developer manuals, wiki	Partner portal: Documentation and guidelines on software development frameworks (requiring authorized access)	Developer manuals, wiki, public resources from open source software community
		Q&A Forums	Developer forum, user forum, idea forum	–	Public developer forum, e.g., StackOverflow

sets of instances for the variation points. In the next section, we discuss each group of ecosystems in detail. A complete list of search terms and the list of ecosystems can be found in our dataset [13].

3 Design Options of Store-Oriented Software Ecosystems

Three major design options of store-oriented software ecosystems emerge from our investigation in Sect. 2. In the following, we present the design options in order of their popularity among our list of ecosystems. We describe each design option by answering two questions: (a) What is the architectural landscape of the design option? We abstract from the knowledge of variabilities and suggest an architectural landscape for the design option using the UML notation. As a part of it, thought bubbles are used to refer to the actors' business goals of participating in an ecosystem, e.g., "*I want to access a big market of users*". (b) What are exemplary real-world ecosystems applying the design option? While naming exemplary ecosystems, we elaborate on one ecosystem in detail.

3.1 Resale Software Ecosystem

37% of ecosystems in our list provide software products and services to a mass number of end-users. In addition, a mass number of extenders develops extensions on top of the platforms. After the extensions are developed, they are sold several times. We call this group of ecosystems *resale software ecosystems*.

(a) *Architectural landscape*: Table 1 provides details on how existing ecosystems realize the variants of the variability model. Figure 1(a) outlines the architectural landscape of resale software ecosystems. **Extender**: Both trusted partners and independent developers extend the platforms and publish their extensions on the stores, where a mass number of end-users can access them. **Fee**: In general, resale software ecosystems can be commercial or non-commercial. A majority of 82% in our list is commercial, while the users have to pay for the extensions. There might be an entrance fee and platform fee, which varies based on an ecosystem provider's strategies. **Feedback Loop Facilitator**: In order to create a positive feedback loop between the mass number of extenders and end-users, software components like rating and ranking are employed. **Knowledge Sharing**: Q&A forums such as developer and user forums support knowledge sharing and social interaction between the end-users and developers.

Further Characteristics: A majority of providers of resale software ecosystems, i.e., 84%, offers a family of software products. In support of multiple platforms, the stores are divided to sub-stores, each for extensions of one platform. Since the developers work highly independent of the ecosystem providers, another concern is to separate their revenue streams. This is achieved by including billing and purchase features on the stores.

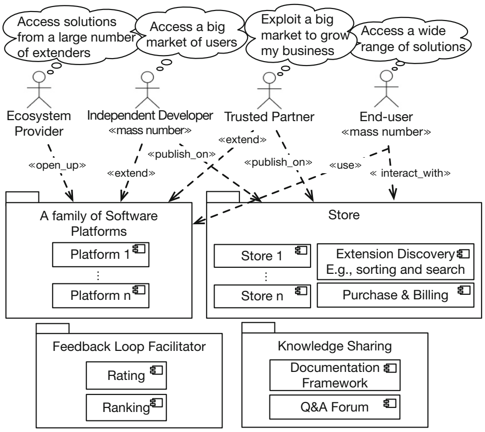
(b) *Exemplary real-world ecosystems*: Apple Inc. is the provider of a resale software ecosystem while the iOS and MacOS are the software platforms. There are App Store and Mac App Store, where third-party software and digital products are published. Figure 1(b) provides concrete instances of the variants in the Apple ecosystem. IBM, Salesforce, and Samsung are examples of Apple's trusted partners. The source code is mainly closed. Developers pay an annual entrance fee and entering the ecosystem as a user requires purchasing Apple devices. In addition, a feedback loop is created using a star system and different ranking categories like featured Apps. Apple Developer Forum[4] and Support Communities[5] are examples of Q&A forums. Further examples of the resale software ecosystems are Adobe, Salesforce, Facebook, Esri, Informatica, Autodesk, and FreshDesk.

3.2 Partner-Based Software Ecosystem

Another group of ecosystems is the ecosystems of carefully selected partners. This includes 35% of our list. We call them *partner-based software ecosystems*.

[4] developer.apple.com/devforums/. Accessed April 2018.

[5] discussions.apple.com/. Accessed April 2018.



(a) Architectural Landscape

Variation Point	Variant	Apple
Extender	Trusted Partner	IBM, salesforce, Samsung, etc
	Independent Developer	Mass number of independent developers
Openness	Open	Few libraries like WebKit
	Closed	iOS, MacOS
Fee	Entrance Fee	Annual payment
	Platform Fee	Indirectly through purchasing devices
Feedback Loop Facilitator	Rating & Reviewing	Star rating system, Reviewing
	Ranking	Lists of featured, popular, & new Apps
Knowledge Sharing	Documentation Framework	Apple Developer Documentation
	Q&A Forums	Apple Developer Forum, Apple Support Communities

(b) Exemplary Ecosystem

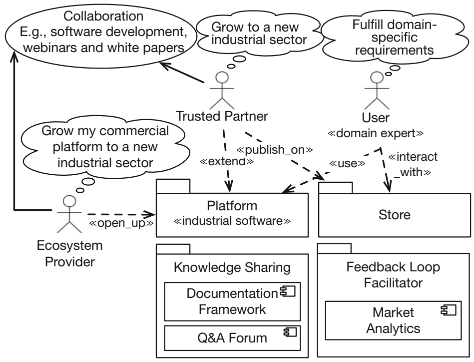
Fig. 1. Resale software ecosystem: an architectural solution to business scalability

Providers of these ecosystems are companies offering complex software solutions for industrial sectors. Supply chains, warehouse operation, and aerospace are examples of such industrial sectors. Moreover, users in the partner-based software ecosystems are also companies that we call *domain experts*, because they are companies with high knowledge and expertise in certain domains. These domain experts offer the solutions to further end-users.

(a) *Architectural landscape*: Table 1 provides details on how the existing partner-based software ecosystems realize the variants of the variability model. Furthermore, Fig. 2(a) sketches the architectural landscape of the partner-based software ecosystems. **Extender**: Trusted partners are the only extenders in the ecosystems. The ecosystem providers collaborate with the partners to develop joint solutions in new industrial sectors or for new application use cases. The collaborations come in form of software development, road map sessions, and webinars as well as marketing the joint solutions. To achieve different intensities of collaborations, the ecosystem providers often offer different tiers of partnerships like platinum or gold partners. Joint solutions resulting from a closer collaboration are eligible for certain promotions such as getting listed as featured products on the stores. **Openness**: Source code is closed in general. The platform APIs are monetized through an API management system, which requires authentication and defines a partner's access permissions to the code or APIs. **Fee**: To enter the ecosystems as a partner, candidates need to fulfill certain requirements, e.g., having certain annual revenue. Such requirements vary depending on ecosystem's policies. Prospective partners usually need to periodically pay to participate in the ecosystems or to use the platforms.

Feeback Loop Facilitator: In contrary to resale software ecosystems, rating features are not widely used in partner-based software ecosystems. But, the extensions are rather marketed in the ecosystems. Market analytics features inform the partners about quality of user experience. Examples of the market analytics are customer relationship management (CRM) features and repository mining. Thereby, the partners can stay connected with their customers and track their market growth. **Knowledge Sharing**: Documentation frameworks are provided as a part of partner programs. Partners are given access to partner portals. Due to protection of intellectual property, such portals are not accessible to anyone and require access permission. However, Q&A forums are accessible to the current and prospective users and partners.

Further Characteristics: The extensions on the stores are often labeled as "tested" or "validated". The partner-based ecosystems are highly commercial whereas 96% of the ecosystems in our list are for profit.



(a) Architectural Landscape

Variation Point	Variant	Citrix
Extender	Trusted Partner	Citrix strategic partners: Microsoft, Cisco, Google, etc. Citrix partner programs: Service providers, System Integrators, SaaS Referral Partner, etc.
Openness	Open	Partially only to the partners
	Closed	Platforms are generally closed
Fee	Entrance Fee	Periodic payment depending on partner type
	Platform Fee	On-demand pay-as-you-go
Feedback Loop Facilitator	Market Analytics	Citrix Marketing Concierge
Knowledge Sharing	Documentation Framework	Citrix Partner Central
	Q&A Forums	Citrix Discussions, Citrix User Group Community (CUGC)

(b) Exemplary Ecosystem

Fig. 2. Partner-based ecosystem: an architectural solution to strategically grow a commercial platform

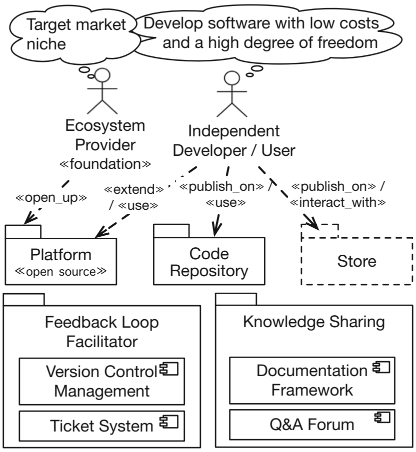
(b) *Exemplary real-world ecosystems*: Citrix is a provider of cloud computing services. Among others, the services include server and desktop virtualization and cloud computing. Figure 2(b) provides the variants of the Citrix ecosystem. Citrix establishes long-term collaborations with strategic partners like Microsoft, Cisco, and Google. The Citrix platforms are closed source. Platform fee is defined as a pay-as-you-go model that allows the partners to pay on-demand for the Citrix services. Citrix Ready Marketplace[6] is the store, where the partners' extensions are published. As a part of market analytics, Citrix mailing servers support communication between the users and partners. Another example is Citrix Marketing Concierge,

[6] citrixready.citrix.com/. Accessed April 2018.

which is to manage email campaigns, webinars, and roadshows. In addition, knowledge sharing is enabled by Citrix Product Documentation[7] and Partner Central[8]. Using the partner central, the partners access different partner programs. Furthermore, Citrix Discussions[9] and User Group Community[10] are the forums respectively used by the partners and users. Further examples of the partner-based software ecosystems are SAP, VMware Vsphere, Symantec, IFTTT, ExtendSim, and SolidWork.

3.3 OSS-Based Software Ecosystem

28% of ecosystems in our list have grown around open source software platforms. We call them *open source software-based* or shortly *OSS-based software ecosystems*. Ecosystem providers are in form of foundations or consortia. The members of such foundations are software companies or individuals, which unify to create an ecosystem around open source software. Mozilla and Eclipse foundations are examples of such providers.



(a) Architectural Landscape

Variation Point	Variant	Apache Cordova
Extender	Trusted Partner	Apache Foundation
	Independent Developer	Committers & contributors
Openness	Open	Apache Cordova, Plug-ins
Fee	Entrance Fee	Free of charge
	Platform Fee	Free of charge
Feedback Loop Facilitator	Ticket System	Atlassian Jira
	Version Control Management	Cordova plug-in versioning
Knowledge Sharing	Documentation Framework	Cordova Documentation Framework
	Q&A Forums	Public Q&A Forums

(b) Exemplary Ecosystem

Fig. 3. OSS-based ecosystem: an architectural solution to address market niche

(a) *Architectural landscape*: Table 1 summarizes the way that the OSS-based ecosystems realize the variants. Figure 3(a) illustrates the architectural landscape of OSS-based software ecosystems. **Extender**: Independent developers extend the platforms by directly accessing the source code. Such developers play both roles of extender and user, because the extensions are pieces

[7] docs.citrix.com/. Accessed April 2018.
[8] www.citrix.com/partnercentral/. Accessed April 2018.
[9] discussions.citrix.com/. Accessed April 2018.
[10] www.mycugc.org/. Accessed April 2018.

of code that are collaboratively developed by them. **Openness**: A direct access to the source code gives the developers a high degree of freedom. This mostly releases them from certain business and technical requirements imposed by the ecosystem providers. **Fee**: No entrance or platforms fee is usually demanded. In addition, in 64% of the ecosystems, the extensions are for free. In this situation, the developers might be non-commercially motivated and willing to extend an open source platform to fulfill domain-specific requirements [14]. Still the rest of ecosystems, i.e., 36%, generates revenue for their developers.

Feedback Loop Facilitator: A positive feedback loop between the developers, who commit to the code, and the ones, who reuse it, is created using version control management and ticket system features. A version control management like Apache Subversion supports forking, branching, and merging the code. Additionally, a ticket system like Jira[11] helps to track and communicate issues and bugs. **Knowledge Sharing**: Both Q&A forums and documentation frameworks are heavily used in the OSS-based software ecosystems. An example of Q&A forums is Stack Overflow[12] that facilitates questioning and answering on a wide range of topics in programming languages. An example for documentation frameworks is Markdown[13], which is a markup language with syntax formatting to create enhanced documents like wikis.

Further Characteristics: Extensions are published on a public code repository, which, in 78% of the cases, GitHub.com is used. In addition, some ecosystems provide online stores. We assume this happens when an ecosystem is mature and offers commercial and ready-to-use pieces of code. In addition, in OSS-based software ecosystems, ranks are generated implicitly once the developers bookmark or fork a project.

(b) *Exemplary real-world ecosystems*: Fig. 3(b) outlines the variants of the Apache Cordova ecosystem. Apache Cordova[14] is a framework to develop cross-platform mobile applications. Apache Software Foundation[15] is the ecosystem provider. Adobe, IBM, and Mozilla are the exemplary members. Developers are called committers and contributors. They develop applications using CSS3, HTML5, and JavaScript rather than relying on platform-specific APIs. Using Apache Cordova, the code is wrapped to be run on different platforms like Android and iOS. While the source code and the extensions reside on GitHub.com, a list of extensions including their metadata like supported platforms is on a store namely Cordova Plug-ins[16]. Moreover, a feedback loop between the developers is facilitated by the Cordova plug-in versioning as a part of the framework. Furthermore, a ticket

[11] www.atlassian.com/software/jira. Accessed April 2018.
[12] stackoverflow.com/. Accessed April 2018.
[13] daringfireball.net/projects/markdown/. Accessed April 2018.
[14] cordova.apache.org/. Accessed April 2018.
[15] www.apache.org/. Accessed April 2018.
[16] cordova.apache.org/plugins/. Accessed April 2018.

system is established by using Jira. As a part of knowledge sharing, a documentation framework[17] provides information on the API reference. The ecosystem does not own any Q&A forum, but many related threads can be found in public forums like Microsoft developer network (MSDN) and Stack Overflow. Further examples of OSS-based software ecosystems are Cloud Foundry, Ubuntu, Odoo, OpenFOAM, CTAN: Packages, Mozilla, Zotero, LibreOffice, and Eclipse.

4 Discussion

In this section, we further discuss the design options with respect to their overall contribution. The arrangement of the actors and software features contribute to an outstanding business goal in each architectural landscape. Considering the high number of extenders, users, and extensions on the stores, the resale software ecosystem can be seen as an architectural solution to achieve **business scalability**. Rating, ranking, and billing features support to establish a market between the extenders and users, while making them highly independent of the ecosystem providers. Furthermore, using the partner-based ecosystem design option, ecosystem providers monetize the platforms and extensions by establishing different degrees of collaborations with trusted partners and including the market analytics features to promote the joint solutions. This ultimately addresses the **commerciality** of an ecosystem. Finally, the OSS-based software ecosystem design option degrades the protection of intellectual property by opening the source code to the developers. However, due to the high availability of tools and resources for free and open-source software (FOSS) community, the **cost** of opening a platform decreses [3]. Moreover, as the degree of openness increases, an ecosystem suceeds to address **innovation** by providing extensions for market niche [15]. Furthermore, from the modeling perspective, we notice that the openness and fee variation points can not directly be expressed by UML.

5 Conclusion and Future Work

Many modern software companies create store-oriented ecosystems of third-party providers and users on top of their platforms; online stores serve as distribution channels for third-party developments. This architectural approach has been applied widely; however, the diversity of the existing ecosystem designs hinders prospective ecosystem providers to gain a sound overview of the existing designs and to understand how to design a store-oriented software ecosystem that fulfills certain business decisions.

In this paper, we provide empirical evidence that three designs options of these ecosystems are frequently applied in practice: (1) Resale software ecosystem, (2) Partner-based ecosystem, and (3) OSS-based ecosystem. We provide

[17] cordova.apache.org/docs. Accessed April 2018.

insight into their business decisions and the overall contribution in terms of business goals. This knowledge helps future ecosystem providers to decide on when to apply any of these design options according to their needs. In the future, practical effectiveness and possible combinations of the design options can be further studied by architects on real projects while including new business decisions in the design options.

References

1. Bosch, J.: From software product lines to software ecosystems. In: International Software Product Line Conference, pp. 111–119. CMU (2009)
2. Bosch, J., Bosch-Sijtsema, P.: From integration to composition: on the impact of software product lines, global development and ecosystems. J. Syst. Softw. **83**(1), 67–76 (2010)
3. Manikas, K., Hansen, K.M.: Software ecosystems-a systematic literature review. J. Syst. Softw. **86**(5), 1294–1306 (2013)
4. Sadi, M.H., Yu, E.: Modeling and analyzing openness trade-offs in software platforms: a goal-oriented approach. In: Grünbacher, P., Perini, A. (eds.) REFSQ 2017. LNCS, vol. 10153, pp. 33–49. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-54045-0_3
5. Berger, T., Pfeiffer, R.-H., Tartler, R., Dienst, S., Czarnecki, K., Wasowski, A., She, S.: Variability mechanisms in software ecosystems. Inf. Softw. Technol. **56**(11), 1520–1535 (2014)
6. Jazayeri, B., Zimmermann, O., Engels, G., Kundisch, D.: A variability model for store-oriented software ecosystems: an enterprise perspective. In: Maximilien, M., Vallecillo, A., Wang, J., Oriol, M. (eds.) ICSOC 2017. LNCS, vol. 10601, pp. 573–588. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69035-3_42
7. Van Angeren, J., Kabbedijk, J., Jansen, S., Popp, K.M.: A survey of associate models used within large software ecosystems In: International Workshop on Software Ecosystems, pp. 27–39. CEUR-WS (2011)
8. Jazayeri, B., Platenius, M.C., Engels, G., Kundisch, D.: Features of IT service markets: a systematic literature review. In: Sheng, Q.Z., Stroulia, E., Tata, S., Bhiri, S. (eds.) ICSOC 2016. LNCS, vol. 9936, pp. 301–316. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46295-0_19
9. Zimmermann, O., Wegmann, L., Koziolek, H., Goldschmidt, T.: Architectural decision guidance across projects-problem space modeling, decision backlog management and cloud computing knowledge. In: Working IEEE/IFIP Conference on Software Architecture, pp. 85–94. IEEE (2015)
10. Rozanski, N., Woods, E.: Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives. Addison-Wesley, Boston (2012)
11. Holzer, A., Ondrus, J.: Mobile application market: a developer's perspective. Telematics Inf. **28**(1), 22–31 (2011)
12. Grover, V., Kohli, R.: Cocreating IT value: new capabilities and metrics for multifirm environments. MIS Q. **36**, 225–232 (2012)
13. Dataset. Technical report (2018). https://www.overleaf.com/read/njqzqhsmvctk
14. Hanssen, G.K.: A longitudinal case study of an emerging software ecosystem: implications for practice and theory. J. Syst. Softw. **85**(7), 1455–1466 (2012)
15. Gawer, A., Cusumano, M.A.: Industry platforms and ecosystem innovation. J. Prod. Innov. Manage. **31**(3), 417–433 (2014)

Business Process Variability and Public Values

Boris Shishkov[1,3](✉) and Jan Mendling[2]

[1] Institute of Mathematics and Informatics,
Bulgarian Academy of Sciences, Sofia, Bulgaria
[2] Institute for Information Business,
Vienna University of Economics and Business, Vienna, Austria
jan.mendling@wu.ac.at
[3] Institute IICREST, Sofia, Bulgaria
b.b.shishkov@iicrest.org

Abstract. A business process is a structure of inter-related activities that are executed in order to achieve a specific business objective. Organizations often maintain multiple variants of a given business process because of changing conditions, different regulations in different countries, or other contextual factors. We aim at specifying the relationship between a generic business process and its different variants, taking the perspective of public values, such as privacy, accountability, and transparency. The business process variants in turn may be a basis for software specifications – in this, business processes would be bridging between societal demands (possibly concerning public values) and the corresponding technical (software) functionalities. Our contribution is featuring a meta-model that describes business processes on a value-independent level; they can be extended towards value-specific business process variants that can be related in turn to software architectures. We reflect this in proposed value operationalization guidelines, using concepts from business process design as a basis; those guidelines assume coming firstly through technology-independent artefacts and secondly – through technology-specific artefacts, to arrive at software specifications that are adequate with regard to public-values-related demands.

Keywords: Business process variability · Public values · Software design

1 Introduction

A **business process** is a *structure of inter-related activities* that are executed in order to achieve a specific *business objective* [29]. Activities of business processes are increasingly supported by **ICT** (**Information and Communication Technology**), no matter if they are *intellectual* (e.g. business calculations) or *manual* (e.g. warehouse picking), *routine* (e.g. standard calculations) or *non-routine* (e.g. brain scan assessment) [7]. Against this background, there is an increasing awareness of the importance of **public values** ("*values*", for short), such as *privacy*, *accountability*, and *transparency*, especially as it concerns business processes [8]. The consideration of such *values* in the development of **software products** is for these reasons an important concern in recent debates on *ethical design* [3, 11]. If *business process*

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 401–411, 2018.
https://doi.org/10.1007/978-3-319-94214-8_31

design and *software specification* must take *values* explicitly into account, it is an open question how to adequately **operationalize** values, “translating” them into *value-sensitive software artefacts*. A key problem in this regard is that *values are abstract, non-functional, and relevant to social sciences* while *software specifications are technical, functional, and relevant to computing paradigms*. So far, **Requirements Engineering** (**RE**) has been addressing the *domain-imposed* and *user-defined requirements* with regard to the software system-to-be, touching upon *functional* and *non-functional concerns*. Further, a non-functional requirement on externally observable properties of a system may lead to functional requirements on the internal structure of the system [1]. Therefore, considering values as non-functional requirements could seem “tempting” in this regard. Nevertheless, since non-functional requirements are usually *technical* (for example: *recoverability*, *response time*, and so on) and the requirements engineering experience is in translating non-functional requirements to functional requirements, *it is still a question how to “translate” values to functional requirements*. Furthermore, **Value-Sensitive Design** (**VSD**) is about *weaving values in the design* of (technical) systems. For this, the role of values is advocated among stakeholders, such that there is such an awareness since the very early stage of a (software) project [8]. Nevertheless, VSD is abstract whereas engineering is concrete and hence we argue that VSD does not have “solid” bridges to engineering.

In addressing the above problem, we are inspired by the observation that *business processes* are *de facto* “bridging” between *societal public demands* and the corresponding *technical* (*software*) *functionalities*. That is because business processes are essentially *human-driven* and at the same time it is through business processes that people and enterprises *utilize ICT* [29]. Thus, it is expected that *business processes could play a crucial role in closing the gap between values and software functionalities*, as much as they are considered as a means of implementing strategies [26]. Our contribution is featuring a **formal meta-model that describes business processes on a value-independent level, which can be extended towards value-specific business process variants**. The underlying idea builds on the observation that organizations often maintain multiple **variants** of a given business process because of *changing conditions*, *different regulations in different countries*, or *other contextual factors* [30]. Correspondingly, our meta-model specifies the *relationship between a generic business process (featuring the base features of the business process) and its different variants*, by the help of *values*; further, if information systems are to be developed, **we consider business processes as a key foundation with regard to the specification of software**. We provide corresponding guidelines concerning the **operationalization of values in the context of software specification,** using business process design concepts as a basis.

Further in the paper: In Sect. 2, we consider *essential concepts* discussed above: *value*, *requirements*, *business process variability*. Our proposed *meta-model* and *guidelines* are presented in Sect. 3. In Sect. 4, we relate our work to *relevant streams of research*: *feature modeling*, *aspect-oriented design*, and *configurable process models*, such that we position further our contribution, emphasizing on its useful features. We conclude the paper in Sect. 5.

2 Background

Referring to the notions considered in the Introduction, we will firstly elaborate on *values and their relation to requirements* (Subsect. 2.1) and secondly – *on business process variability* (Subsect. 2.2).

2.1 Values and Requirements

We argue that essentially, **values** are desires of the general public (or public institutions/organizations that claim to represent the general public), that are about properties considered societally valuable, such as *respecting the privacy of citizens* or *prohibiting polluting activities*. Even though values are to be broadly accepted (that is why they are **public**), they may concern individuals (for example: considering privacy) [32]. Hence, put broadly, values concern the *societal expectations with regard to the way services should be delivered*. Furthermore, we argue that "values" become actual "values" only if *resources are committed for this* (for example, a government finds privacy so important that time and money are invested to regulate and enforce privacy); otherwise things only remain at the level of "hollow" abstract desires (such as for example: "Make the World a better place") that are stated but are never effectively realized.

Since most current *technical systems* are essentially *goal-driven* [25], it is interesting to analyze values vs. goals conceptually, acknowledging that *enterprises can adopt values as part of their goals*. This is often done *under public pressure or through legislation*, as values may *conflict* with enterprise goals, such as *profitability* or *cost-saving* [14]. Values may also *differ from the user goals* because often values concern *third parties* and the user would not care about third parties as long as the user demands are fulfilled. Hence, values may be reflected in goals even though those would usually be *societal* (third-party) *goals* and not *enterprise goals* or *user goals*.

We propose a value categorization (Fig. 1) according to which values are desires relevant to particular *persons* (either *physical* or *legal* persons) or their *societal environment*. As such, values may either concern a particular individual or society altogether. Hence, we can distinguish between *individual values* (for example, privacy) and *societal values* (for example, sustainability). We also distinguish between *basic values* (for example, love), *moral values* (for example, justice), *physical values* (for example, nature), and *virtual values* (for example, intelligence). This categorization is inspired by [24, 28, 34].

Further, values may be different with different *stakeholders* and might differ among *countries* and *cultures*, and eventually change over *time*. Next to that, agreeing on a particular value would not necessarily mean agreeing on its *operationalization* [9]. There may be *different operationalizations and implementations of the same value*. Finally, different values might be in conflict between each other (referred to as *value tensions*), meaning that fulfilling one value and fulfilling another value would not be possible at the same time. Nevertheless, resolving value tensions might resort to ideas for handling goal conflicts [33] and is therefore outside the scope of this paper.

Since *weaving values in software functionalities* could only be materialized through considering the corresponding *requirements specification*, it is important to discuss the relation between the value concept and the requirement concept. As seen from the current discussion (see above, see the Introduction), *values* are *desires* or *goals*, not *requirements*. Values are abstract and not directly related to an enterprise or software system [8], as opposed to requirements [1]. Moreover, values are construct by and for society and not by and for the enterprise domain in which a specific system will be used. Those domains may overlap but are not the same. Values that are adopted as goals by an enterprise would thus impact the requirements on a system that the enterprise wants to introduce in order to realize its goals. Hence, *the impact of values cannot be limited to non-functional requirements*. It is therefore considered important to clearly distinguish *values* from *requirements* and acknowledge the limitations of requirements engineering with regard to the development of value-sensitive software systems.

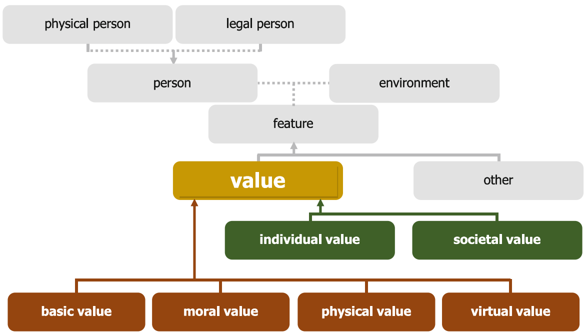


Fig. 1. Categorizing values

At the same time, it is important to align those concepts and position requirements accordingly. According to [2], *RE* is partially about achieving a *coherent description of the causal relationships between the phenomena in a particular domain*. This concerns the *domain-imposed requirements*. They are important because whatever we design, our designed artefact would be functioning in its environment or domain [29]. For this reason, the *regulations* and *rules* governing that domain should also have an impact on the designed artefact and its behavior [20]. Next to that, *RE* is also about *what the user wants* the designed artefact to do – this concerns the *user-defined requirements* [29]. Obviously, it is most important that the user-defined requirements are consistent with the domain-imposed requirements – otherwise, the designed artefact would be irrelevant with regard to its environment. Still, the user-defined requirements go beyond that and bring to the design the particular user demands. As for *values*, we argue that they are relevant in two ways. Firstly, they are relevant to the domain-imposed requirements because of the *consideration of societal issues that concern the domain*. Secondly, they are also relevant to the user-defined requirements because the *delivery of a value-sensitive service to the user* would differ from what the service delivery would have been with no consideration of values.

Therefore, it is not trivial to simply impose values on top of a traditional software development process that assumes analysis, requirements specification, design, implementation, and testing [15]. For this reason, we propose a *meta-model* and corresponding *design guidelines*.

2.2 Business Process Variability

The idea of *adjusting the behavior* according to pre-configured parameters has been included in various proposals for configurable business process models [27]. Configurable business process models are business process models that make various *alternative design choices* explicit by the help of configurable elements such as *activities* and *gateways*. These alternatives have to be selected *at design time* in order to arrive at a configured business process model. This configured business process model is a specific **`variant`** of the *generic business process model* and it can then be used for the *implementation of software systems*.

The general idea of configurable business process models fits well the overall ambition of *value-driven modeling*. A specific approach to configurable business process models is: a *questionnaire-driven configuration* [19], *aggregated business process models* [23] and *configurable multi-perspective process models* [17]. Those have in common that *the configuration can be tied to specific configuration parameters*. In this way, several configurable elements can be configured together. In our view, this concept is highly suitable to address the idea of value-driven modeling since *values can be used to configure entire parts of the business process model* in order to respond to corresponding demands.

3 Proposal

In this section, we propose a conceptualization and design guidelines of value-driven modeling.

3.1 Conceptualization

In order to *close the gap between abstract values and software specifications* in the context of enterprise systems and business processes, we make several **`assumptions`**:

- When considering an *organization* and a *software system that is supporting it*, we assume that *there are values that need to be reflected in the software design* (otherwise there would be no relevance to the topic of the current paper), and:
  - It is assumed that those values are *known*;
  - It is assumed that they *have to be weaved into the system design* and it is outside the paper's scope discussing why is it beneficial for society that this happens.
- We assume that all relevant values are identified *at design time*.
- We assume that for any value, there is a known *corresponding business process variant*; then, selecting it is expected to lead to the value fulfillment.

Based on those *assumptions* and the above *elaborations*, we introduce our main modeling concepts (meta-model) using the notations of UML - Class Diagram [31]. Figure 2 shows that there are *five key concepts* (**society**, **value**, **business process variant**, **software architecture**, and **information system**) represented as **classes** (*named boxes*), complemented by corresponding **associations** (*lines*).

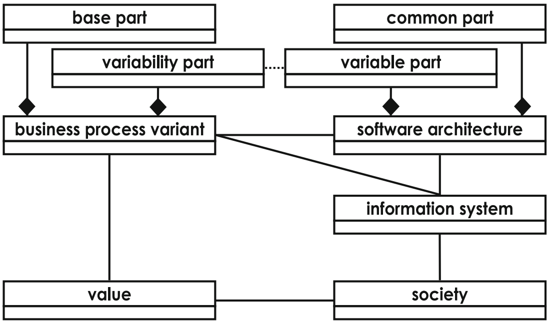


Fig. 2. Concepts

Values are *defined in the society* as represented by the association line between "value" and "society" and a *value* has its *corresponding business process variant* as defined by the association line between "value" and "business process variant" (this is according to the assumptions). In turn, a *business process variant* consists of one or more *base parts* and one or more *variability parts* as defined by the composition signs in the figure. Further, it is the *business process variant* that essentially concerns the *architecture* of the supportive software because the software would have to support partially or completely the business process variant, which is specified by the association line between "business process variant" and "software architecture". A *software architecture* in turn consists of one or more *common parts* and one or more *variable parts*, represented by the composition signs in the figure, and it is the *business process variability parts* that guide the design of corresponding *software architecture variable parts* (this is indicated by the dotted line in the figure). Finally, this results in an *information system* that has both *human* and *technical* aspects and is therefore related not only to its *underlying software architecture* but also the *supported business process variant*. This is indicated by the corresponding association lines. It is therefore the *information system* that would eventually deliver services to its customers that are value-sensitive, thus relevant in terms of values to the society.

In this way, **values are reflected in business process variants** that in turn **shape the software architecture** that is underlying with regard to the **information system** that guarantees those values for the society.

3.2 Guidelines

As suggested already, our proposed design guidelines are about the **value-driven modeling of business process variants** and a further **mapping towards software specifications**. This is represented in Fig. 3 using the notations of UML - Activity Diagram [31]. As the figure suggests, we have *two parallel processes*, a **value-independent** one and a **value-specific** one. The former is about the *base part modeling* (reflecting invariant business process behavior) and its mapping towards corresponding design that is featuring *core parts of the software architecture*; the latter is about the project-driven *consideration of a particular value* that leads to the modeling of corresponding *business process variability issues* that are in turn mapped towards corresponding design that is featuring *variable parts of the software architecture*. Further, all *business-process-modeling-related tasks* are **technology-independent** (see the brick-backgrounded area in the figure) while all *software-design-related tasks* are **technology-specific** (see the dotted area in the figure). Finally, the technology-independent activities are to be mutually *in synch*, just as the technology-specific activities (this is indicated by the dashed lines in the figure).

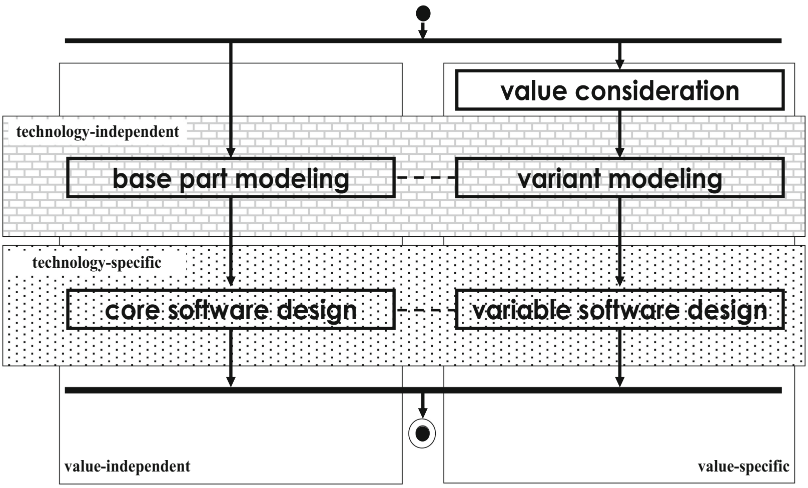


Fig. 3. Design guidelines

That is how we vision the **bridging role of business process variants** with regard to *values that need to be operationalized* and the corresponding *software specifications*.

Nevertheless, those guidelines need *further* (technical) *elaboration* and a follow up *validation*. This is left beyond the scope of the current paper and is planned as future research. Still, we have identified and studied *relevant research streams* (see the following section) that essentially ground our proposed guidelines (and the meta-model), emphasizing on their useful features.

4 Applicability – Related Work

The work presented in this paper is related to the following streams of research: **feature modeling**, **aspect-oriented design**, **configurable process models**.

Feature Modeling (FM) is concerned with the specification of *commonalities and variations of a software product*, e.g. to support the development of *software product lines* [6]. In *FM*, **variants are designed top-down**. Various methods exist for facilitating FM with *Feature-Oriented Domain Analysis* (*FODA*) being the most prominent one [12]. **Our work not only shares the emphasis on variations with FM but it also offers a novel perspective, by anchoring the choice of different variants on the abstract level of the business model**.

Aspect Orientation (AO) approaches variation from the perspective of *integrating additional functionality where needed* [13]. This is considered useful for adding **crosscutting concerns** to a software system such as to address *security requirements*, *response time*, *recoverability issues*, and so on. Specific operations can be used to *weave such functionalities into the original software core*. *AO* has been further integrated with *process modeling languages* like *BPMN* [4] in order to address *crosscutting business concerns* such as *compliance*, for example. **In our work, we envision the weaving (together) of separate building blocks, which partially builds on ideas of AO**.

In the research area of Configurable Process Models, several *languages* have been proposed to support the *specification of variability* [16], with *C-EPCs* [22], *Pesoa for BPMN* [21], and *Provop* [10], superimposed variants for *UML Activity Diagrams* [5] being among the most prominent ones. Those languages essentially share the idea of *full specification of variation at design time*. **Our work builds upon the idea that variants can be specified and selected on an abstract level**. This idea has been *instantiated* with *C-EPC* by the help of a *questionnaire with closed questions*, such that *answer options* can be translated to corresponding *variants* [18].

5 Conclusion

Considering software specifications that are based on business process models, we have addressed in this paper business processes in their role of bridging between societal public demands and the corresponding technical (software) functionalities, touching upon public values (desires of the general public). In particular, we have considered a value-driven specification of business process variants as a way of closing the gap between abstract public values and required corresponding value (software) operationalizations. This assumes not only identifying the business process variants but also reflecting them in turn in corresponding software specifications – this all prepared at design time. Hence, when considering a generic (value-independent) business process and a value-related demand, a business process variant may be specified (it is already value-specific but technology-independent); in turn, the business process variant could be reflected in a technology-specific software specification. It is expected that such a design approach would increase effectiveness and efficiency when it is about the development of value-sensitive software systems. We have proposed a meta-model and design guidelines accordingly, and we have related our work to relevant streams of research. For this reason, the current paper is considered to have both analytical and propositional value. Nevertheless, the lack of sufficient elaboration and also the lack of validation frame our work as research in progress. We plan as future research to elaborate further on our proposal and validate its applicability by means of case studies.

Acknowledgement. This work is supported by the TU Delft - *Delft Pilot* project and we would like to thank *Jeroen van den Hoven* for his support and guidance.

References

1. Akkermans, H., Gordijn, J.: What is this science called requirements engineering? In: Proceedings of RE 2006 - 14th IEEE International Requirements Engineering Conference, Minneapolis/St. Paul, USA, pp. 273–278 (2006)
2. Anish, P.R., Daneva, M., Cleland-Huang, J., Wieringa, R.J., Ghaisas, S.: What you ask is what you get: understanding architecturally significant functional requirements. In: Proceedings of RE 2015 - 23rd International Requirements Engineering Conference, Ottawa, ON, pp. 86–95 (2015)
3. Baldini, G., Botterman, M., Neisse, R.: Ethical design in the internet of things. In: Sci Eng Ethics (2016)
4. Charfi, A., Müller, H., Mezini, M.: Aspect-oriented business process modeling with AO4BPMN. In: Kühne, T., Selic, B., Gervais, M.-P., Terrier, F. (eds.) ECMFA 2010. LNCS, vol. 6138, pp. 48–61. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13595-8_6
5. Czarnecki, K., Antkiewicz, M.: Mapping features to models: a template approach based on superimposed variants. In: Glück, R., Lowry, M. (eds.) GPCE 2005. LNCS, vol. 3676, pp. 422–437. Springer, Heidelberg (2005). https://doi.org/10.1007/11561347_28
6. Czarnecki, K., Kim, C.H.P.: Cardinality-based feature modeling and constraints: a progress report. In: Proceedings of International Workshop on Software Factories, San Diego, CA, pp. 16–20. ACM (2005)
7. Frey, C.B., Osborne, M.A.: The future of employment: how susceptible are jobs to computerisation? In: Technological Forecasting and Social Change, vol. 114 (2017)
8. Friedman, B., Hendry, D.G., Borning, A.: A survey of value sensitive design methods. In: A Survey of Value Sensitive Design Methods, 1, Now Foundations and Trends, 76 p. (2017)
9. Grenholm, C.-H., Kamergrauzis, N. (eds.): Sustainable Development and Global Ethics. Acta Universitatis Upsaliensis, Uppsala (2007)
10. Hallerbach, A., Bauer, T., Reichert, M.: Capturing variability in business process models: the Provop approach. J. Softw. Maint. Evol. **22**, 6–7, 519–546 (2010)
11. Heersmink, R., Hovenvan den, J., Eckvan, N.J., Bergvan den, J.: Bibliometric mapping of computer and information ethics. Ethics Inf. Technol. **13**, 241–249 (2012)
12. Kang, K.C., Cohen, S.G., Hess, J.A., Novak, W.E., Peterson, A.S.: Feature-Oriented Domain Analysis (FODA) Feasibility Study (No. CMU/SEI-90-TR-21). Carnegie-Mellon Univ. Pittsburgh (1990)

13. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.-M., Irwin, J.: Aspect-oriented programming. In: Akşit, M., Matsuoka, S. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0053381
14. Kinsella, W.J.: From big science to postmodern science: technology-intensive research in an era of competing public values. In: Proceedings of International Symposium on Technology and Society Technical Expertise and Public Decisions, Princeton, NJ, pp. 15–24. IEEE (1996)
15. Kruchten, P.: The Rational Unified Process, an Introduction. Addison-Wesley, Boston (2003)
16. La Rosa, M., Van Der Aalst, W.M., Dumas, M., Milani, F.P.: Business process variability modeling: a survey. ACM Comp. Surv. (CSUR) **50**(1), 2 (2017)
17. La Rosa, M., Dumas, M., Ter Hofstede, A.H.M., Mendling, J.: Configurable multi-perspective business process models. Inf. Syst. **36**(2), 313–340 (2011)
18. La Rosa, M., Van der Aalst, W.M., Dumas, M., Ter Hofstede, A.H.M.: Questionnaire-based variability modeling for system configuration. Softw. Syst. Model. **8**(2), 251–274 (2009)
19. La Rosa, M., Lux, J., Seidel, S., Dumas, M., ter Hofstede, A.H.M.: Questionnaire-driven configuration of reference process models. In: Krogstie, J., Opdahl, A., Sindre, G. (eds.) CAiSE 2007. LNCS, vol. 4495, pp. 424–438. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72988-4_30
20. Liu, K.: Semiotics in Information Systems Engineering. Cambridge University Press, Cambridge (2000)
21. Puhlmann, F., Schnieders, A., Weiland, J., Weske, M.: Variability mechanisms for process models. PESOA-Report TR, 17, pp. 10–61 (2005)
22. Recker, J., Rosemann, M., van der Aalst, W., Mendling, J.: On the syntax of reference model configuration – transforming the C-EPC into lawful EPC models. In: Bussler, C.J., Haller, A. (eds.) BPM 2005. LNCS, vol. 3812, pp. 497–511. Springer, Heidelberg (2006). https://doi.org/10.1007/11678564_46
23. Reijers, H.A., Mans, R.S., Van der Toorn, R.A.: Improved model management with aggregated business process models. Data Knowl. Eng. **68**(2), 221–243 (2009)
24. Reynaers, A.-M.: It takes two to tangle: public-private partnerships and their impact on public values. Ph.D. thesis, Universidad Autónoma de Madrid (2014)
25. Rong, G., Liu, X., Gu, S., Shao, D.: A goal-driven framework in support of knowledge management. In: Proceedings of ASPEC 2017 – 24th Asia-Pacific Software Engineering Conference, Nanjing, pp. 289–297 (2017)
26. Rosemann, M., vom Brocke, J.: The six core elements of business process management. In: vom Brocke, J., Rosemann, M. (eds.) Handbook on Business Process Management 1: International Handbooks on Information Systems. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-642-00416-2_5
27. Rosemann, M., Van der Aalst, W.M.P.: A configurable reference modelling language. Inf. Syst. **32**(1), 1–23 (2007)
28. Schwartz, S.H.: An overview of the schwartz theory of basic values. Online Read. Psychol. Cult. **2**(1), 11 (2012)
29. Shishkov, B.: Enterprise Information Systems, a Modeling Approach. IICREST, Sofia (2017)
30. Sinnhofer, A.D., Pühringer, P., Oppermann, F.J., Potzmader, K., Orthacker, C., Steger, C., Kreiner, C.: Combining business process variability and software variability using traceable links. In: Shishkov, B. (ed.) BMSD 2017. LNBIP, vol. 309, pp. 67–86. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78428-1_4
31. UML: The website of the Unified Modeling Language (2018). http://www.uml.org

32. Van den Hoven, J.: Value sensitive design and responsible innovation. In: Owen, R., Bessant, J., Heintz, M. (eds.) Responsible Innovation: Managing the Responsible Emergence of Science and Innovation in Society. Wiley, Hoboken (2013)
33. Van Lamsweerde, A., Darimont, R., Letier, E.: Managing conflicts in goal-driven requirements engineering. IEEE Trans. Softw. Eng. **24**(11), 908–926 (1998)
34. Zhang, J.: A brief study of the hierarchy value thought of the pre-qin confucianism. In: Li, D. (ed.) Values of Our Times. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38259-8_17

Composite Public Values and Software Specifications

Magdalena Garvanova[1](✉), Boris Shishkov[2,4], and Marijn Janssen[3]

[1] Faculty of Information Sciences,
University of Library Studies and Information Technologies, Sofia, Bulgaria
m.garvanova@unibit.bg
[2] Institute of Mathematics and Informatics,
Bulgarian Academy of Sciences, Sofia, Bulgaria
[3] Faculty of Technology, Policy and Management,
Delft University of Technology, Delft, The Netherlands
M.F.W.H.A.Janssen@tudelft.nl
[4] Institute IICREST, Sofia, Bulgaria
b.b.shishkov@iicrest.org

Abstract. Public values are desires of the general public, that are about properties considered societally valuable, such as respecting the privacy of citizens or prohibiting polluting activities. "Translating" public values into functional solutions is thus an actual challenge. Even though Value-Sensitive Design (VSD) is about weaving public values in the design of (technical) systems, it stays insufficiently concrete as it concerns the alignment between abstract public values and technical (software) solutions. Still, VSD indirectly inspires ideas in that direction as for example the idea to consider business process variants for achieving such an alignment. Nevertheless, this is all about "atomic" public values (encapsulating only one particular behavioral goal) while one would often face public values that are "composite" in the sense that they reflect a particular human attitude rather than just a desired behavioral goal. In the current paper, we propose a value decomposition approach that allows for operationalizing composite public values. We also present experimental results featuring data analytics using self-administrated surveys.

Keywords: Public values · Software specification · Data analytics

1 Introduction

`Public values` ("values", for short) are *desires of the general public*, that are about *properties considered societally valuable*, such as respecting the privacy of citizens or prohibiting polluting activities [10]. "Translating" *values* into *functional solutions* is thus an actual challenge. Even though *Value-Sensitive Design* (*VSD*) is about *weaving values in the design* of (technical) systems [3], we argue that it stays insufficiently concrete as it concerns the *alignment between abstract values and technical (software)*

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 412–420, 2018.
https://doi.org/10.1007/978-3-319-94214-8_32

solutions. Still, VSD indirectly inspires ideas in that direction as for example the idea of Shishkov & Mendling to consider *business process variants* as a "bridge" in achieving such an alignment [10].

Nevertheless, this is all about **atomic values** (encapsulating only one particular behavioral goal) while one would often face values that are **composite** in the sense that they reflect a particular human attitude rather than just a desired behavioral goal. An example for the former is the desire to realize something in a *privacy-sensitive* way [8]; an example for the latter is the desire to achieve *egalitarianism* [2].

Whereas, *atomic values* are weave-able in the (software) design, as studied by Shishkov et al. [8–10], they are claimed to be not very "instrumental" as it concerns social feedback. In our view that is because most of those values (such as *privacy*, *transparency*, *accountability*, and so on) are to be considered in a particular context [8, 9, 11]. Then people consider them differently depending on the context. For example: USUALLY, privacy is desired but when HUNTING TERRORISTS, it might be acceptable by many people that authorities compromise their privacy. Therefore, studying in general what somebody's attitude is towards privacy (for example), could be of limited use. For this reason, we argue that *atomic values could only be adequately operationalized if this concerns context-aware systems* [9].

Composite values, in contrast, are not so easy to weave in the design (because they are even more abstract than atomic values) but it is easier to capture public opinion concerning them through surveys (or other analyses), as it is claimed by Veenhoven & Kalmijn [13] – they argue that many issues that concern composite public values (such as *egalitarianism*, *utilitarianism*, *autonomy*, *embeddedness*, and so on) *can be measured using surveys*.

Therefore:

- *Atomic values* can be operationalized but it is not easy justifying this as a public demand;
- It is not straightforward operationalizing *composite values* but the need for this can be "measured".

We propose a **value decomposition approach** that allows for operationalizing *composite values*. We also present **experimental results** featuring *data analytics using self-administrated surveys*.

Even though the **value concept** (assuming *atomic values*) is crucial with regard to this work, we are not presenting and discussing ATOMIC *values* in the current paper because this is done in other papers from the current proceedings – hence, we refer readers to [9, 10].

The remaining of the paper is organized as follows: In Sect. 2, we briefly introduce and discuss composite values, emphasizing on their societal relevance. In Sect. 3, we present our value decomposition approach that is supposed to be helpful to designers in their operationalizing also composite values if necessary. In Sect. 4 we present relevant experimental results. Finally, in Sect. 5, we conclude the paper.

2 Composite Values

In considering *composite values* from a social sciences perspective, we refer to Shalom Schwartz according to whom every culture can be described by 7 universal value orientations (categories), namely: *embeddedness*, *intellectual autonomy*, *affective autonomy*, *hierarchy*, *egalitarianism*, *harmony*, and *mastery* [6]:

- Embeddedness ("EMB", for short) focuses on maintaining the status quo and limiting the actions and inclinations that can disrupt the solidarity of the group or the imposed traditional order;
- Intellectual autonomy ("AUT", for short) expresses the aspirations of individuals to pursue their own ideas and independent intellectual purposes;
- Affective autonomy (AUT) expresses the desire of individuals to acquire affective positive experience;
- Hierarchy ("HIE", for short) emphasizes the legitimacy of the unequal distribution of power, roles, and resources;
- Egalitarianism ("EGA", for short) reveals the transcendence of individual interests in favor of voluntary commitment and concern for the welfare of others;
- Harmony ("HAR", for short) discloses the unity with the environment;
- Mastery ("MAS", for short) gives priority to active self-assertion and control of the social and natural environment.

EMB, EGA, and HAR are hence *collectively-oriented* values while AUT, HIE, and MAS are *individually-oriented* ones. Further, "tensions" are possible between two composite values if they cannot be fulfilled simultaneously, for example:

- EMB vs. AUT (the undifferentiated versus the differentiated from the group individual);
- HIE vs. EGA (inequality versus equality);
- MAS vs. HAR (control and change versus adaptation to the environment).

3 Concepts and Approach

Referring to the discussion carried out in the previous two sections, and considering two main concepts, namely "atomic value" and "composite value", we arrive at a conceptual view, as illustrated in Fig. 1, using the notations of UML - Class Diagram [12]: those two concepts are represented as two main value types. As the figure indicates, a *composite value* is composed of one or more *atomic values*. Further, the figure suggests, it is not straightforward decomposing a *composite value* in terms of corresponding *atomic values* (see the question mark in the figure) and a limitation of the current paper is that we have not explicitly tackled this issue leaving it for further research.

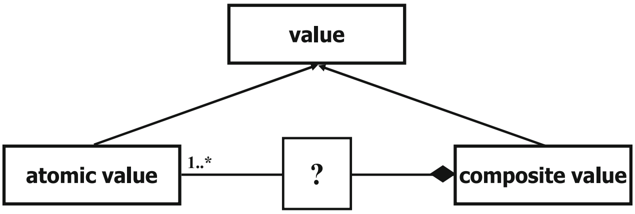


Fig. 1. Atomic values and composite values – conceptual view

Still, it is our **assumption** that *such a decomposition is possible*, acknowledging nevertheless that such complex social categories are not straightforwardly implementable – how, if we take the example of *egalitarianism*, an information system would establish that all people are equal and have equal rights and opportunities? We argue that an information system could "enforce" a number or relevant atomic values, such as *transparency*, *justice*, and *accountability* whose implementation **in combination** would in turn contribute to achieving *egalitarianism*. Hence, in this particular case, one would intuitively decompose *egalitarianism* into *transparency* + *justice* + *accountability* but maybe this would be more difficult for other *composite values*.

Thus, we assume that this is possible even though we have not yet delivered exhaustive *justification* that any *composite value* can be decomposed in terms of corresponding *atomic values*.

Anyway, the above assumption and the conceptual positioning (see Fig. 1) are already useful, inspiring the derivation of a **general approach** envisioning composite values and the way to consider them in support of system design – see Fig. 2; we have used the notations of the UML - Activity Diagram [12], keeping the approach also consistent with the **SDBC approach** [7].

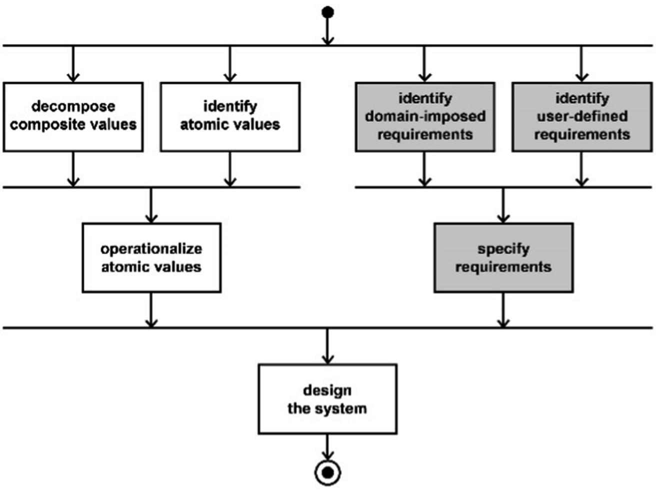


Fig. 2. Approach elaboration

As the figure suggests, it is not only the **requirements specification** [1] but also the **operationalization of atomic values** [9], that are needed as input for the (software) system design – we refer to the conclusion of Shishkov & Mendling that it is "important to clearly distinguish values from requirements and acknowledge the limitations of requirements engineering with regard to the development of value-sensitive software systems" [10]. For this reason we do not consider the operationalization of values as part of the requirements specification, as seen from the figure. Both *domain-imposed requirements* and *user-defined requirements* are functional while *values* are *non-functional* in essence - for this reason, they need to be operationalized and this means "translating" them into *functional solutions*, and this goes beyond *requirements engineering*. It is not even the non-functional requirements that change this because values cannot be limited to non-functional requirements, as studied in [10].

Further, it could be that:

- We have already a demand to enforce particular atomic values (for example: the system should be *privacy-sensitive*) or;
- We have a public demand to reinforce a particular composite value (for example: the system should be instrumental with regard to pushing forward egalitarianism in society); then, it would be necessary to somehow decompose this composite value (taking the previously discussed example, *egalitarianism* is decomposed into *transparency* + *justice* + *accountability*).

Finally, as above-mentioned, the proposed design approach is consistent with SDBC [7] and all further design/modeling activities could proceed according to SDBC (and for this reason, we abstract from those activities in the current paper); moreover, SDBC has been applied in addressing the specification of value-sensitive systems [8, 9]. We hence only emphasize on the importance of decomposing composite values, such that they are adequately reflected in the corresponding software specifications.

In the following section we present experimental results that justify the claim that it is easier to capture public opinion concerning composite values (see Sect. 1). We consider composite values in general and **cultural values** – in particular, referring to the categorization of Schwartz (see Sect. 2).

4 Experimental Results

In *October – December 2017*, an **online questionnaire survey** via Google forms, applying Schwartz's methodology for measuring cultural values [6] among *Bulgarian entrepreneurs* was carried out. It contains 44 items (values) such as social power, success, freedom, etc. By using a 9-point scale (from 0 – not important to 7 – extremely important and [−1] – contrary to my values) every respondent assesses the importance of each value as a guiding principle in his/her life. Those values are grouped into six value categories (orientations) and three bipolar dimensions (alternatives).

4.1 Briefing

The number of Bulgarian entrepreneurs studied was 234. All of them fall within the age range 19-68 y.o. (average age M = 35.35; SD = 10.91): 68 are men (29.1%), 166 – women (70.9%); education – secondary 97 (41.5%), college 8 (3.4%), and university 129 (55.1%); marital status – single 68 (29.1%), married – 97 (41.5%), separated/divorced – 20 (8.5%), in cohabitation with partner 47 (20.1%), and widower/widow 2 (0.9%); residence – Sofia (capital) 167 (71.4%), regional city 48 (20.5%), another town 11 (4.7%), and village 8 (3.4%). All respondents develop private business in the field of information technology, commerce, food processing, education, pharmacy and healthcare, and others. They are owners of small and medium-sized enterprises.

4.2 Results

The empirical data is processed using *IBM SPSS Statistics 19* [5]. To test the hypotheses of the study a series of Paired-Samples T-Tests is considered. The results about value alternatives are summarized in Tables 1 and 2.

Table 1. Paired samples statistics

		Mean	N	Std. deviation	Std. error mean
Pair 1	EMB	5.33	234	0.87	0.06
	AUT	5.21	234	1.02	0.07
Pair 2	HIE	3.86	234	1.29	0.08
	EGA	5.49	234	0.90	0.06
Pair 3	MAS	5.48	234	0.98	0.06
	HAR	5.18	234	1.33	0.09

Table 2. Paired samples test

	Paired differences					t	df	Sig.(2-tailed)
	Mean	Std. dev.	Std. error mean	95% Confidence interval of the difference				
				Lower	Upper			
Pr 1	0.12	0.91	0.06	0.004	0.237	2.04	233	0.043
Pr 2	−1.62	1.33	0.09	−1.796	−1.454	−18.74	233	0.000
Pr 3	0.31	1.30	0.08	0.140	0.474	3.62	233	0.000

The mathematical and statistical analysis reveals that embeddedness dominates autonomy ($x = 5.33$ vs. $x = 5.21$ at $t = 2.04$ and $p = 0.043$), egalitarianism over hierarchy ($x = 5.49$ vs. $x = 3.86$ at $t = -18.74$ and $p = 0.000$), and mastery over harmony ($x = 5.48$ vs. $x = 5.18$ at $t = 3.62$ and $p = 0.000$), as mean values of the variables are statistically significant at $p < 0.05$. In other words, in the case of entrepreneurs, value conflicts are dealt with in favor of values of status quo, equal social relations and exploitation of environment. The results obtained partly confirm the hypothesis that individual values have predominance over collective values.

The analysis of the structural and hierarchical organization of values allows for a profound look into the value priorities of Bulgarian entrepreneurs. For that purpose, the mean of value categories is sorted in descending order. Rank 1 receives the value with the highest mean and rank 7 – the lowest one. By Paired-Samples T-Test statistically (in)significant differences between each pair of variables are tested. The type of values with statistically significant differences in the mean is classified with a single rank and those with insignificant difference share the same rank with other values (see Table 3). The structural-hierarchical model, where collective values are marked with a black circle and the individual ones – with a white triangle, is shown in Fig. 3.

Table 3. T-values from the comparison of each pair of value categories ($^{\alpha}p < 0.05$; $^{\beta}p < 0.01$; $^{\gamma}p < 0.001$)

	EGA	MAS	EMB	AUT	HAR	HIE	M	SD	R
EGA	–	0.07	4.03^{γ}	4.37^{γ}	4.11^{γ}	18.74^{γ}	5.49	0.90	1.5
MAS		–	2.85^{β}	5.05^{γ}	3.62^{γ}	23.12^{γ}	5.48	0.98	1.5
EMB			–	2.04^{α}	2.09^{α}	17.94^{γ}	5.33	0.87	3
AUT				–	0.40	18.00^{γ}	5.21	1.02	4.5
HAR					–	12.74^{γ}	5.18	1.33	4.5
HIE						–	3.86	1.29	6

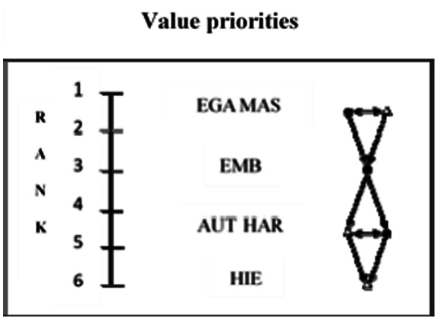


Fig. 3. Structural-hierarchical model of entrepreneurs surveyed in 2017

The visualization of the data in Fig. 3 shows that the value hierarchy of Bulgarian entrepreneurs consists of 4 structural levels: Level I with a shared rank of 1, 5 places egalitarianism and mastery, Level II with rank 3 – embeddedness, Level III with rank 4, 5 – autonomy and harmony, and Level IV with rank 6 is hierarchy. Value priorities are outlined by egalitarianism, mastery, and embeddedness. Hence the empirical facts describe the entrepreneurs' value order as internally contradictory and in a process of transformation. Evidence of this is the high position of egalitarianism and embeddedness, which are defined as values from the old ideological system of socialism [4], as well as the lower importance of autonomy and hierarchy which are values of modern society.

On the basis of the presented results, it can be speculated that the entrepreneurial culture in Bulgaria is still in the process of development and formation. The lack of a strong business elite and economic leadership, measured through the prism of cultural values and related behavioral models, is to be noted. The high importance of embeddedness in comparison with autonomy (intellectual and affective), for example, is probably due to the fact that in Bulgaria many businesses are family-run. This largely reproduces the characteristics of the national culture that occupies the forefront of embeddedness, conservatism, and collectivism.

5 Conclusion

Public values are an intuitive appealing concept that is not always easy to capture. This especially holds for atomic public values whose capturing would often assume complex trade-offs and situation-dependency. At the same time, atomic values can be methodologically reflected in technical (software) functional solutions (hence – operationalized). In contrast, composite values are difficult to operationalize but are easier to capture and measure, possibly through surveys. In this paper we have proposed value decomposition with regard to composite values, in terms of corresponding atomic values. Acknowledging the possibility to methodologically operationalize atomic values, we have justified by means of an experiment the possibility to capture and analyze composite values. Thus, capturing composite values + decomposing them in terms of atomic values + operationalizing the atomic values is expected to be an adequate approach for dealing with (composite) values, especially if they need to be reflected in specifications of information systems. This is research in progress and both the propositional and the validation parts of the current paper require further elaboration. We plan this as future research.

Acknowledgement. This work is supported by: (i) the TU Delft - Delft Pilot project; (ii) National Science Fund at the Bulgarian Ministry of Education and Science Project: DN 15/2/11.12.2017. We would like to thank Jeroen van den Hoven for his support and guidance.

References

1. Akkermans, H., Gordijn, J.: What is this science called requirements engineering? In: Proceedings of the RE 2006 14th IEEE International Requirements Engineering Conference, Minneapolis/St. Paul, USA, pp. 273–278. IEEE (2006)
2. Branzei, R., Dimitrov, D., Tijs, S.: Egalitarianism-based Solution Concepts. In: Models in Cooperative Game Theory, Springer, Berlin, Heidelberg (2008) https://doi.org/10.1007/978-3-540-77954-4_4

3. Friedman, B., Hendry, D.G., Borning, A.: A survey of value sensitive design methods. Found. Trends, p. 76 (2017)
4. Hoppe, H.H.: A Theory of Socialism and Capitalism, Economics, Politics, and Ethics. Springer, Berlin (1989). https://doi.org/10.1007/978-94-015-7849-3
5. Pradhananga, Y., Karande, S., Karande, C.: CBA: cloud-based bigdata analytics. In: Proceedings of the International Conference on Computing Communication Control and Automation, Pune, 2015, pp. 47–51 (2015)
6. Schwartz, S.H.: The refined theory of basic values. In: Roccas, S., Sagiv, L. (eds.) Values and Behavior. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56352-7_3
7. Shishkov, B.: Enterprise Information Systems, a Modeling Approach. IICREST, Sofia (2017)
8. Shishkov, B., Janssen, M.: Enforcing Context-Awareness and Privacy-by-Design in the Specification of Information Systems. In: Shishkov, B. (ed.) BMSD 2017. LNBIP, vol. 309, pp. 87–111. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78428-1_5
9. Shishkov, B., Larsen, J.B., Warnier, M., Janssen, M.: Three categories of context-aware systems. In: Shishkov, B. (ed.) Proceedings of BMSD 2018 - 8th International Symposium on Business Modeling and Software Design Lecture Notes in Business Information Processing, vol. 319. Springer, Cham (2018)
10. Shishkov, B., Mendling, J.: Business process variability and public values. In: Shishkov B. (ed.) Proceedings of BMSD 2018 - 8th International Symposium on Business Modeling and Software Design. Lecture Notes in Business Information Processing, vol. 319. Springer, Cham (2018)
11. Shishkov, B., van Sinderen, M.: From user context states to context-aware applications. In: Filipe, J., Cordeiro, J., Cardoso, J. (eds.) ICEIS 2007. LNBIP, vol. 12, pp. 225–239. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88710-2_18
12. UML, The Unified Modeling Language (2018). http://www.uml.org
13. Veenhoven, R., Kalmijn, W.J.: Inequality-adjusted happiness in nations egalitarianism and utilitarianism married in a new index of societal performance. J. Happiness Stud. **6**(4), 421–455 (2005)

Towards a Methodology for Designing Micro-service Architectures Using μσADL

Tasos Papapostolu(✉) and Dimitar Birov

Sofia University "St. Kliment Ohridski", 1000 Sofia, Bulgaria
{papapostol,birov}@fmi.uni-sofia.bg

Abstract. Microservice Architecture is an innovative architectural style used for the design of software applications as independent, loosely coupled and strongly cohesive deployable services. Architecture Description Languages (ADLs) are formal languages used in the domain of Software Architecture in order to describe software intensive systems. In this paper, we introduce μσADL, a new ADL which is an extension on top of jADL (a formal ADL). μσADL aims to provide to software architects the means to adequately describe software systems that adopt this new architectural style. It provides an additional layer of abstraction omitting or hiding rigorous definitions. The basic concepts of μσADL are presented through an illustrative example. We propose a methodology for designing software systems that are built based upon this architectural style, using μσADL. Through an illustrative example, we present how from a Business Process Modelling Notation diagram, designers can obtain a verifiable formal specification of the architecture. Additionally, designers or programmers that follow this methodology can obtain source code stubs.

Keywords: Microservices · Architecture Description Language
Software architecture · jADL · Business Process Modelling

1 Introduction

Microservice Architecture (Microservices) [2, 3] is a newly adopted architectural style which emerged in the last decade and becomes more and more popular. A number of industry leading companies, such as Amazon, Netflix, etc., have migrated to microservices and initiated the development of a new technology of software development with very encouraging and promising results.

Theory and practice of software architecture have been developed in order to abstract over the structure and architecture of (distributed) applications. Though the microservices architectural style itself is not yet precisely defined, a broadly accepted definition is the one given by Lewis and Fowler [1]. They define microservices as "*an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API. These services are built around business capabilities and independently deployable by fully automated deployment machinery.*". They have, also, outlined some of the common characteristics of this style: (i) *Componentization via Services*, (ii) *Organized around Business Capabilities*, (iii) *Decentralized Governance*, (iv) *Decentralized Data Management* and (v) *Design for failure* [1].

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 421–431, 2018.
https://doi.org/10.1007/978-3-319-94214-8_33

Architecture Description Languages (ADLs) are formal languages used in the domain of Software Architecture in order to formally describe software intensive systems. As [9] point out there is a lack of such a formal language when it comes to specifically describing microservice architectures and architects tend to use languages that describe service-based architectures like SoaML, SOMA, etc.

In this paper, we are introducing μσADL, an ADL which is an extension of jADL [7, 8]. jADL is a formal ADL developed by the authors for describing static/dynamic/mobile architectures. μσADL aims to provide architects and various stakeholders the means to adequately describe software systems that adopt this new architectural style alongside their software qualities. It also provides practical means to architects so that they can easily describe microservice architectures. It provides an additional layer of abstraction omitting or hiding unnecessary rigorous definitions (e.g. generic database connectors below) easing the description of microservices.

We, also, propose a methodology for describing software systems that are built based upon this architectural style, using μσADL and jADL, consisting of 4 parts. (i) The extraction of an initial architectural sketch (Fig. 2) from a BPMN representation where each process (or a number of processes depending on the architect's choice) can be modeled as a microservice. At this stage the granularity of each microservice is defined. (ii) The specification of the architecture using μσADL. The language provides the necessary constructs to define both each of the microservices and the overall architecture of the system (i.e. their communication mechanisms, etc.). (iii) The translation of the μσADL description to jADL description. The compiler built for jADL (an initial prototype which is continuously updated) can be used for the formal analysis and validation of the defined architecture. (iv) Using the translator (which we are still developing - it has been tested for simple architectures so far) we translate the jADL architectural description into π-ADL [13] in order to use the tool PiADL2GO [14], so that we can reach to the generation of code. This last part is outside of the scope of this paper and will be presented in later work after the translator has been more intensely tested and finalized.

In the next section we discuss related work and in Sect. 3 we briefly present the basic structure of the microservices defined in the language and their communication mechanisms. In Sect. 4 we present an illustrative example where we define the architecture of a system using the microservices architectural style, through a Business Process Modelling Notation diagram (a standardized method for modelling business processes). Finally, conclusion and future work is discussed.

2 Related Work

There is a growing number of studies concerning microservice architectures which focus on monitoring and management [4], or the recovery of a micro-service architecture from an existing software system [5] etc. Despite that we could not find an approach such as the one described here for the formal description of a microservice architecture starting from a BPMN representation and reaching to programming code or secondary stubs.

A similar approach was presented in [6], yet it concerned the description of service-oriented architectures.

3 Microservices in μσADL

μσADL as a regular architectural language has two parts: textual representation of an architectural script, as well as a graphical part – graphical representation of the architecture. A microservice in μσADL is comprised of ports, a set of required attributes, (optionally) its private database (which can be directly accessed only by itself) and its *behavior*. Ports are defined the same way they are defined in jADL (two types – *provides* and *requires* [7, 8]) and they are configured (if necessary) using the *config* statement [7, 8]. Each microservice represents a computational and data store element. Thus, it represents a component when it is translated from μσADL to jADL and all the statements/operators/etc. defined in jADL for components can be used when defining a microservice. In this section, we focus on the communication mechanisms between different microservices and the definition of their databases.

3.1 Data Storage

In monolithic applications it is usually preferred a single logical database when it comes for storing persistent data. On the other hand, in a microservice architecture it is favored a decentralized data management approach; each microservice manages its own database, either different instances of the same database technology or entirely different database systems (*Polyglot Persistence*) [1].

Following this principle, in μσADL we allow for each microservice to define its own instance of a database (e.g. *inventory* description below). Using the keyword ***database*** and inside the ***{ }*** the architect can define the necessary attributes for creating the connector he/she desires for a given microservice and a database. This way we provide an elegant way for defining the connections between a microservice and its database in μσADL, hiding all the formal requirements of jADL. Using this simple description, we can then automatically generate the appropriate connector in jADL.

For example, assuming we have a microservice located in the same location with its database (localhost) and we need a JDBC [10] standard connector. The description of the database and its connector in μσADL would be:

```
database {
  location: localhost;
  connector: JDBC;
  schema: invSchema;
  username: user1;
  password: mypass;
}
```

The description presented above would result in creating in jADL a new database component and its appropriate connector. The connector in jADL would be:

```
interface IConnJDBC {
  service void sendQuery (sqlString data);
  service void getQueryRes (sqlString data);   }

connector ConnJDBC {
  provides role IConnJDBC pClient;
  requires role IConnJDBC rClient;
  provides role IConnJDBC pDB;
  requires role IConnJDBC rDB;
  attribute string location = "localhost";
  attribute string username = "user1";
  attribute string password = "mypass";
  attribute string schema = "invSchema";

  config pClient as {
    service void getQuery (sqlString data) {
      rDB.sendQuery(data);
    }  }
  config pDB as {
    service void sendQuery (sqlString data) {
      rClient.getQuery(data);
    }  }  }
```

The number of attributes is not limited to the ones shown in this example. The architect can add any number of attributes describing the database and its connection, which are then integrated in the code of the generic connector generated in jADL.

3.2 Communication Between Microservices

Microservice architecture aims to lead to applications where the coupling is as loose as possible and the cohesion is as strong as possible. An approach usually described as *smart endpoints and dumb pipes* [1]. It can be viewed as the filters in a Unix sense; microservices receive a request, apply the appropriate logic and produce a response. There are two ways primarily used for communication when building an application with microservices; *direct communication* using lightweight protocols (e.g. REST [11]) or *messaging* over a lightweight message bus [1].

In μσADL this is modelled in the following way. In the first case we have a typical Client-Server architectural pattern where one microservice acts as a client and sends a request to a second microservice (acting as a server), from which it awaits a response. The definition of the attributes concerning their locations (so that they are discoverable) and the ones concerning their protocols (the way the communication will be taking place) is enough for the appropriate connector to be created and the two microservices (in μσADL) or components (in jADL) to be able to communicate. Generic connectors are used here, provided by jADL, and the generation of such connectors when translating to jADL is similar to the one shown above in Sect. 3.1. Due to limited space we won't deal extensively with this type of communication between microservices, but an example of how this can be represented in jADL can be found in [8].

The second way of microservice communication is through a lightweight message/event bus. Each microservice that is subscribed to the bus produces messages/events that pushes to the bus and consumes messages/events from it. μσADL provides generic message and event buses for architects to use. Alongside with the use of predefined integrated in jADL *communication traits* [7], varying types of buses can be described (e.g. publish/subscribe queues, routing queues etc.). They can be called directly, instantiated and used in the architectural description. Due to lack of space more information about buses, communication traits and the implementation of Message/Event Bus Architectural Pattern in the language can be found in [7].

4 Illustrative Example

Business Process Modelling Notation (BPMN) is a standardized visual notation for modelling business processes. In the figure below, we present a simple process of online shopping through a shopping site.

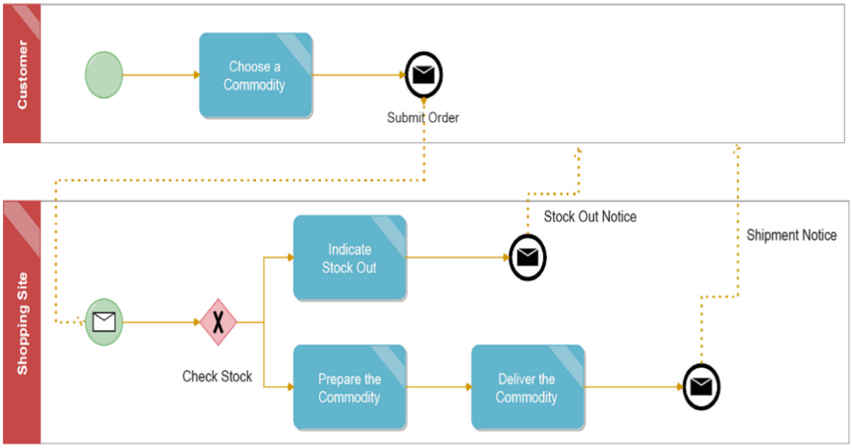


Fig. 1. Online shopping process in BPMN [12]

A customer chooses a commodity and sends its request to the site. The inventory is checked and either an out-of-stock notice is sent or a notice concerning the shipment details. The architecture of this system is dynamic: commodities can be added or removed and the way of delivering the goods may vary from customer to customer.

From Software Architecture point-of-view this is a typical Client-Server communication model. The client (customer) sends its request to the server (shopping store) and after the request is processed a response is sent accordingly.

We focus now on the server and how it is organized. A typical approach from the past years would be a monolithic architecture. A single-tiered software application in which the user interface and data access code are combined into a single program from a single platform. It provides certain advantages, like simplicity when developing or deploying and (usually) a relatively easy way for scaling by running multiple instances behind a load balancer. But, as it has been observed, once an application becomes larger and larger significant drawbacks appear; the scaling of components with different resource requirements (CPU or memory intensive) is not possible, an update of a component requires the redeployment of the whole application, it requires long-term commitment to the technology stack (sometimes even to specific versions) chosen at the beginning of the development, etc.

Using the microservices architectural style described above the server can be componentized as follows. Each of the processes can be modeled as a separate microservice – the receiving of an order, the check of the inventory and the shipment information (as shown in Fig. 2). These three microservices communicate with each other using a message bus (Fig. 2). In architectural terms these various microservices can be viewed as components (locus of computation) and the message bus as a connector (locus of communication). In Fig. 2 we can see an initial architectural sketch concerning the architecture of the system (extracted from the BPMN model of Fig. 1) and in Fig. 3 a formal graphical representation of this architecture (translated in jADL) is presented.

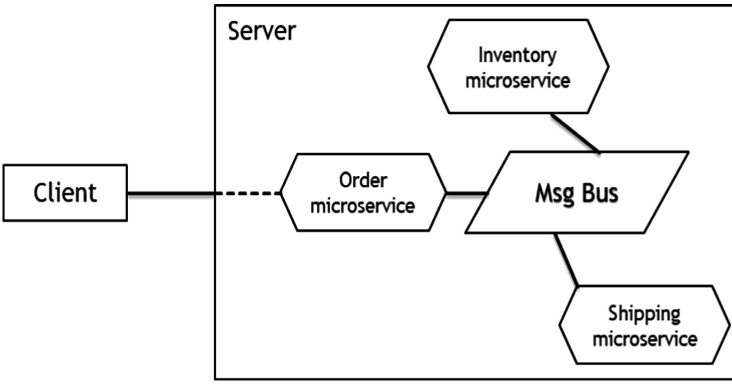


Fig. 2. Online shopping system architectural sketch

The first step of the methodology is the extraction of an initial architectural sketch of the architecture of the software system from a BPMN model. Here we chose to model each of the business processes as a separate microservice. At this stage the granularity of the microservices in an architecture is up to the architect. The definition of the appropriate granularity is still a field where there is a lot of on-going research and it is one of the tools we aim to integrate in μσADL in future work so that we can provide a formal and automated way for its validation. The second step of the methodology proposed concerns the description of the architecture in μσADL. We present the description of two of the total three microservices defined (due to lack of space) and the overall architecture of the server component (the microservices and the message bus used for their communication). The language constructs provided by μσADL proved to be adequate for the description of each microservice and their communication mechanisms. Rigorous and too formal semantics are "hidden" in μσADL and the architect can define the architecture in a simple and elegant way. During the third step of the methodology an automatic translation of the description from μσADL to jADL takes place. This is done in order to use the compiler built for jADL (an initial prototype which is continuously upgraded) for the formal analysis and validation of the defined architecture. Due to limited space the whole generated textual architectural description in jADL is not presented here, but a graphical representation of the server can be seen in Fig. 3. The final step of our methodology aims to lead to the generation of implementation code stubs. As mentioned in the introduction this step is out of the scope of this paper since it is still under development and experimentation. So far, we have implemented a translator from jADL to π-ADL (which has only been tested for simple architectural descriptions) in order to use the PiADL2GO generator which generates implementation code in the GO programming language. In future work we aim to present a fully-functional version of the translator and additionally to experiment with the generation of stubs in different programming languages (e.g. Java).

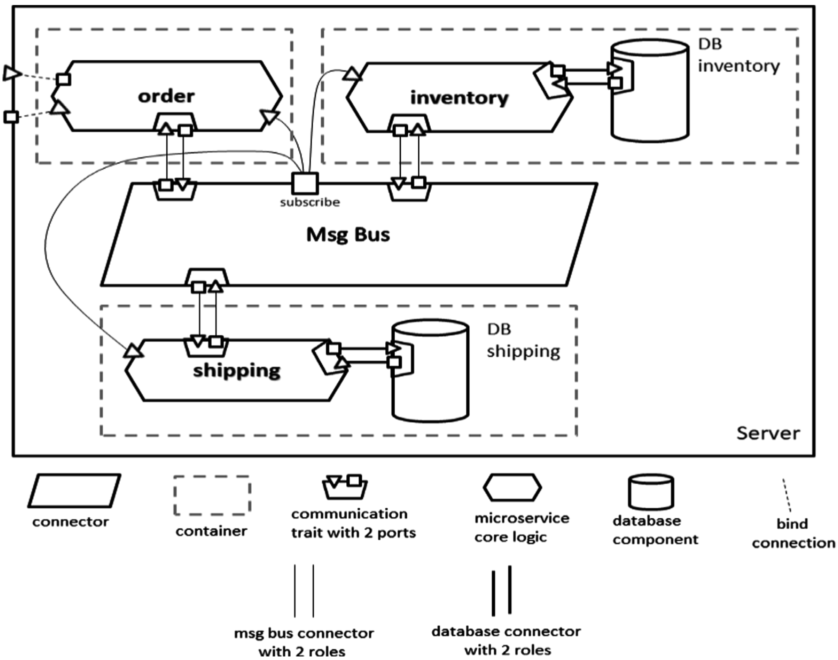


Fig. 3. Graphical representation in jADL of the server component

Using this architectural style, we achieve decentralized governance, the scaling and/or update of each component can be achieved independently of the rest of the components and the choice of the technology used (programming language, database type, etc.) can be done autonomously for each of the components. Additionally, the adding of new functionalities can be easier, in respect to monolithic architectures, since we can simply add a new microservice without (usually) having to worry about the other parts (components) of the application.

```
microservice Order {
  provides port IProcess req;
  requires port IResponse reply;

  trait CTrait aggregate CommTrait {
    config p as {
      service void getMsg (Message msg) {
        reply(msg);
      }   }   }

  instance com1 = new CTrait();

  config req as {
    service void procRequest (type data) {
      com1.r.sendMsg(com1, data);
    }   }   }
```

In order for the microservices to communicate we chose the message bus architectural pattern (MBAP) described in [7]. This is integrated in jADL and the communication trait needed is:

```
trait CommTrait {
  provides port IReceiveMsg p;
  requires port ISendMsg r;   }
```

When a microservice (or a component in jADL) uses this trait, it needs to define the *behavior* of the *provides p* port as shown in the code for *Inventory* and *Order*.

We chose this implementation of the MBAP for simplicity and due to lack of space. The architect can define different implementations of the pattern according to the needs of the system.

```
microservice Inventory {
  requires port ISubscribe r;

  trait InvTrait aggregate CommTrait {
    config p as {
      service void getMsg (Message msg) {
        reply(msg);
    }  }  }

  instance com1 = new CTrait();

  database {
    location: "localhost";
    connector: "JDBC";
    schema: "invSchema";
    username: "user1";
    password: "mypass";
  }

  config pDB as {
    service void getQueryRes (type data) {
      com1.r.sendMsg(com1, data);
    }   }   }
```

In the declaration of this microservice we have an additional database declaration which will lead to the creation (in jADL) of the connector explained in Sect. 3.1. Upon the instantiation of this microservice inside the server component (using the keyword ***new***) a database component will be created and attached to the component inventory using the previously mentioned connector. This is achieved through the use of *communication traits* and a graphical representation of this connection can be seen in Fig. 3. The same applies for the declaration of the microservice *Shipping* which is omitted here.

```
component Server {
  provides port IProcess req;
  requires port IResponse reply;
  instance order = new Order();
  instance inventory = new Inventory();
  instance shipping = new Shipping();
  instance msgBus = new MessageBus();

  attach(msgBus.s, order.r);
  attach(msgBus.s, inventory.r);
  attach(msgBus.s, shipping.r);
  order.r.subscribeTo(com1);
  inventory.r.subscribeTo(com1);
  shipping.r.subscribeTo(com1);    }
```

After the instantiation of the server component we can define the architecture of the online shopping system which comprises of the *server* previously described, a *client* and a *connector*. Such a Client-Server architecture (and in a more complex case with the use of a *Load Balancer* server) is adequately described in jADL in [8].

In this section, through an illustrative example, we have presented how our proposed methodology for the formal description of microservice architectures can be applied. Using a simple generic BPMN model that describes the business processes concerning an online shopping store we have reached to the (automatically generated) formal description of the architecture in jADL.

5 Conclusion

We have presented μσADL, an extension to the jADL language. It is used for the description of microservice architectures. Through an illustrative example, where we used a simple scenario of an online shopping store, we presented our methodology for describing a microservice architecture using μσADL and jADL.

Our future work lies in the finalization of the translator so that GO programming language implementation code can be generated. Also, the additional development of the compiler of jADL so that more and/or more complex implementation code studs can be generated in different languages and/or the creation of secondary models.

References

1. Fowler, M., Lewis, J.: Microservices: a definition of this new architectural term. http://martinfowler.com/articles/microservices.html. Accessed 01 Apr 2018
2. Amundsen, M., McLarty, M., Mitra, R., Nadareishvili, I.: Microservice Architecture - Aligning Principles, Practices, and Culture. O'Reilly Media, Newton (2016)
3. Newman, S.: Building Microservices - Designing Fine-Grained Systems. O'Reilly Media, Newton (2015)
4. Mayer, B., Weinreich, R.: A dashboard for microservice monitoring and management. In: 2017 IEEE International Conference on Software Architecture Workshops (ICSAW), pp. 66–69 (2017)
5. Granchelli, G., Cardarelli, M., Francesco, P.D., Malavolta, I., Iovino, L., Salle, A.D.: Towards recovering the software architecture of microservice-based systems. In: 2017 IEEE International Conference on Software Architecture Workshops (ICSAW), pp. 46–53 (2017)
6. Oquendo, F.: π-ADL for WS-composition: a service-oriented architecture description language for the formal development of dynamic web service compositions. In: SBCARS (2008)
7. Papapostolu, A., Birov, D.: Structured component and connector communication. In: Proceedings of BCI 2017, Macedonia (2017). https://doi.org/10.1145/3136273.3136291
8. Papapostolu, T., Birov, D.: Architectural self-adaptation and dynamic reconfiguration in jADL. In: Proceedings of the Forty-Seventh Spring Conference of the Union of Bulgarian Mathematicians, pp. 168–177 (2018)
9. Francesco, P.: Architecting microservices. In: 2017 IEEE International Conference on Software Architecture Workshops (ICSAW) (2017)

10. Ibm.com. IBM Knowledge Center. https://www.ibm.com/support/knowledgecenter/en/SSCQGF_7.1.1/com.ibm.IBMDI.doc_7.1.1/referenceguide37.htm. Accessed 01 Apr 2018
11. Spring.io. Understanding: REST. https://spring.io/understanding/REST. Accessed 01 Apr 2018
12. Online Shopping Process BPMN. https://www.edrawsoft.com/template-online-shopping-process-bpmn.php. Accessed 01 Apr 2018
13. Cavalcante, E., Oquendo, F., Batista, T.: π-ADL: a formal description language for software architectures. Technical report, UFRN-DIMAp-2014-102-RT (2014)
14. ArchWare Team: piadl2go: Software Architecture-Based Code Generation. https://www-archware.irisa.fr/software/piadl2go/. Accessed 01 Apr 2018

Monitoring the Software Development Process with Process Mining

Saimir Bala(✉) and Jan Mendling

Vienna University of Economics and Business, Vienna, Austria
{saimir.bala,jan.mendling}@wu.ac.at

Abstract. Software projects typically need to be monitored in detail regarding when what was done in order to demonstrate adherence to methodologies, rules, regulations, guidelines or best practices. To this end, it is of utmost importance to obtain factual knowledge from empirical evidence about the actual software development process. A major problem in this context is the lack of a centralized control of by a central system. Although it is hard to obtain full knowledge of the overall software development process, several cues can be gathered by analyzing pieces of information that are stored by supporting IT systems (e.g., issue trackers and version control). This position paper presents research in progress for extracting process knowledge from the historical data of software artifacts. This work extends the applicability of process mining techniques to software processes.

Keywords: Mining software development · Process mining
Mining sofrware repositories · Project mining · Artifacts evolution

1 Introduction

Monitoring the process of software development is highly important to project managers who want to deliver good software in time and within budget. In practice, various project management approaches exist that guide the managers during the development of software projects. Usually these guidelines and methodologies stem from experience. However, in practical scenarios, every software development endeavor is different. Therefore, there is the need to extract the real software process from the data. A starting point is the empirical evidence given from the historical evolution of the artifact. This provides important cues for monitoring the adherence of actual work to existing projects plans and milestones.

Literature has addressed several aspects about getting process insights from historical data. However, the problem has been tackled separately from different disciplines. Contributions from the process mining discipline focus on obtaining process models from well structured event logs [1]. Contributions from the mining software repositories area focus on obtaining results from a software engineering point of view, e.g., code quality, code complexity, user analysis, functional

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 432–442, 2018.
https://doi.org/10.1007/978-3-319-94214-8_34

dependencies, software visualization, etc. [16]. Lastly, text mining has no notion of process, but it is a fundamental field of study that helps dealing with unstructured data, such as user comments in a software repository [3].

Process mining techniques fall short when it comes to mining the overarching software process. Nevertheless, several pieces of traces of the software process can be found in system logs. Therefore, our driving question can be stated as follows. **RQ:** *What can be discovered from software development data for obtaining knowledge about the process?*. To answer the question, we consider extending the process mining discipline towards software development. As a result, this work aims at assisting project managers in decision making. This is done by extracting knowledge from data that is generated during the lifetime of software projects. Unstructured data from the tools (e.g. user comments from Version Control System (VCS) and Issue Tracking System (ITS)) are also taken into account. The expected result is a body of techniques and tools that can be used to for analyzing software data in a process mining fashion. In this way, we create bridge between the research areas of process mining and mining software repositories.

The rest of the paper is organized as follows. Section 2 presents the problem and related literature, and draws the requirements for a solution. Section 3 provides the challenges and how they can be tackled. Section 4 draw the conclusions and provides a plan to address these challenges.

2 Background

2.1 Problem Definition

Software development processes are characterized as follows. First, although there is a planning phase, they are conducted creatively, i.e. there exists no strict process model that is followed by the developers. Second, they follow guidelines and methodologies on software development project, such Rational Unified Process (RUP), Scrum[1], Waterfall, etc. These guidelines are recommendations that stem from experience. However, none of them is extracted from empirical evidence. Third, typically there is a plethora of tools used and process evidence is scattered among different logs. Fourth, there is typically no central system to control the overarching process execution. Fifth, it is highly relevant to track the process because of project managers want to be able to efficiently detect performance measures such as number of issues, bottlenecks, timeliness, handover of work, productivity, etc.

Software development processes fall into the category of *project-oriented business processes* [6]. That is, ad-hoc plans performed with limited resources and time, but with a clear goal: develop a software artifact. Unlike classic business processes which are best captured with notations such as Petri nets and BPMN, software development processes fall into the category of *project-oriented business processes*, which are usually captured by models such as Gantt and PERT diagrams.

[1] https://www.scrum.org.

An example of software development is the following. A new version of a software needs to be developed by company X. In general, company X knows how to develop software. However, before running straightly into the development phase, the company first makes sure to gather and properly formulate all the requirements. In a Scrum scenario these requirements would be written down as user stories. At a later stage, the project manager needs to plan the time and resources allocated to the respond to project deadlines. He can go through the list of requirements that need to be implemented and assign to an effort estimation value to each of them. With the plan done, the development phase can commence. During the development phase, resources address the task in a creative way, choosing the order of the tasks according to their own knowledge and expertise, until eventually all the tasks are terminated.

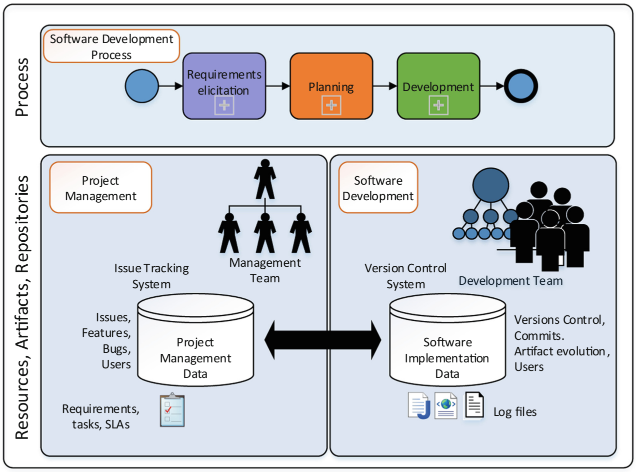


Fig. 1. Problem context

Several tools are used by software project participants to support their work. Therefore, traces about the overall process are typically scattered among different repositories and different artifacts, e.g., spreadsheets, word processor documents, programming languages files, emails, etc. This makes it difficult to obtain full knowledge about the overall business process. Two major challenges are to *correlate* heterogeneous events from different sources of evidence pertaining the same process and to *extract* information from unstructured data such as user comments when working on tasks. Thus, obtaining full process knowledge is a complex task.

Although the data is scattered and often unstructured, there are still repositories that we can study in order to obtain process knowledge. An important dimension of the software development process is the work perspective. It is particularly interesting for project managers to know if planning phase was realistic with respect to the development efforts. Two software tools that are commonly used in software development are ITS (e.g., JIRA, GitHub Issues) and VCS (e.g. Subversion, Git). The evolution of these two repositories can be analyzed to extract relevant knowledge about the software development process. Figure 1 illustrates the problem context.

Existing software repositories allow for many ways to access their log files. However, the most relevant information that they provide are about actions done by the user to change the repository state. In the more specific case, VCS logs consist of an ordered set of *commits* bearing information about users, files, timestamp, comments and type of change that were stored at particular moment in time. ITS typically come with richer information, most importantly they inform about users, task, type of task (e.g., bug, new feature, requirement, etc.), timestamps, related issues, etc. To extract knowledge about the software development process, these properties of repositories must be taken into account. Table 1 gives an overview of the main information that can be found in systems, such a VCS.

Table 1. An excerpt of a VCS log data

Id	Resource	Date	Comment	Diff
1	John	2017-01-31 12:16:30	Create readme file	diff –git a/README.md b/README.md @@ -0,0 +1 @@ +# StoryMiningSoftwareRepositories
2	Mary	2017-02-01 10:13:51	Add a license	diff –git a/README b/README @@ -1,0 +2,3 @@ +The MIT License (MIT) + +Copyright (c) 2015 Mary+
3	Paul	2017-02-02 16:10:22	Updated the requirements.	diff –git a/README.md b/README.md @@ -1,4 +1,5 @@ + # string 1, string 2, string 3 diff –git a/requirements.txt b/requirements.txt @@ -0,0 +1 @@ +The software must solve the problems
4	Paul	2017-02-02 15:00:02	Implement new requirements	diff –git a/model.java b/model.java @@ -1,9 +1,10 @@ +public static methodA(){int newVal=0; @@ -21,10 +23,11 @@ + ”1/0”,,”0/0”, diff –git a/test.java b/test.java @@ -0,0 +1,2 @@ +//test method A +testMethodA()

Because the project managers can benefit from a process view to better analyze hidden aspects (such as the behavioral one) of the software development process, we focus on adapting and transforming the data to fit process mining. Thus, we can better define our problem based on the four aspects of process mining [1]: *(i)* time perspective; *(ii)* control-flow perspective; *(iii)* organizational perspective; and *(iv)* case perspective. In the following, we will define four requirements, each addressing one process mining perspective.

2.2 Related Literature

This problem is related to three areas: *(i)* Process Mining (PM); *(ii)* Mining Software Repositories (MSR); and *(iii)* Text Mining (TM).

Process mining contributions have focused on transforming this problem into a process mining problem. These approaches enrich VCS log data with case and activity information and consequently use process mining to discover a model. In this category, Kindler et al. [11,12] can discover a Petri net from a structured and enriched version control log. This approach was further improved by Rubin et al. [19] and a ProM[2] was provided. Poncin et al. [17] provide the FRASR framework for preprocessing software repositories such that they can be used in ProM. While providing interesting insights, these contributions leave out many important aspects of software development projects, such as for instance trying to understand whether the process was done according to the plan.

MSR focuses on software engineering aspects, like code quality metrics, modularization, code complexity, user networks, and other important metric. In general, methods from MSR provide useful dependency analyses of the repository structure. Contributions in this area focus on the users, the artifacts and the repository evolution [9,23], and network analysis of file dependency graph based on commit distance [2,22]. Also techniques for trend analysis [20] and interdependencies between developers [14] are proposed. However, none of these works aims at extracting knowledge about the business process.

TM focuses on obtaining structured information from unstructured textual data, and mainly uses Natural Language Processing (NLP). Works that use NLP can be found in both PM and MSR. In the Business Process Management (BPM) area, NLP techniques have been used to understand process activities [13,15] and analyze software processes under a knowledge-intensive perspective [10,18]. Likewise, in the MSR area, NLP has been used as an information extraction tool to obtain informative metrics from a software engineering perspective [8,21].

This paper combines ideas from the above mentioned areas to devise algorithms for mining the software development process by exploiting techniques from MSR.

2.3 Research Question and Solution Requirements

In the light of the above considerations, we derive the following requirements for extending process mining towards the analysis of software repositories.

R1. Mining the Time Perspective. Given a software repository, extract information about the temporal order of the activities. For example, the development activity takes 2 weeks on average, the average time of task creation is 15 min, etc.

R2. Mining the Case Perspective. Given a software repository, extract information about the case perspective. For example, all the bugs are solved in a 3

[2] http://www.promtools.org.

steps iteration, or a quality piece of code takes a conversation with 3 people and is successfully merged into the main branch after 1 week, etc.

R3. Mining the Organizational Perspective. Given a software repository, extract information about the organizational perspective. For example, the software development is carried out by a team of 4 people, the actual user roles of the company are developer and tester, etc.

R4. Mining the Control-Flow Perspective. Given a software repository, extract information about the control-flow perspective. For example, the testing is always done before development, or while new features are worked on, also new requirements are created, etc.

3 Extracting Knowledge on Software Development

This section presents the approach for addressing the requirements. In the following, we describe approaches and challenges to address the four requirements.

3.1 Mining the Time Perspective

Mining the time perspective from a software repository is defined as extracting temporal process knowledge. This problem relates to monitoring whether the process was executed respecting a predefined plan. Typically the process involves various actors which track their work through the use of VCS. Often, the plans are not represented in standard process notation. Rather, Gantt or PERT charts are used.

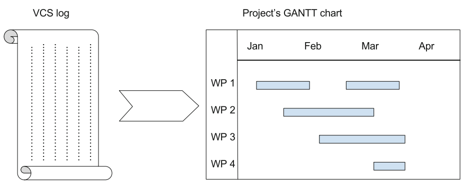


Fig. 2. Mining the time perspective of software development into a Gantt chart

Challenges are *(i) time approximation* (i.e., when the activity really started, compared to when it was registered in the log); *(ii) granularity* (i.e., be able to switch from a detailed view of the single events and a coarse-grained view of the overall project); and *(iii) coverage* (i.e., how much work effort was put within the duration of the activity). Figure 2 depicts the idea of mining a Gantt chart from VCS logs. The output is presented in a way that is informative to project managers. Details about the technique can be found in [6].

3.2 Mining the Case Perspective

The identification of process cases is not an easy task when it comes to software development data. However, a highly informative case candidate can be considered the data artifact itself. According to [1], cases can also be characterized by the values of data elements. In line with this, it is possible to devise techniques that study the *artifact evolution*. Although it can be input to process mining techniques, this evolution can also be analyzed as a time series. An example of such evolution is given in Fig. 3, which shows the lines of code (LOC) changes over time.

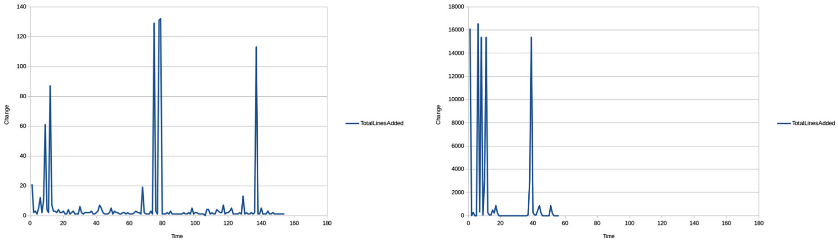


Fig. 3. Evolution of files in a version control systems: LOC changes over time

Challenges related to the artifact evolution pertain *prediction* of plateaus and *pattern recognition*. The results reflect work patterns over the artifact. For example, when a data file is not being modified anymore, its time series has a plateau and it can be interpreted that the document is now ready for release. A relevant problem in software development is bad modularization. Especially, the dependency of two artifacts on one another is considered a bad practice. Two time series can be compared together and file dependencies can be found that reflect work coupling. This is equivalent to finding dependence between two processes which might have been designed for different purposes.

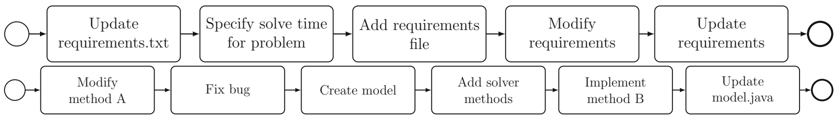


Fig. 4. Processes of two work-dependent files

We have devised a proof-of-concept in [7]. Figure 4 shows a possible outcome of the technique. The result is obtained by identifying couples of files with similar evolution and mining a business process from the comments. As the result is aggregated, several comments have been combined together an mined as a story. Among other things, the technique allows to profile existing software development projects.

3.3 Mining the Organizational Perspective

Software development projects are knowledge intensive, thus involve creativity and flexibility. Project participants are free to tackle development tasks according to their expertise and ideas to solve ad-hoc problems. De facto, members may behave according several roles in the organization. Therefore, the discovery of the roles of a software project may give important insights on the actual project. Role discovery can help in the monitoring existing projects in two ways. First, emerging roles can be analyzed in order to have a better skills profiling of the resources. Second, emerging roles can be used to check whether resources are breaking contractual agreements or norms.

We have developed an approach for tackling the problem of mining the organizational perspective in software development projects. The approach provides information about the actors involved in the business process and their relations. In substance, it provides automatic classification of resources into company roles (e.g., developer, tester, etc.) based on the comments of project participants. It works on VCS logs and provides information about the people, their roles, other systems involved, the organization hierarchy, the social network, and resource profiling. Figure 5 shows an example of resource profiles automatically inferred from VCS comments.

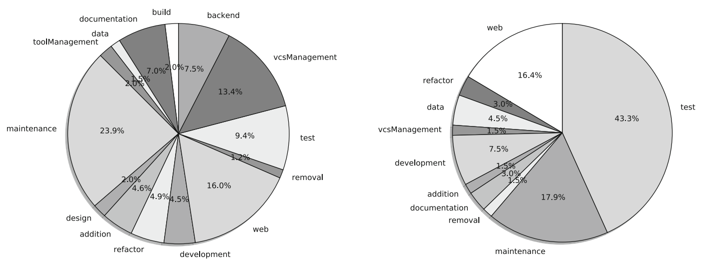


Fig. 5. Profiles of a developer and a tester, from left to right.

Challenges of using NLP concern the *jargon* used by software developers. They often use technical terms, very short phrases, or a number of codes and hyperlinks. These makes NLP techniques score low results even on simple tasks like sentence parsing, or name-entity recognition. Therefore, approaches should take into account several factors when mining for the organizational perspective. The result allows the manager to have measurable information about the resources. For example, do two people work better together and how can we build the best team? Details of the approach to extract resource profiles can be found in [4].

3.4 Mining the Control-Flow Perspective

Important knowledge of the control flow of a software development process can be extracted from VCS and ITS. One important mechanism widely used in software development for collaboration are *pull requests*. A pull request is issued every time a developer wants to contribute to the main source with a new piece of code. Pull requests trigger communication among different people and may help shedding light into problems, learn new things and obtain new ideas on the existing software. It is interesting for project managers and developers to better understand factors and behavior that makes a pull request successful.

In ongoing work, we are investigating the relation of the pull request process and surrounding factors such as the generation of new ideas or change of behavior. We have obtained a data set of pull requests spanning over two years from an real world repository. User conversations have been manually annotated with codes that classify them into categories of comments. We are current able to discriminate whether a comment is a new idea, an assumption, a merge that closes a pull request, etc. Considering the pull request identifiers as cases and the codes as activities, we are able extract the process of the pull requests. It is interesting to compare how the conversation (i.e., behavioral perspective) unfolds before and after a pull request is merged. We plan to explain these differences by also taking into account further information from text such as emotions.

The challenge of mining the control-flow usually pertains the completeness of data and the case and activity identification. While PM contribution have been focusing mostly on the control-flow perspective, techniques from MSR hardly go beyond mining the bug lifecycle [5]. The nature of software development process makes it difficult to recognize recurrent activities when only VCS are analyzed.

4 Expected Contribution and Implications for Research

This position paper presents research on mining the software development process. Approaches towards discovering the perspectives of time, resources and cases have been developed and the results suggest that obtaining process knowledge from software repositories brings new insights for the project managers [4,6,7]. In future work, we plan to complete the research by tackling the problem of discovering the flow-perspective of the software development process. Usefulness will be evaluated through user studies.

This research extends the process mining field towards its adoption on software repositories. Project managers would benefit from this research by having a process perspective on their ongoing projects through visual models and diagrams, thus overcoming existing flat approaches based on simple indicators such as burndown charts or change plots offered by existing tools.

References

1. van der Aalst, W.M.P.: Process Mining: Data Science in Action. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4
2. Abate, P., Boender, J., Di Cosmo, R., Zacchiroli, S.: Strong dependencies between software components. In: 2009 3rd International Symposium on Empirical Software Engineering and Measurement, ESEM 2009, pp. 89–99 (2009)

3. Aggarwal, C., Zhai, C.: Mining Text Data. Springer, Heidelberg (2012). https://doi.org/10.1007/978-1-4614-3223-4
4. Agrawal, K., Aschauer, M., Thonhofer, T., Bala, S., Rogge-Solti, A., Tomsich, N.: Resource classification from version control system logs. In: EDOC Workshop, pp. 249–258, September 2016
5. Akbarinasaji, S., Caglayan, B., Bener, A.: Predicting bug-fixing time: a replication study using an open source software project. J. Syst. Softw. **136**, 173–186 (2018)
6. Bala, S., Cabanillas, C., Mendling, J., Rogge-Solti, A., Polleres, A.: Mining project-oriented business processes. In: Motahari-Nezhad, H.R., Recker, J., Weidlich, M. (eds.) BPM 2015. LNCS, vol. 9253, pp. 425–440. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23063-4_28
7. Bala, S., Revoredo, K., de A.R. Gonçalves, J.C., Baião, F., Mendling, J., Santoro, F.: Uncovering the hidden co-evolution in the work history of software projects. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 164–180. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_10
8. Chen, T.H., Thomas, S.W., Hassan, A.E.: A survey on the use of topic models when mining software repositories. Empir. Softw. Eng. **21**(5), 1843–1919 (2016)
9. D'Ambros, M., Lanza, M., Lungu, M.: Visualizing co-change information with the evolution radar. IEEE Trans. Softw. Eng. **35**(5), 720–735 (2009)
10. de A.R. Gonçalves, J.C., Santoro, F.M., Baião, F.A.: Let me tell you a story - on how to build process models. J. UCS **17**(2), 276–295 (2011)
11. Kindler, E., Rubin, V., Schäfer, W.: Activity mining for discovering software process models. Softw. Eng. **79**, 175–180 (2006)
12. Kindler, E., Rubin, V., Schäfer, W.: Incremental workflow mining based on document versioning information. In: Li, M., Boehm, B., Osterweil, L.J. (eds.) SPW 2005. LNCS, vol. 3840, pp. 287–301. Springer, Heidelberg (2006). https://doi.org/10.1007/11608035_25
13. Leopold, H. (ed.): Natural Language in Business Process Models. LNBIP, vol. 168. Springer, Cham (2013). https://doi.org/10.1007/978-3-319-04175-9
14. Lindberg, A., Berente, N., Gaskin, J.E., Lyytinen, K.: Coordinating interdependencies in online communities: a study of an open source software project. Inf. Syst. Res. **27**(4), 751–772 (2016)
15. Mendling, J., Leopold, H., Pittke, F.: 25 challenges of semantic process modeling. Int. J. Inf. Syst. Softw. Eng. Big Co. **1**(1), 78–94 (2014)
16. Pinzger, M., Kim, S.: Guest editorial: mining software repositories. Empir. Softw. Eng. **21**(5), 2033–2034 (2016)
17. Poncin, W., Serebrenik, A., van den Brand, M.: Process mining software repositories. In: 2011 15th European Conference on Software Maintenance and Reengineering (CSMR), pp. 5–14. IEEE (2011)
18. Richetti, P.H.P., de A.R. Gonçalves, J.C., Baião, F.A., Santoro, F.M.: Analysis of knowledge-intensive processes focused on the communication perspective. In: Carmona, J., Engels, G., Kumar, A. (eds.) BPM 2017. LNCS, vol. 10445, pp. 269–285. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-65000-5_16
19. Rubin, V., Günther, C.W., van der Aalst, W.M.P., Kindler, E., van Dongen, B.F., Schäfer, W.: Process mining framework for software processes. In: Wang, Q., Pfahl, D., Raffo, D.M. (eds.) ICSP 2007. LNCS, vol. 4470, pp. 169–181. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72426-1_15
20. Ruohonen, J., Hyrynsalmi, S., Leppänen, V.: Time series trends in software evolution. J. Softw.: Evol. Process **27**(12), 990–1015 (2015)

21. Thomas, S.W., Hassan, A.E., Blostein, D.: Mining unstructured software repositories. In: Mens, T., Serebrenik, A., Cleve, A. (eds.) Evolving Software Systems, pp. 139–162. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-45398-4_5
22. Weicheng, Y., Shen, B., Xu, B.: Mining GitHub: why commit stops - exploring the relationship between developer's commit pattern and file version evolution. In: Muenchaisri, P., Rothermel, G. (eds.) APSEC 2013, Ratchathewi, Thailand, 2–5 December 2013, vol. 2, pp. 165–169. IEEE Computer Society (2013)
23. Zaidman, A., Rompaey, B.V., Demeyer, S., van Deursen, A.: Mining software repositories to study co-evolution of production & test code. In: ICST, pp. 220–229. IEEE Computer Society (2008)

A Conceptual Tool to Improve the Management of Software Defects

Nico Hillah(✉)

University of Lausanne, Lausanne, Switzerland
nico.hillah@unil.ch

Abstract. Software teams address software defect problems in a simple way: they identify them, assign them and resolve them. Nevertheless, studies have proven that having only these activities as approaches to handle a large and increasing number of software defects is inefficient. As a solution to this, we propose in this study a managerial conceptual tool for mining software defects in order to improve the management of SDs. With our proof of concept, we demonstrate how SDs mining management can be enhanced from a strategic and operational view. This is done through the precise definition of software defects' management objectives in line with the objectives of the software product owner.

Keywords: Defects mining · Software defect management · Control measures

1 Introduction

IEEE standard 1044-2009 [1] defines a defect as: *"An imperfection or deficiency in a work product where that work product does not meet its requirements or specifications and needs to be either repaired or replaced"*.

Not only the software defects (SDs) are present in the whole life cycle of a software product, but different studies also proved that 80% of the total cost of the software life cycle is associated with the management of the SDs [2]. Having this high impact on the software product, SDs management must be crucial to software teams as well as to organizations. Nowadays, the management of SDs does not only consists of identifying, assigning, and correcting them, but also in mining them. The purpose of this study is to focus on the mining aspect of SDs management.

In fact, there are different studies which propose solutions on how to mine SDs [2–4]. However, most of these existing techniques are limited to the collection, the classification, and the assignment of the SDs. In addition, these techniques do not cover the question of how to define specific SDs mining management objectives that are aligned with first, the SDs management objective, and second, with the objectives of the software product owner. This results in a poor resource allocation in mining SDs as well as in the absence of control over the SDs management in a software life cycle. In this regard, the problem we address in this paper is *how to improve and control SDs mining management in alignment with the business objectives of the software owner?*

B. Shishkov (Ed.): BMSD 2018, LNBIP 319, pp. 443–451, 2018.
https://doi.org/10.1007/978-3-319-94214-8_35

As a solution to this problem, we are proposing the use of our conceptual tool to control the mining management of the SDs. This conceptual tool is a guideline with four stages.

The paper will proceed as follows: first, we will define the software defect and its management approaches. Secondly, we will present the conceptual tool that we used to conduct the proof of concept. Finally, we will present the results and the advantages that we gained from applying it.

2 Related Works

The defects are the source of software failures and problems. Software failures are defined as "*Termination of the ability of a product to perform a required function or its inability to perform within previously specified limits*" [p. 5, 3].

In the last decade, SDs management has received a considerable amount of attention from researchers. In fact, SDs management has been the center of interest for many studies in different software studies subdomains such as software project management, software engineering and evolution [6, 7]. Due to the diversity of these studies, we group them into branches based on their interest in SDs management.

The first branch deals with questions such as how to collect and store these SDs. Studies related to this branch provide answers to questions such as how to collect SDs or which SDs characteristics must be documented [8]. These studies propose solution tools named bug-tracking systems to help collect SDs. They take the form of a central hub accessible by project managers and software developers to manage the software products. Some of these online tools are Jira [9] and Bugzilla [10].

The second branch deals with questions such as how to assign SDs to developers or how to deal with the problem of an SDs duplication [11]. The research in this branch proposes techniques and methods such as algorithms to automatically assign SDs to the right developer [12–14] and also techniques to eliminate the duplication of SDs [15].

The third branch deals with the triage and the mining of SDs. In the software life cycle, the mining of defects presents many advantages [2]. Researchers as well as practitioners in this branch have proposed schemas and taxonomies for mining SDs. The best-known schemas are (1) The Orthogonal Defect Classification (ODC) of IBM [16], the root cause analysis [1], (2) the HP Defect origins, types and modes [17] and standards like the IEEE standard 1044-2009 [5]. In the same context, they also apply data mining methods such as the Naïve Bayes Model [13] or the regression model [2] to classify SDs. In fact, the classification of defects helps the software development teams to reduce the cost of correcting SDs and helps them detect defective modules. This study is conducted as part of this last branch. In fact, our goal is to propose a conceptual tool to improve the quality of software mining results in an organization, since these results will lead to decision making concerning the quality of the software systems. The need to assure that the mining is rightly performed with defined targets will improve decision making concerning the state of software quality.

3 Presentation of the Conceptual Tool

In order to provide a means to avoid the insignificant SDs mining results to software product owners, we decided to propose this conceptual tool.

Mining SDs is a complex set of activities; it goes from selecting a technique to mine the SDs, interpreting the obtained results, to taking a decision based on the obtained results. Moreover, each software system is unique, thus needs a specific SDs mining management strategy, e.g., the SDs of the system Waterfox are not the same for Firefox, even though they have similar functionalities and purpose. Due to this complexity, inefficient SDs mining can lead to situations such as:

(1) the results obtained from the SDs mining are irrelevant for the product owner;
(2) the SDs mining is requiring much more resources than planned and software teams failed to take decisions in order to improve the quality of the software system based on the mining results;
(3) the mining goals are poorly aligned with the strategy and the objectives of SDs management and the product owner's business needs and;
(4) control and evaluation measures for obtained results are missing. To avoid these problems, we are proposing this conceptual tool to guide SDs miners wishing to improve their mining project.

Although there are similar existing conceptual tools in the literature for business domain [18], our conceptual tool is designed to target the SDs mining management field. The aim of this conceptual tool is to help software teams to define their SDs mining management strategy and to specify concrete actions to put in place this strategy in mining SDs. The conceptual tool is defined fourfold:

(1) The first step consists of defining the SDs mining management strategy in alignment with the needs of the software product owner. The strategy must be broken down into short or medium term goals to achieve. The software team, as well as the product owner, must approve these goals, e.g., a defect mining management strategy may be the improvement of the software quality by developing error-free programs for each software version released. The approval of these goals will lead to the second stage.
(2) The next step, which is the operational level, is to convert this strategy into concrete objectives. Referring to the previous example, the set of objectives will be to improve the detection of the defect modules and predict SDs.
(3) Following this, each objective must be broken down into terms of specific actions to be performed, e.g., classifying SDs according to their priorities. In addition, members of the SDs mining are responsible to implement each of these actions.
(4) Following this and depending on the actions put in place, software teams must carefully select control measures to evaluate the state of the actions, e.g., the ratio of the corrected high level prioritized SDs over the total number of SDs received.

Finally, the software team must define a list of actions to establish in order to correct cases where the set objectives have not been reached, e.g. reorganization of the process to detect SDs. Figure 1 presents the process to follow to implement the proposed conceptual tool.

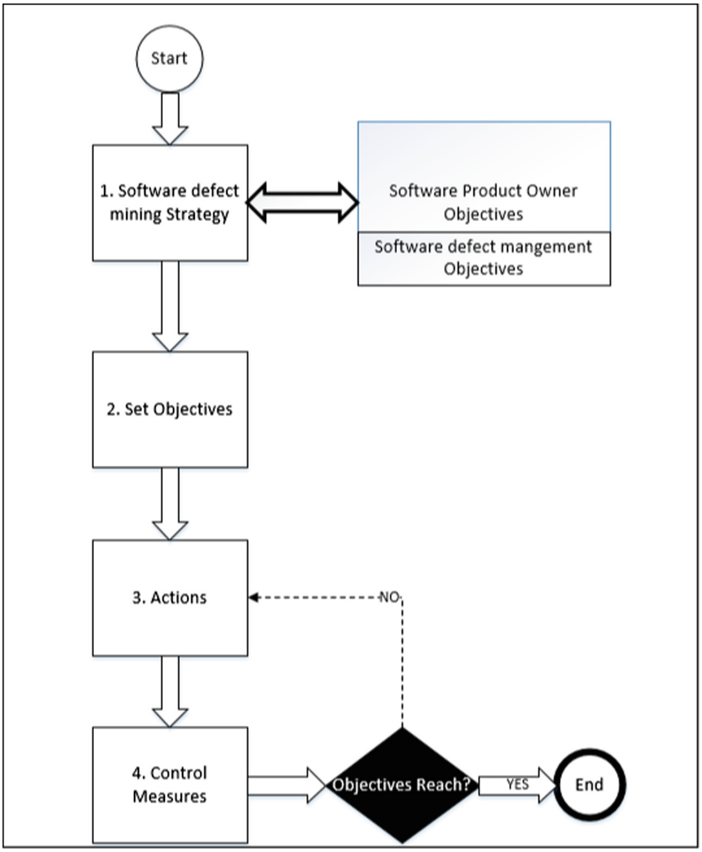


Fig. 1. The process of the conceptual tool

4 The Application of the Software Defects Managerial Conceptual Tool

In order to apply the proposed conceptual tool to improve and control the SDs mining management in practice, we decided to conduct a proof of concept of a software system that we will name system A. This system is developed using the scrum method. The owner of this system A is an education company. Its purpose is to help schools in managing the grades of their students. The overall objectives set by the owner of the system A is to have a software system with a considerable high quality, with emphasis on its availability to the users, especially during the exam period. In line with this objective, the software team objective aims to improve the assignment and the correction of the SDs.

4.1 Stage 1 and 2: Strategy Definition and Set of Objectives

In alignment with the owner's objective, our strategy would be to mine SDs in order to reduce the impact of SDs on the system to limit the system's unavailability time (stage 1). In the next step (stage 2), we cascade the defined strategy in different objectives such as to reduce the impact of defects on system A and possibly to improve the correcting process of the SDs. In the next step, we defined a set of actions to implement the objective of reducing the impact of defects on the system (stage 3). We first need to know the actual number of defects of this system A and then classify them according to their impact. To do this, we classify SDs according to their severity in order to analyze the different impact that they are having on the system availability. In the next session, we will present how we classified the SDs of system A, and then we will present the application of the final stage on this system A.

4.2 Stage 3: The Classification of SDs of System A

We analyzed the SDs of system A over a period of a year, from January 2015 to December 2015. System A has 522 SDs. We analyzed the SDs of this system by classifying them according to the defect severity attribute of IEEE 1044-2009 standards (see Table 2). This severity attribute is one of the most used attributes in SDs classification in practice [19]. The main advantage of choosing the severity attribute is the possibility for managers to identify which defect should be first corrected [19]. The IEEE's standard defines this attribute as *"The highest failure impact that the defect could (or did) cause, as determined by (from the perspective of) the organization responsible for software engineering."* [5]. There are five values of severity. They are classified from the most significant to the least significant (see Table 1).

Table 1. Severity values [5]

Attribute	Value	Definition
Severity	Blocking (B)	Testing is inhibited or suspended pending correction or identification of suitable workaround
	Critical (C)	Essential operations are unavoidably disrupted, safety is jeopardized, and security is compromised
	Major (Mj)	Essential operations are affected but can proceed
	Minor (Mn)	Nonessential operations are disrupted
	Inconsequential (I)	No significant impact on operations

4.3 Stage 4: The Section of Control Measures

There are two important aspects to consider when selecting the evaluation metrics at the fourth stage of this conceptual tool. The first one is to choose metrics based on the objective or action to evaluate, e.g. a ratio of the corrected SDs over the total number of SDs received to evaluate the SDs' correction process. The second one is to take into consideration the critical level [20] of the system being managed. This critical level can

Table 2. System A software defects classification

	Severity					
	B	C	Mj	Mn	I	Total
Jan	2	3	28	10	0	43
Feb	1	5	15	11	0	32
Mar	7	8	34	14	0	63
Apr	2	5	38	9	2	56
May	3	1	20	5	0	29
June	0	2	25	15	0	42
July	0	1	5	4	0	10
Aug	2	7	8	7	3	27
Sept	5	11	18	15	4	53
Oct	7	4	22	6	0	39
Nov	15	8	37	20	3	83
Dec	7	5	18	12	3	45
Total	**51**	**60**	**268**	**128**	**15**	**522**

relate to its business, security, and safety aspect. In addition, to determine the critical level of the system, software teams must consult and get the approval of the product owner.

To track and evaluate the success of our objective, we selected a metric as an indicator (stage 4). In this regard, we defined the SDs indicator as a ratio of the number of a type of SDs over the total SDs number received within a month. This ratio informs us about the type of defects that is problematic during the month. We define a problematic case as follows: when the number of a certain type of SDs is higher or equal to one-third of the total SDs number of a month. One-third of the total SDs is an agreed upon limited number a type of SDs may have during a month. The selection of this metric was based on system A's critical mission, which is its availability during the exam periods. We defined the indicator as follows:

$$\begin{gathered} n\,expresses\,the\,type\,of\,SDs\,to\,evaluate \\ X,\ the\,value\,of\,SDs\,and\,t\,being\,a\,time\,period \\ If\ X_n t \geq \frac{\sum (X)t}{3} \\ then\ investigate \end{gathered} \tag{1}$$

Following this, we defined a list of actions to undertake in order to correct problematic cases. These actions are:

- to conduct an investigation within the problematic type of SDs to identify mis-corrected defects;
- to reorganize the process of correcting the problematic type of SDs;
- to check other indicators such as the number of correcting defects over the total SDs within this category.

Our choice of action depends on the investigation results. In this regard, an investigation must be conducted when an indicator is reached, in order to identify the problem and to provide the right fix on time. In Fig. 2, we present the application of our indicator on SDs of system A in 2015.

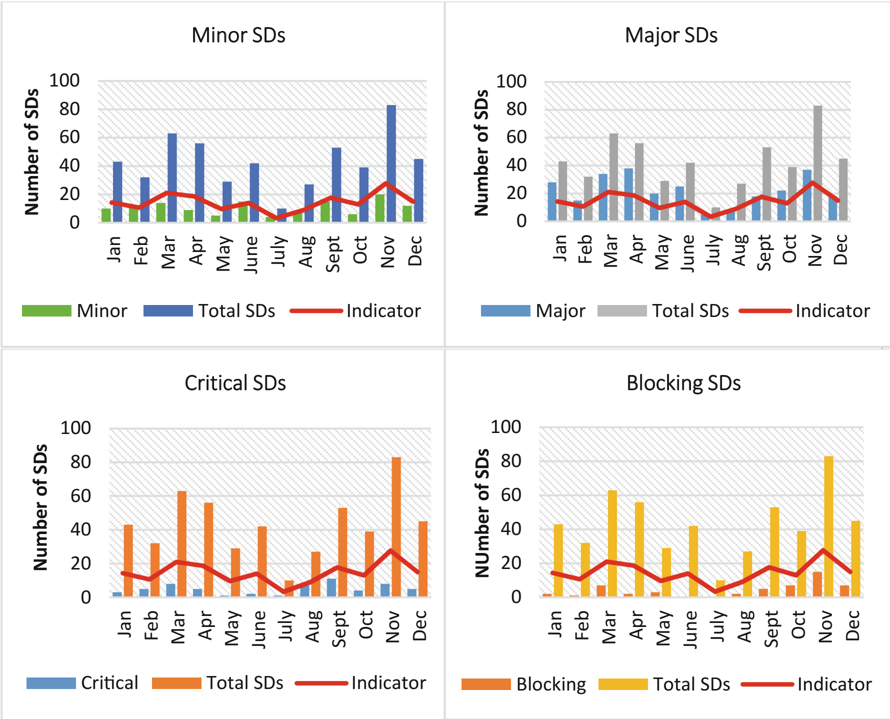


Fig. 2. Classified SDs of system A with the indicators.

5 Discussion and Contribution

The results clearly show us that the group of minor, blocking, and critical SDs management has reached the objective set in relation to the organization's objectives. In opposition, the major type of SDs failed to reach the fixed goal. In fact, from January until June, the number of the major SDs was considerably high. After investigating those months, we found that the high number of SDs of January was due to the duplication of SDs. Similarly, the month of March inherited some of the SDs of January that were incorrect. E.g., mistakes found in the names of some of the students were related to the use of ACSII format in system A and corrected in the system in January; but the same mistakes reappeared in the month of March, due to the use of an API to connect system A to an external system B. This information leads to the assignment process reorganization for the major type of SDs.

Applying this conceptual tool gives not only the insight of the SDs mining management, but also of the entire SDs management. In fact, knowing the status of each type of SDs will guide the SDs manager to focus on the problematic group of SDs and to reorganize the resource allocation in handling these groups of SDs. This improves the decision-making in managing SDs. Consequently, it improves the SDs management altogether.

The application of this conceptual tool is a manner not only to improve the management of SDs but also to align this management with the objectives of the software product owner. Its implementation is also flexible concerning the objectives set by each organization and its software department. In addition, the selection of control measures to evaluate the management must be customized for each software product.

This conceptual tool alerted us to bring the management of SDs into line. Most of all, it did not demand many interventions from us once we set it up. We propose this conceptual tool not only as contribution, but we also demonstrate its application in a real case.

6 Conclusion

Our proof of concept presents some of the advantages that software teams can gain from implementing our conceptual tool. This tool not only helps to define the precise objectives in line with the objectives of software owners in the context of SDs mining management but also guides the owners to evaluate the state of their SDs management. Indeed, the defined control measures will alert them to possible existing problems related to the management of their SDs and, therefore, of their software products. Knowing this, they will be able to take the right actions to handle the SDs. Herewith they will, on the one hand, considerably achieve the set goals, on the other hand, improve the quality of the software product, and reduce the cost of its development or maintenance. In our future work, we will provide a deep insight into the process of defining and implementing appropriate SDs management strategies by looking at the interdependence among the SDs management branches.

References

1. Leszak, M., Perry, D.E., Stoll, D.: Classification and evaluation of defects in a project retrospective. J. Syst. Softw. **61**, 173–187 (2002)
2. Rajbahadur, G.K., Wang, S., Kamei, Y., Hassan, A.E.: The impact of using regression models to build defect classifiers. In: Proceedings of the 14th International Conference on Mining Software Repositories, pp. 135–145 (2017)
3. Kagdi, H., Collard, M.L., Maletic, J.I.: A survey and taxonomy of approaches for mining software repositories in the context of software evolution. J. Softw. Maint. Evol. Res. Pract. **19**(2), 77–131 (2007)
4. Davies, S., Roper, M., Wood, M.: Comparing text-based and dependence-based approaches for determining the origins of bugs. J. Softw. Evol. Process **26**(1), 107–139 (2014). Comparing approaches for determining bug origins
5. 1044–2009 IEEE Standard Classification for Software Anomalies (2009)

6. Fischer, M., Pinzger, M., Gall, H.: Analyzing and relating bug report data for feature tracking. In: WCRE, vol. 3, p. 90 (2003)
7. Cavalcanti, Y.C., da Mota Silveira Neto, P.A., Machado, I.D.C., Vale, T.F., Almeida, E.S., Meira, S.R.D.L.: Challenges and opportunities for software change request repositories: a systematic mapping study. J. Softw. Evol. Process **26**(7), 620–653 (2014)
8. Tian, Y., Lo, D., Sun, C.: DRONE: predicting priority of reported bugs by multi-factor analysis, pp. 200–209 (2013)
9. Atlassian: Jira | Logiciel de suivi des tickets et des projets. Atlassian. https://fr.atlassian.com/software/jira. Accessed 06 Apr 2018
10. Home :: Bugzilla :: bugzilla.org. https://www.bugzilla.org/. Accessed 06 Apr 2018
11. Runeson, P., Alexandersson, M., Nyholm, O.: Detection of duplicate defect reports using natural language processing. In: Proceedings of the 29th International Conference on Software Engineering, pp. 499–510 (2007)
12. Anvik, J., Hiew, L., Murphy, G.C.: Who should fix this bug?. In: Proceedings of the 28th International Conference on Software Engineering, pp. 361–370 (2006)
13. Murphy, G., Cubranic, D.: Automatic bug triage using text categorization. In: Proceedings of the Sixteenth International Conference on Software Engineering & Knowledge Engineering (2004)
14. Aljarah, I., Banitaan, S., Abufardeh, S., Jin, W., Salem, S.: Selecting discriminating terms for bug assignment: a formal analysis, pp. 1–7 (2011)
15. da Mota Silveira Neto, P.A., Lucrédio, D., Vale, T., de Almeida, E.S., de Lemos Meira, S.R.: The bug report duplication problem: an exploratory study. Softw. Qual. J. **21**(1), 39–66 (2013)
16. Chillarege, R., et al.: Orthogonal defect classification-a concept for in-process measurements. IEEE Trans. Softw. Eng. **18**(11), 943–956 (1992)
17. Huber, J.T.: A comparison of IBM's orthogonal defect classification to Hewlett Packard's defect origins, types, and modes. Hewlett Packard Company (1999)
18. Wheelen, T., Hunger, D.: A Descriptive Model of Strategic Management. Scribd. https://www.scribd.com/document/29959620/A-Descriptive-Model-of-Strategic-Management-Wheelen-amp-Hunger. Accessed 26 Apr 2018
19. Wagner, S.: Defect classification and defect types revisited. In: Proceedings of the 2008 Workshop on Defects in Large Software Systems, pp. 39–40 (2008)
20. Rushby, J.: Critical system properties: survey and taxonomy. Reliab. Eng. Syst. Saf. **43**(2), 189–219 (1994)

Author Index