The Enterprise Engineering Series

Henderik A. Proper · Bas van Gils
Kazem Haki *Editors*

Digital Enterprises

Service-Focused, Digitally-Powered, Data-Fueled

Springer

The Enterprise Engineering Series

Enterprise Engineering is an emerging discipline for coping with the challenges (agility, adaptability, etc.) and the opportunities (new markets, new technologies, etc.) faced by contemporary enterprises, including commercial, nonprofit and governmental institutions. It is based on the paradigm that such enterprises are purposefully designed systems, and thus they can be redesigned in a systematic and controlled way. Such enterprise engineering projects typically involve architecture, design, and implementation aspects.

The Enterprise Engineering series thus explores a design-oriented approach that combines the information systems sciences and organization sciences into a new field characterized by rigorous theories and effective practices. Books in this series should critically engage the enterprise engineering paradigm, by providing sound evidence that either underpins it or that challenges its current version. To this end, two branches are distinguished: Foundations, containing theoretical elaborations and their practical applications, and Explorations, covering various approaches and experiences in the field of enterprise engineering. With this unique combination of theory and practice, the books in this series are aimed at both academic students and advanced professionals.

Hend*erik* A. Proper • Bas van Gils • Kazem Haki
Editors

Digital Enterprises

Service-Focused, Digitally-Powered, Data-Fueled

Springer

Editors
Hend*erik* A. Proper
Institute of
Information Systems Engineering
TU Wien
Vienna, Austria

Bas van Gils
Strategy Alliance
Amersfoort, the Netherlands

Kazem Haki
Campus Battelle, Bâtiment B
Haute Ecole de Gestion de Genève
Geneva, Switzerland

ISSN 1867-8920 ISSN 1867-8939 (electronic)
The Enterprise Engineering Series
ISBN 978-3-031-30213-8 ISBN 978-3-031-30214-5 (eBook)
https://doi.org/10.1007/978-3-031-30214-5

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

Paper in this product is recyclable.

Preface

The creation of this book was triggered by three, mutually amplifying, trends that drive enterprises to become "digital enterprises": the transition to the digital age, the emergence of service ecosystems, and the growing role of data as a key underlying resource. As a result of these intertwined, and mutually amplifying, trends, modern-day enterprises (including commercial businesses, government agencies, etc.) are confronted with several challenges that profoundly impact their design, from the definitions of products and services offered to their clients, via the business processes that deliver these products and services and the information systems that support these processes, to the underlying IT infrastructure.

At the same time, there are no simple answers to these challenges; there is no one-size-fits-all approach to deal with them. Therefore, in editing this book, our goal was to explore different relevant aspects of these challenges in more detail while at the same time also providing concrete suggestions for enterprises to meet the resulting challenges. As such, the target audiences of this book are both MSc- and PhD-level students who want to gain insights into key aspects of the challenges confronting digital enterprises, as well as enterprise architects and information managers working in enterprises that are on the road to become a digital enterprise.

In line with this, we have brought together contributions covering four key perspectives, covered in four parts of the book:

- Part I: Involving experience reports on the way enterprises currently already need to meet the discussed challenges
- Part II: Looks at the need for a new *design logic*, in terms of a need for new ways of thinking regarding the design of enterprises
- Part III: Is concerned with the coordination needed among different stakeholders of the ensuing (continuous) transformations
- Part IV: Reflects on the ensuing consequences for *enterprise modeling*, as used to capture both the current affairs of an enterprise and design/study its possible future affairs.

Finally, we would also like to thank our respective employers at the time of authoring and editing this book: the ITIS department of the Luxembourg Institute of Science and Technology, Luxembourg; Strategy Alliance, the Netherlands; and the Institute of Information Management of the University of St. Gallen, Switzerland.

Vienna, Austria — Hend*erik* A. Proper
Amersfoort, the Netherlands — Bas van Gils
Geneva, Switzerland — Kazem Haki

Abstract

Ever since the Industrial Revolution, change has often been driven by the introduction of new technology. It seems that the organization that is best at *leveraging* technology wins in the marketplace—meaning that keeping up (or even ahead) of developments has become a crucial capability for modern organizations. The word *leveraging* is emphasized to stress the fact that technology is not a differentiator per se. Only when it is used successfully does it have any effect. Some claim that the *people* factor is actually key (Kane et al., 2019). The plethora of changes that the digital transformation has brought about, and the many more that we are not even aware of yet or have not even been thought of yet, provides organizations with deep and fundamental challenges. *How to excel as an enterprise, while everything is changing constantly?* There are hardly any securities left; traditional business models are continuously challenged by digitally inspired and empowered startups.

The creation of this book was triggered by three, mutually amplifying, fundamental trends driving enterprises to change: the transition to the digital age, the emergence of service ecosystems, and the growing role of data as a key underlying resource. As a result of these trends, modern-day enterprises are confronted with several challenges. These challenges impact the “design” of these enterprises, from the definitions of products and services offered to their clients, via the business processes that deliver these products and services and the information systems that support these processes, to the underlying IT infrastructure.

When using the term *enterprise*, we specifically do not only refer to commercial businesses, but rather more broadly to any *purposeful undertaking by a socio-technical system*. This indeed includes commercial businesses, but also includes government agencies, NGOs, factories, mobility networks, logistics networks, etc.

The aim of this book is, therefore, to explore different relevant aspects in more detail while at the same time also providing concrete suggestions for enterprises to meet the resulting challenges. In line with this, this book brings together contributions covering four key perspectives:

1. *Experience reports* on how enterprises deal with these trends in practice
2. The need for a new *design logic*

3. The consequences for *architectural coordination* of the needed transformations
4. The ensuing consequences for *enterprise modeling*

Each of these perspectives will be covered in a separate part of this book, containing stand-alone parts with contributions by several authors. Each of the parts contains a concluding chapter reflecting on the key insights provided by the different contributions.

Contents

1 Introduction 1
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

Part I Experience Reports

2 Introduction 11
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

3 Investigating 5,140 Digital Transformation Projects 13
Hans Mulder, Jim Johnson, Klaas Meijer, and Jim Crear

4 From Product-Oriented Insurance Company to Customer-Centric Service Provider 33
Ralph van Vliet and Joppe Ter Meer

5 How Blockchain Technology Affects the Performance of Financial Services 53
Martin Smits, Hans Weigand, Joost de Kruijff, and Jerom de Valk

6 Analysis and Design of Digital Value Co-creation Networks: Insights from Digital Platforms 73
Michael Blaschke

7 ArchiMate Extension to Value Co-creation: The Smart Airport Case Study 105
Christophe Feltus, Hend*erik* A. Proper, Andreas Metzger, and Juan Francisco Garcia López

8 Conclusion 135
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

Part II The Need for a New Design Logic

9 Introduction 139
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

10 Digital Transformation Requires a New Organizational Logic 141
Janne J. Korhonen

11 Organizational Identity and Self-Awareness: Creating Convergence Between Enterprise Engineering and Organizational Design 157
Rodrigo Magalhães, José Tribolet, and Marielba Zacarias

12 Conclusion 177
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

Part III An Architectural Coordination Perspective

13 Introduction 181
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

14 IT Architecture Principles: Foundation for Digital Transformation? 183
Michiel Borgers and Frank Harmsen

15 The Need for New Architectural Truths 201
Marlies van Steenbergen

16 Presentation and Validation of an Integrative Approach for Digital Transformation Evaluation 211
Tamara Högler, Johan Versendaal, and Ronald Batenburg

17 Conclusion 231
Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

Part IV An Enterprise Modeling Perspective

18 Introduction 235
Bas van Gils, Kazem Haki, and Hend*erik* A. Proper

19 Enterprise Cartography: From Theory to Practice 237
Pedro Sousa, José Tribolet, and Sérgio Guerreiro

20 A Conceptual Model for Expressing the Evolution of Enterprise Architecture Models 265
Nuno Silva, Pedro Sousa, and Miguel Mira da Silva

21 Next-Generation Enterprise Modeling 279
Bas van Gils and Hend*erik* A. Proper

22 Conclusion 307
Bas van Gils, Kazem Haki, and Hend*erik* A. Proper

Part V Epilogue

23 Final Conclusions and Outlook 311
Hend*erik* A. Proper, Bas van Gils, and Kazem Haki

Bibliography 315

About the Editors

Hend*erik* A. Proper Erik to friends, is Full Professor in Enterprise and Process Engineering at the TU Wien (Vienna University of Technology), in Austria. His general research interest concerns the foundations and applications of domain modeling in an enterprise context. He has regularly lectured within different MSc programs offered by the University of Luxembourg (LU), the University of Lorraine (FR), TU Wien (AT), the University of Namur (BE), Antwerp Management School (BE), and the TIAS Management School (NL).

Erik received his master's degree from the University of Nijmegen, the Netherlands, in May 1990 and received his PhD (with distinction) from the same university in April 1994. After receiving his PhD, Erik became Senior Research Fellow at the Computer Science Department of the University of Queensland, Brisbane, Australia. In 1995, he became Lecturer at the School of Information Systems from the Queensland University of Technology, Brisbane, Australia.

From 1997 to 2001, Erik worked in industry: first as a consultant at Origin, Amsterdam, the Netherlands, and later as a research consultant and principal scientist at the Ordina Institute for Research and Innovation, Gouda, the Netherlands.

In June 2001, Erik returned to academia, to become Professor at the Radboud University Nijmegen. In January of 2008, he went back to combining industry and academia, by combining his professorship with consulting and innovation at Capgemini, with the aim of more tightly combining his theoretical and practical work.

In May 2010, Erik moved to the Luxembourg Institute of Science and Technology as FNR PEARL Laureate and Senior Research Manager. Initially, he continued combining this with his part-time chair at the Radboud University Nijmegen in the Netherlands, to then in June 2017 following this up with an adjunct chair in Data and Knowledge Engineering at the University of Luxembourg.

Finally, in January 2023, Erik moved to the TU Wien (Vienna University of Technology) to take up a Full Professorship in Enterprise and Process Engineering.

Bas van Gils is a driven and experienced consultant, trainer, and researcher. He is a managing partner at Strategy Alliance, which is also the home for his consulting activities. The last few years, he has helped professionals and organizations in realizing their digital aspirations: from strategy to realization. The core disciplines in his work are (1) digital transformation, (2) enterprise architecture, and (3) data management. His motto is: in an increasingly digital world, you have to put the people first. He has worked in different industries, both in Europe and the United States as a teacher and consultant.

He is also Professor at Antwerp Management School, which is where his academic work is presently rooted. He is involved in courses with a broad range of topics, including architecture and data management. Bas frequently publishes both academic and professional papers and books. He is a frequent speaker at conferences.

It is increasingly apparent that *data* is one of the key *assets* for many organizations. The corollary is that *data management* is becoming more and more important. Bas has helped many organizations in their journey to become more *data driven*. These projects cover a wide spectrum of topics, ranging from data management processes, organization structures (CDO Office), and training of professionals. The key is to get *just enough grip on data* in order to *create value with data*. Recently, Bas has published his experiences in this field with a book: *Data Management: A Gentle Introduction*. This book covers both the theory and practice of this exciting field.

Kazem Haki is Associate Professor at the Geneva School of Business Administration (HES-SO, HEG Genève). He obtained his habilitation qualification (Venia Legendi) in information systems (IS) from the University of St. Gallen, Switzerland, and a doctorate's degree in the same field from the University of Lausanne (HEC Lausanne), Switzerland. His research interests include the dynamics and evolution of socio-technical phenomena such as digital platforms, platform ecosystems, and enterprise architecture. His studies have been published in leading IS journals such as *MIS Quarterly* (FT50 journal), *Journal of the Association for Information Systems*, *European Journal of Information Systems*, *Business & Information Systems Engineering*, and *MIS Quarterly Executive*, among others. His studies also appear in the proceedings of flagship IS conferences such as International Conference on Information Systems and European Conference on Information Systems.

About the Authors

Andreas Metzger is Adjunct Professor at the University of Duisburg-Essen and heads the group "Adaptive Systems" at paluno, The Ruhr Institute for Software Technology. His current research interests include the use of machine learning in software engineering and business process management. He received his diploma and his PhD in computer science from the Technical University of Kaiserslautern in 1998 and 2004, respectively.

Andreas is Steering Committee Vice Chair of the European Technology Platform NESSI (The Networked European Software and Services Initiative) and was Deputy General Secretary of BDVA (The Big Data Value Association) from 2015 to 2021. Among other leadership roles in EU projects, he was the technical coordinator of the Big Data Value PPP lighthouse project TransformingTransport and chief architect of the Future Internet projects FInest and FIspace.

He has co-authored over 100 peer-reviewed papers, articles, and book chapters. He was PC chair of several international conference tracks and workshops, such as the Euromicro Conference on Software Engineering and Advanced Applications (SEAA), and acted as program committee member for numerous international conferences and workshops, such as the European Software Engineering Conference (ESEC/FSE), the International Conference on Services Computing (SCC), and the International Conference on Information Systems (ICIS).

Christophe Feltus graduated as an electromechanics engineer from the Institut Supérieur Industriel des Art et Métiers Pierrard (Belgium) and doctor of science (computer science) from the University of Namur (Belgium). He worked for several years in private companies as Production Head at Pfizer SA in Jette, Project Coordinator at Nizet Entreprise in Louvain-la-Neuve, and Assessor for the Civil Belgium Aviation Administration in Brussels, Belgium. Then, he joined Luxembourg Institute of Science and Technology (LIST—formerly the Public Research Centre Henri Tudor) in the Grand-Duchy of Luxembourg as Senior Engineer in 1999 to work in the field of service science and innovation. There he has taken part in projects related to IT security, IT governance, enterprise architecture modelling, optimization, and data science. During his research, Christophe has actively contributed to the writing of scientific articles for journals, conferences, and book chapters, to the organization of scientific conferences and workshops, and to the supervision of master's and doctoral students.

Frank Harmsen has an MSc in Computer Science from Radboud University Nijmegen and a PhD in Computer Science and Management Science from the University of Twente. In 1995, he was a visiting researcher in the area of Situational Method Engineering at Tokyo Institute of Technology. He has been working as a consultant, project and program manager, and director since 1996. Since 2008, he has been Endowed Professor at Maastricht University (School of Business and Economics), in the digital transformation space.

Hans Mulder is an entrepreneur and founder of Viagroep nv, Professor at Antwerp Management School (AMS), and Lecturer at the Police Academy of the Netherlands and Nyenrode Business Universiteit. Hans holds a PhD from Delft University of Technology, published over 100 articles in specialist journals and international magazines, and is (co-)author of several books such as *Enterprise Ontology* and *Éénvoud in Complexiteit*. Hans is regularly engaged as an IT expert when conflicts between companies need to be resolved in or out of court, such as participating in arbitration, mediation, and expert reports.

Hans Weigand is Associate Professor in the Information Management Group of the Tilburg University School of Economics and Management (TiSEM). In general, he is interested in how organizational and technical aspects of information systems interact, taking a language/action approach. His research topics include business ontologies, shared ledger systems, design science research, and smart auditing.

Janne J. Korhonen is Adjunct Postdoctoral Researcher at the Department of Industrial Engineering and Management, Aalto University School of Science. He received his D.Sc. (Tech) degree in Enterprise Systems at Aalto University in 2018, M.Sc. (Tech) degree in Computer Science at Helsinki University of Technology in 2003, and B.A. degree in Cognitive Science at University of Helsinki in 2007. His research interests include organization design, organizational change and transformation, and organizational capabilities. Dr. Korhonen is also an independent organization design consultant, enterprise architect, coach, and advisor, specialized in diagnostic assessment and development of organizational and leadership capabilities. His multi-sector business and IT consulting experience spans 25 years.

Jerom de Valk is presently Digital Trust Manager at PwC in the Netherlands.

Jim Crear is Chief Information Officer and Architect for The Standish Group's Master Adjudicator. He is an experienced executive with over 50 years of experience in the field of information technology. He has had a diverse career leading global

IT organizations including academia, biomedical, manufacturing, and software companies. For over 30 years, Jim was intimately involved in all aspects of the CHAOS research and publications. As The Standish Group's Master Adjudicator and manager of the adjudication department, Jim has personally reviewed tens of thousands of project profiles that make up the over 25 years of project information in our CHAOS database. Jim has been a stalwart of contributions to over a dozen books and thousands of research reports.

Jim Johnson is the founder of The Standish Group International Inc. in 1985 and is the current chairman and chief dreamer. The Standish Group is the leading information technology research advisory firm specializing in the analysis of project and system failures. Jim has been involved in the computer industry professionally for almost 60 years and is the author of over a dozen books and thousands of research papers and articles. Jim is a frequent keynote speaker at conferences and educational events. He has a combination of technical, marketing, and research achievements focused on mission-critical applications and technology. As a former professor at Antwerp Management School, Jim is the world's leading authority on software project performance.

Johan Versendaal has been Professor at HU University of Applied Sciences Utrecht since 2008. In 2014, he also became Professor of E-business at the Open University of the Netherlands. In these roles, he co-authored scientific papers in the field of business/IT alignment, business process and rules management, and digital transformation and data science. His current research interest includes the ethical perspective of the design and implementation of ICT, which is also reflected in the name of the research group at HU: digital ethics.

Johan studied computer science at Delft University of Technology and obtained his PhD on design methods and software architectures for information systems. From 1991 to 2002, he worked in business for the Bureau for System Development (currently Atos) and Baan (currently Infor). From 2002 to 2012, he also worked as Assistant Professor at Utrecht University.

Joost de Kruijff is an independent consultant and PhD student at Tilburg University. His research focus is on commitment-based smart contract languages.

Joppe Ter Meer is Senior Management Consultant at the Dutch consulting firm Novius/RHDHV Digital. Joppe Ter Meer is specialized in digital transformations, leading organizations through complex transformations, guiding innovation processes, and bridging business and IT. Furthermore, Joppe has expertise in strategy to execution, enterprise architecture, and project and program management.

Joppe performs well in complex projects without a well-defined structure. He works with a close eye for the people behind the content. Currently, Joppe operates as Digital Transformation Manager for ABN AMRO.

José Tribolet is Emeritus Distinguished Professor of Instituto Superior Técnico, University of Lisbon, Portugal, and founder and ex-president of INESC—Instituto de Engenharia de Sistemas e Computadores (1970–2020). He is Researcher Emeritus at the Information and Decision Support Systems Group at INESC-ID.

His main academic interests are in enterprise architecture, enterprise engineering, enterprise governance, and enterprise transformation. He was General Chairman of the EEWC Conference in 2014 (Funchal, Portugal) and in 2019 (Lisbon, Portugal). He was until December 2016 an associate editor of the journal BISE—*Business & Information Systems Engineering*. He was until July 2020 co-editor and is now co-editor emeritus of the Springer Enterprise Engineering book series.

Juan Francisco Garcia López is Strategy and Innovation Manager in Airport Department at Indra, leading the participation of Indra Airport Department in R&D projects (7th Framework Program, Horizon 2020, SESAR).

Juan Francisco holds a master's degree in computer science and engineering from the Universidad Politecnica de Madrid and a Project Management Professional (PMP) certification.

He is passionate about airport operations and one of his main interests is to contribute to the improvement of flight operations and passenger experience in airports through the introduction of innovative concepts and systems.

During a career of more than 20 years in the airport systems engineering field, Juan Francisco started developing airport systems, continued leading projects as project manager, and is now contributing to the definition and implementation of new operational concepts and airport systems in some European programs. As part of his responsibilities, Juan Francisco is working on the alignment of the prototypes developed in those innovation programs with the road map of airport products in Indra, ensuring the outcomes of these R&D projects are finally deployed in real airport environments.

Klaas Meijer is an experienced IT project and program manager who worked for global IT organizations as well as governmental institutes over the last 35 years. He has led multiple kinds of projects and programs mainly at utilities, insurance firms, and governmental departments with international teams.

Marielba Zacarias has devoted her career to the information systems field. From 1982 to 2000, she worked as a programmer, as a systems analyst, and later as a software project manager in Venezuela. In 2001, she moved to Portugal where she started lecturing and research activities at Algarve University in Faro, Portugal. In 2008, she completed her PhD degree in Informatics and Computers Engineering at the Instituto Superior Técnico of Lisbon. She is currently Assistant Professor at

Algarve University and Researcher at the Research Centre for Tourism, Sustainability and Well-Being (CinTurs). Her main academic interests are in enterprise architecture, enterprise engineering, enterprise ontologies, and interaction design, having several scientific papers published in these areas.

Marlies van Steenbergen was a principal consultant of enterprise architecture at Sogeti Netherlands for more than 20 years. In this capacity, she helped many enterprises to professionalize their enterprise architecture function. Before that, she worked as a researcher and enterprise architect at KPN Telecom. In 2011, she did her PhD on the maturity and effectiveness of enterprise architecture at Utrecht University. She is a frequent lecturer and publisher in both academic and professional outlets. She is a co-author of various books on enterprise architecture and co-founder of the dynamic architecture (DYA) approach. In 2018, she received the NAF medal for her meaningful contribution to the IT architecture community in the Netherlands. Since 2017, she is Professor of digital ethics at Utrecht University of Applied Science.

Martin Smits is Associate Professor of Information Systems and Management at the Tilburg School of Economics and Management (TiSEM) of Tilburg University. He is a member of the CentER (Center for Research in Economics and Business) and Tranzo (the Center for Research in Care and Wellbeing at the Tilburg School of Social and Behavioral Sciences (TSB)). His research focuses on the impact of IT on performance of business processes in organizations and business networks. Topics include business process analysis, IT strategy, and innovation. He is particularly interested in electronic markets, intermediation, and market performance. He designs and analyzes process models, system dynamics models, and business models, in particular to analyze and predict the impact of technology innovations in health care (e-health), government (e-government), and financial services (fintech).

Michael Blaschke is Management Consultant with international work experience at Kearney Digital, Accenture Strategy, and SAP. Michael advises clients globally on technology strategy, architecture, and transformation. He is a member of the Metaverse core team at Kearney Digital and was a member of the Digital Asset core team at Accenture. He serves as a co-host on the podcast Bitcoin, Fiat & Rock'n'Roll. Michael holds a bachelor's degree in information systems, a master's degree in computer science, and a PhD in information systems from the University of St. Gallen (HSG) and the London School of Economics (LSE). In foundational research supported by the Swiss National Science Foundation (SNSF), he investigates digital platforms to publish in the proceedings of key international conferences and journals.

Michiel Borgers finished his study in Computer Science at the TU Delft in 1997. After advising organizations for 12 years, he became Chief Architect at the Dutch Tax and Customs Administration in 2008. Between 2017 and 2022, Michiel was the CIO of Maastricht University. Since August 2022, he is the IT Director of the Dutch Ministry of Defense. Michiel received his PhD in 2021. Over the years, he published more than 40 articles, including many scientific papers. Michiel was a visiting lecturer at several universities and colleges and supervisor of several master students. Since 2022, Michiel has joined the MUSTS research program as a guest researcher at Maastricht University.

Miguel Mira da Silva is Full Professor of information systems at the Instituto Superior Técnico (University of Lisbon, Portugal) and the Coordinator of the Digital Transformation Group at the INOV research institute. Miguel also coordinates an online master's degree in enterprise information systems, as well as several training programs and courses. He graduated and received an MSc degree in computer engineering from the Instituto Superior Técnico, a PhD in computing science from the University of Glasgow, a *Sloan Fellowship* master degree in management from the London Business School, and more recently a habilitation (agregação) in information systems from the Instituto Superior Técnico.

Nuno Silva is an enterprise architect at NOS SGPS and a business architecture contributor at the TM Forum. Nuno graduated and received an MSc and PhD in Information Systems and Computing Engineering from the Instituto Superior Técnico (University of Lisbon, Portugal). Nuno has been involved in information systems research and practice for 8 years. Both his research and professional interests are in Agile methodologies, IT strategy, and business/IT alignment through the adoption, use, and maintenance of enterprise architectures. Throughout his career, Nuno has been involved in several projects focusing on Agile digital transformation and helping define IT strategies.

Pedro Sousa is Associate Professor in the Department of Computer Science and Engineering, Instituto Superior Técnico, Lisbon University, where he created and has been responsible for the courses on enterprise architecture (EA) and business process management (BPM) and engineering.

He is a senior consultant in the domain of EA, BPM, and IT governance for over 25 years. He has worked in large companies across multiple industries, both in the public and private sector, in several countries in Europe, Africa, and South America.

He is also the author and responsible for the Atlas Enterprise Cartography Tool to support low-effort architecture creation and maintenance, from strategic and capabilities to business processes and services down to information and IT architectures.

Ralph van Vliet At the time of production of the respective chapter, Ralph van Vliet was a business consultant at the Dutch consulting firm Novius/RHDHV Digital. Ralph is specialized in organizational transformation and design, enterprise architecture, and data management.

As a driven professional, Ralph receives energy from solving customers' challenges on the interface of business and IT. Ralph holds an entrepreneurial and analytical mindset and is social, creative, and loyal by nature. Ralph is fascinated and intrigued by emerging and disruptive innovations in the IT domain. Currently, Ralph operates as business consultant for Share a Byte.

Rodrigo Magalhães is a retired professor of management, currently affiliated as a researcher to INESC, Lisbon, Portugal. He was Dean of Business Management at the Kuwait College of Science and Technology and formerly a faculty member at the School of Management and Economics at the Catholic University of Portugal. He also held a visiting position at the Department of Computer Science and Engineering, Instituto Superior Tecnico, University of Lisbon. Prof. Magalhães was Co-Editor-in-Chief of the Springer journal *Organizational Design and Enterprise Engineering* and has an extensive publishing record in the areas of design management, information systems management, organizational change, knowledge management, organization learning, business process management, and e-learning.

Ronald Batenburg is trained as a sociologist and obtained his masters at Utrecht University and his PhD at Groningen University, the Netherlands. How job structures evolve within economies, sectors, and organizations and how these determine the quality of work of professionals are the central questions he worked on at different universities and research institutes. As of 2009, he is a program coordinator at the Netherlands Institute for Health Services Research (Nivel) and specialized in the area of human resources for health. The main projects he leads concern the labor market, work, and education in health. This includes how institutional reforms, organizational development, task delegation, and substitution change job structures and the work of healthcare professionals. From a policy perspective, he evaluates how healthcare policy and management—in particular (strategic) health workforce planning—can improve the alignment between healthcare demands and the critical supply of human resources for health. Since 2017, Ronald is Endowed Professor in health workforce and organization studies at Radboud University Nijmegen, Department of Sociology.

Sérgio Guerreiro is Assistant Professor at the Instituto Superior Técnico, Universidade de Lisboa, Portugal. Concluded the PhD degree in Information Systems and Computer Engineering at the Instituto Superior Técnico/Technical University of Lisbon (IST/UTL) in 2012. Concluded the MSc degree in 2003 and the BSc degree in 1999. Both degrees also in Information Systems and Computer Engineering at IST/UTL.

Tamara Högler has a diploma in geo-ecology from the KIT Karlsruhe Institute of Technology and obtained her PhD in economics at the Open University of the Netherlands. Throughout her career, Tamara was involved in or coordinated national and European R&D as well as industrial projects with a focus on digitalization and innovation support for SMEs. Based on this experience, Tamara works as an independent consultant, advising SMEs on digital transformation and supporting them in acquainting and implementing R&D projects, mainly in the broad field of IT.

Between 2001 and the time of production of the respective chapter, Tamara worked for several acknowledged German research institutions, companies, and a non-profit organization in research or leading positions.

Chapter 1
Introduction

Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

1.1 Motivation for This Book

Ever since the industrial revolution, change has often been driven by the introduction of new technology. It seems that the organization that is best at *leveraging* technology wins in the marketplace—meaning that keeping up (or even ahead) of developments has become a crucial capability for modern organizations. The word *leveraging* is emphasized to stress the fact that technology is not a differentiator per se. Only when it is used successfully does it have any effect. Some claim that the *people* factor is actually key (Kane et al., 2019). The plethora of changes that the digital transformation has brought about, and the many more that we are not even aware of yet, or have not even been thought of yet, provides organizations with deep and fundamental challenges. *How to excel as an enterprise, while everything is changing constantly?* There are hardly any securities left; traditional business models are continuously challenged by digitally inspired and empowered startups.

The creation of this book was triggered by three, mutually amplifying, fundamental trends driving enterprises to change: The transition to the digital age The emergence of service ecosystems The growing role of data as a key underlying resource As a result of these trends, modern-day enterprises are confronted with several challenges. These challenges impact the "design" of these enterprises, from

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_1

the definitions of products and services offered to their clients, via the business processes that deliver these products and services, and the information systems that support these processes, to the underlying IT infrastructure.

When using the term *enterprise*, we specifically do not only refer to commercial businesses, but rather more broadly to any *purposeful undertaking by a socio-technical system*. This indeed includes commercial businesses but also includes government agencies, NGOs, factories, mobility networks, logistics networks, etc.

In the remainder of this chapter, we will discuss the three trends in more detail (Sects. 1.2–1.4), as well as the challenges they pose on enterprises. In doing so, we will also conclude that these challenges are manifold and highly intertwined and that, as a result, there is no one-size-fits-all approach to deal with them.

The aim of this book is, therefore, to explore different relevant aspects in more detail while at the same time also providing concrete suggestions for enterprises to meet the resulting challenges. In line with this, this book brings together contributions covering four key perspectives:

1. *Experience reports* on how enterprises deal with these trends in practice
2. The need for a new *design logic*
3. The consequences for *architectural coordination* of the needed transformations
4. The ensuing consequences for *enterprise modeling*

Each of these perspectives will be covered in a separate part of this book, containing stand-alone chapters with contributions by several authors. Each of the parts contains a concluding section reflecting on the key insights provided by the different contributions. An overarching summary of the insights provided in this book as a whole, together with suggestions for future research efforts, is provided in a final concluding section.

1.2 The Transition to the Digital Age

Periods of great change in human society have often been driven by the advent of disruptive technologies, such as the introduction of the printing press, the steam engine, cars, or the telephone. The introduction of each of such technologies has brought about fundamental change in society's culture and economy.

Information technologies (IT), including telecommunication technology, mobile computing, pervasive computing, cloud computing, big data, artificial intelligence (AI), robotics, social media, etc., provide the next wave of disruptive technologies that take our society from the industrial age to the digital age, where people, information, artifacts, and knowledge converge, collaborating and innovating at an unimagined intensity.

In the western world, digital has already become the new normal, both in our daily lives and at our work (Negroponte, 1996). It seems as if every aspect of our lives is being impacted on by this transition. Letters are all but replaced by e-mail, books are digitized, while we track our health through wearable technology (leading to the so-called quantified self Swan, 2012). The same holds for enterprises.

The emergence of IT initially enabled enterprises to automate their information processing activities. As such, the initial use of IT aimed to amplify the cognitive abilities of humans (Magalhães and Proper, 2017), which was only a foreboder of things to come. Taylor (1982) already provided interesting insights into the possible far-reaching consequences on communication and coordination in enterprises, in particular in shaping the administrative structures of enterprises.

Soon after enterprises started to use IT to automate their information processing activities, IT also started to be used to steer and control machinery. This enabled us to amplify our human abilities not only in a cognitive sense but also in a physical sense, resulting in the automation of manual processes, e.g., using computer-integrated manufacturing and robotics (Scheer, 1988; Kosanke et al., 1999).

The ongoing miniaturization of hardware, the integration of IT and communication technologies, the networking of IT on a global scale (i.e., the Internet), the advent of mobile computing, and the introduction of different networked sensors/actuators (e.g., the Internet of Things Weber and Weber, 2010) also enabled us to amplify our communication/dialoguing capabilities as well as (remote) sensing/actuating capabilities.

Recent developments in AI, where traditional symbolical approaches (e.g., *logic* and *rule-based*) have been complemented with statistical approaches, made possible by the availability of large amounts of (training) data (see Sect. 1.4), have enabled us to not just amplify our abilities but even to completely take over (and improve) human roles and activities. In our homes, AI-enabled thermostats enable us to continuously optimize our energy consumption. When on the road, AI-powered apps on our mobile phones help us find the best way to reach our destination. When working in a multilingual environment, AI techniques help us translate documents. In healthcare, AI-based solutions aid doctors in producing better diagnoses. When mining resources in hard to reach places, AI comes to the aid as well, to, e.g., drive autonomous vehicles (Marr, 2018). And, as humanity ventures out to new frontiers, such as deep space, or the bottom of the ocean, AI can play a crucial role by taking over tasks that would be too dangerous for humans to perform (AZoRobotics, 2018; O'Kane, 2016).

As a result, we are now confronted with a new *socio-digital-physical*[1] reality in which social actors (humans) interact closely with digital actors while jointly controlling and managing physical machinery (cars, smart buildings, machines, heating systems, lighting systems, traffic control, factories, etc.). According to a recent publication (Brown, 2017), we should even prepare for new forms of diversity in the workforce, where humans should learn to collaborate better with non-humans (e.g., AI-based agents, robots, etc.).

[1] Even though it has become commonplace to use the word "cyber" as a general prefix to refer to computer-controlled activities, we prefer the word digital. The prefix "cyber" originates from cybernetics, which represents the art of steermanship (Ashby, 1956), which was intended as a generic notion and not to be solely linked to the use of computers for steering and control.

Where IT originally was a mere supportive tool for administrative purposes, it is safe to say that in the digital age, IT has established itself as being an integral part of an enterprise's primary processes and has quite often become an integral part of their business models as well. The rapid evolution of IT brings an abundance of new opportunities to enterprises (Capgemini, 2009; Tapscott, 1996; Hagel III and Armstrong, 1997). Services offered by enterprises are increasingly delivered by way of digital channels (Horan, 2000). Technology becomes part of almost everything and most processes have become IT reliant, if not fully automated. The discussion of business-IT alignment (Henderson and Venkatraman, 1993) is subsumed by the broader issue of business-IT fusion (Op 't Land et al., 2008a).

Companies, such as Amazon, Airbnb, Uber, Netflix, Spotify, Bitcoin, etc., provide clear examples of the latter. In addition, the CEO of a major bank can even be quoted as stating "We want to be a tech company with a banking license" (Hamers, 2017), illustrating the point that traditional business models are also shifting toward digital business models.

From an entrepreneurial perspective, the digital age certainly offers many new possibilities to optimize existing processes and services while also offering ample opportunities for new product and services. At the same time, the transition to the digital age also raises fundamental concerns regarding different social and economical aspects, including ethics, security, privacy, and trust.

Where the key driver in the industrial age was the fact that factories became "(steam) engine-powered," the key driver of the digital age is the fact that enterprises (and their enterprises) become "digitally powered."

1.3 The Emergence of Service Ecosystems

The *globalization* of our economy and society has removed physical, economical, cultural, and political barriers, while decisions are no longer based on geographical location and their inherent limitations (Friedman, 2005). As a result, most enterprises are forced to continuously position themselves on a global marketplace. Enterprises can no longer "hide" within the boundaries of their own nation or municipality. The differentiation of an enterprise's services and products needs to be engaged at a global scale.

Consider, as an example, traditional bookstores. These bookstores had to compete with Amazon and its likes, if they want to or not. They can only do so by either becoming a direct competitor of Amazon or by strengthening their differentiators in terms of physical proximity to clients, expert advice, being able to "touch and browse before buying," or bundling their service with complimentary services such as book presentations by authors, a reader's café, etc.

Globalization requires enterprises to increasingly focus on the ability to combine their own core competences with those of others, in order to provide/deliver distinguishing products and/or services. Traditional fixed enterprise structures are being replaced by more dynamic networked structures (Hagel III and Armstrong, 1997;

Hagel III and Singer, 1999; Galbraith, 2000; Tapscott et al., 2000; Malone, 2004; Camarinha-Matos and Afsarmanesh, 2004; Friedman, 2005; Umar, 2005), also blurring borderlines between existing enterprises within the same value chain/web. For instance, Friedman (2005) states that businesses are not formed merely based on the core competencies they have, but rather on their ability to provide services by clever combinations of outsourcing and renting through service providers around the globe.

In parallel to the globalization of the economy, we have witnessed a transition from a goods-based economy to an (increasingly) services-based economy. Consumers, clients, and citizens do not “just” expect a product anymore. They expect integrated service offerings that are updated at the same pace as their own needs. For instance, in the airline industry, jet turbine manufacturers used to follow a classical goods-dominant logic by selling turbines to airlines. However, since airlines are not interested in *owning* turbines, but rather in the realization of *airtime*, manufacturers nowadays sell airtime to airlines instead of jet turbines. Similarly, producers of professional power drilling equipment are now selling holes, i.e., drilling equipment *utilization*, instead of selling drilling equipment.

In the early stages of the transition to the service economy, services were quite often still treated as goods in the sense that the economic exchange focused on the transaction between a goods/service provider and a consumer. For example, when one would like to travel from A to B by train, one would need to purchase some form of a ticket, as if it (being transported from A to B) is a good. More fundamentally, however, the service I would really need is a *mobility* service, in which the transportation means from A to B is more dynamically adjusted to my needs, traffic situations, etc. To this end, however, one would also need to provide (some level of) insight into my preferences during/around the planned trip. For instance, "would one like to be able to work during this trip?", or "would one need to first drop off the children close to A before travelling to B?", et cetera.

In the further development of the service economy, the very notion of *economic exchange*, core to the economy, can now be seen to shift from following a so-called goods-centered dominant logic to a *service-centered dominant logic* (Vargo and Lusch, 2004; Lusch and Vargo, 2006; Vargo and Lusch, 2008; Grönroos and Ravald, 2011; Lusch and Nambisan, 2015; Vargo and Lusch, 2016). While the former focuses on tangible resources to produce goods and embeds value in the transactions of goods, the latter concentrates on intangible resources and the creation of value in relation with customers. Service dominance puts the continuous *value co-creation* between providers and consumers at the core of economic exchange. In response to the emergence of the concept of *value co-creation*, the research teams at the Luxembourg Institute of Science and Technology and the University of St. Gallen initiated the ValCoLa (Value Co-Creation Language) project.[2] Some of the results from this project are included in this book as well (Chaps. 6 and 7).

[2] This project was co-funded by the FNR and the SNSF, the Luxembourgish and Swiss research funding agencies, respectively.

The shift of attention from a goods-centered dominant logic to a service-centered dominant logic, combined with the shift toward networked enterprises, results in the development of *service ecosystems* in which different (changing) coalitions engage in value co-creation by way of resource integration. Such *service ecosystems* are not only limited to traditional companies, government agencies, etc. They also include integrated mobility networks, logistic chains, smart cities, etc.

As a result, in the digital age, enterprises need to be *service-focused*.

1.4 The Growing Role of Data as a Key Underlying Resource

With the increasing digitization of society comes an increased role for data. This already resulted in the emergence of, for instance, *big data*, *data sciences*, and *data analytics*. This resulted some consultancy firms to, e.g., coin the concept of *thriving on data* (Capgemini, 2009).

Meanwhile, data is indeed seen as a key resource. The involved data (resources) can, e.g., pertain to "raw" observations from different sensors/informants, "processed" and/or "enriched" artifacts in terms of, e.g., predictive models, digital replicas of real-world phenomenon, nowadays referred to as *digital twins* (Grieves, 2019), representations of "intentions" (e.g., plans, designs, etc.), "specifications" (source code, work procedures, etc.), or "norms" (regulations, principles, policies, etc.).

We have all grown familiar with the possibilities, and the possible positive and negative consequences, for large-scale data collection and utilization, e.g., conducted by Google, Facebook, etc. Data also provides the necessary training data for statistical AI, the basis for the creation of *digital twins* (Grieves, 2019), while also enabling enterprises to continuously assess their performance in real time (Hugos, 2004) and learn to improve their operations (Hess, 2014). As such, digital enterprises are increasingly *data-fuelled*.

As a result of the growing role of data as a key underlying resource, the systems involved in gathering, storing, processing, analyzing, and visualizing data have evolved to be complex systems themselves, involving different socio-technical actors with their own interests. The data involved may pertain to the behavior of humans, thus making it subject to privacy considerations. Data has some correspondence to "something" in the social, economical, or physical world. As such, there is a need to consider the quality of this correspondence, while some actors may have an interest in maliciously changing the data. Data also comes with the question of ownership. Data may be of strategic value to some actors, leading them to want to control the access for others.

1.5 Conclusion

As a result of the above discussed intertwined, and mutually amplifying, trends, enterprises are more than ever confronted with a need to transform. At the same time, these trends pose fundamental challenges to enterprises on the way they should structure themselves, ranging from the services offered, via the processes, organizational structures, and technological infrastructures.

There are no simple answers to these challenges; there is no one-size-fits-all approach to deal with them. The aim of this book is, therefore, to explore different relevant aspects in more detail while at the same time also providing concrete suggestions for enterprises to meet the resulting challenges. In line with this, this book brings together contributions covering four key perspectives:

- **Part I—**As enterprises currently already need to meet the discussed challenges, we will start with an exploration of a broad range of real-world cases, reporting on how enterprises deal with these trends in practice.
- **Part II—**In this part, we will continue by identifying the need for new ways of thinking regarding the design of enterprises, in other words, the need for a new *design logic*.
- **Part III—**In meeting the challenges, enterprises will need to transform (continuously). Such transformations require coordination among the stakeholders involved. Therefore, this part explores several aspects of this needed coordination.
- **Part IV—**The current affairs of an enterprise, as well as its possible future affairs, enterprise, can be captured in terms of models. Therefore, in part we look at the ensuing consequences for *enterprise modeling*.

Part I
Experience Reports

Chapter 2
Introduction

Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

The goal of this part is to discuss, and explore, experiences of real-world organizations with digital transformation.

As we will see, one major theme is the difficulty of successfully planning and executing transformation projects. It appears that we still have a lot to learn in this respect. The following questions are tackled: "What are the capabilities that are needed for successful transformation?" and "Which configuration of capabilities helps organizations to realize their digital dreams, specially in light of the rapidly changing technological capabilities that are available?".

This part also explores how technologies and their integration in organizations aid in new ways of collaboration. The culmination of this trend lies in value co-creation and smart business networks. Blockchain is considered to be a key enabling technology in this area. This part is organized as follows:

- Chapter 3 starts with an exploration of digital transformation projects. The premise of the study is that complexity leads to inefficiencies and as a community we should therefore battle these complexities. Over 50,000 projects from over 1,000 organizations are studied using both traditional and novel metrics for "success."
- Chapter 4 is an experience report from the digital transformation journey of a Dutch insurance company. In this case, some of the required capabilities (e.g.,

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_2

innovation) were already present at the start of the journey, yet an overarching view on *which* capabilities are required and *how* they can be leveraged to drive the initiative was missing. The objective was to develop this overarching view.

- Blockchain is a technology that is well beyond the buzzword phase. In Chap. 5 an analysis of its use as an enabler in smart networks is explored. The premise for the exploration is the transition to decentralized architectures for business collaboration which require transparency for successful operation. The use of blockchain is analyzed through the lenses of an essential/infological/datalogical typology in an attempt to show how blockchain can be used to implement the three critical capabilities of smart networks.
- In Chap. 6, the focus shifts to digital value co-creation. The assumptions underlying this study are threefold: (1) there is a move from goods-dominant to service-dominant logic, (2) IT is increasingly dominant in value co-creation, and (3) digital value co-creation occurs because of complex interplay between human actors and IT artifacts. The study looks at two critical problems. First, there is limited conceptual/ontological clarity of what digital value co-creation networks are and how they work. Second, there is limited understanding of why so few of these networks survive and what a co-creation capability constitutes. This study uses both literature survey and a single case study.
- In Chap. 7 it is recognized that value co-creation is an approach that would benefit from modeling support. Given that ArchiMate is an open language with extension mechanisms, it makes sense to use this language as a basis. The objective of the chapter is to translate the previously developed meta-model for value co-creation to ArchiMate and to validate the ArchiMate extension through a single case study.

Chapter 3
Investigating 5,140 Digital Transformation Projects

Hans Mulder, Jim Johnson, Klaas Meijer, and Jim Crear

Abstract Today's organizations are a complex knot of people, processes, rules, IT, responsibilities, tasks, and much more. Complexity begets inefficiencies, so to transform our businesses we need a new generation of organizational tools that lead to greater efficiency. Digital transformation is defined as: "Transformation of the enterprise with a major impact on digital resources and capabilities of the enterprise." Thus, digital transformation means designing/redesigning and constructing/reconstructing an organizational process in the same way that an engineer would build an automobile, airplane, or computer. A digital transformation project (DTP) is one that completely automates a business workflow, including integration of third-party resources. What distinguishes DTPs from traditional IT projects that support business processes is the elimination of most of the manual processes as part of the workflow, thus truly transforming the business. An example would be online voting.

3.1 Introduction

In this chapter, the Standish Group researches DTPs. Major changes in the way software projects were accomplished resulted directly from the findings of the Standish's research since 1994 (The Standish Group International, 1994): "The

H. Mulder (✉)
Viagroep nv, Schiedam, the Netherlands

Antwerp Management School (AMS), Antwerp, Belgium
e-mail: hans.mulder@ams.ac.be

J. Johnson · J. Crear
Standish Group International Inc., Boston, MA, USA
e-mail: jim@standishgroup.com; jim.crear@standishgroup.com

K. Meijer
Dutch Tax and Customs Administration, Apeldoorn, the Netherlands
e-mail: klaas.meijer@gmx.com

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_3

numbers even found their way to a report for the President of the United States to substantiate the claim that US software products and processes are inadequate" (Joy and Kennedy, 1999). Currently, The Standish Group's research is used more than ever before, recently in the hearings of the Dutch Parliament (DutchICT, 2014). During the last 22 years, the Standish Group's research held the data private and no outside access was permitted. In 2016 the Antwerp Management School (AMS) and the Standish Group presented a view into the working of the CHAOS database (Mulder and Johnson, 2016). The purpose of the presentation was for the Standish Group and Antwerp Management School (AMS) to start the development of an educational program to recreate the CHAOS database.

3.2 Overview of Research Findings

The current CHAOS database is not a collection of surveys, but rather a collection of project and organizational profiles. There are about 50,000 current project profiles and over 1,000 organizational profiles. The Standish Group collects, adjudicates, and approves about 5,000 new projects per year, or an average of 5 projects per organization. Each organizational profile has 24 data points and each project profile has over 80 data points. The database is used to create our research reports, general queries, single-project assessments, future portfolio predictions, and performance benchmarks. The Standish Group selected 5,140 projects within the CHAOS database that fit the DTP definition: "Transformation of the enterprise with a major impact on digital resources and capabilities of the enterprise." We then compiled the results of these projects to present this special DTP report on project success.

The CHAOS database is coded with six individual attributes of success: OnTime, OnBudget, OnTarget, OnGoal, Value, and Satisfaction. Challenged projects fall outside one or more of these metrics, with some reasoning and flexibility that go into our adjudication process. On the other hand, project failure always has the same definition. A failed project is one that has been cancelled before it is completed, or completed but not used. Those are the only two conditions that put a project into the failed category. Many people add "challenged" to the failed category to make failures look bigger, but one should resist this temptation. There are already enough failures without embellishing the numbers.

DTPs using the *traditional* definition of OnTime, OnBudget, and OnTarget have a slightly lower success rate than the overall projects in the database. Our research, as illustrated in Fig. 3.1, shows that 37% of 50,000 projects in the CHAOS database were successful, while 34% of DTP projects were successful.

Our *modern* definition of success is OnTime, OnBudget, and with a satisfactory result. This means the project was resolved within a reasonable estimated time, stayed within budget, and delivered customer and user satisfaction regardless of the original scope. We have the flexibility to present the results for one to six of these attributes in any combination. We consider the modern resolution to be a better definition of success than the traditional one, because it combines the

Resolution by Traditional Measurement			
Resolution	All	DTP	Delta
Successful	37%	34%	-3%
Challenged	44%	45%	1%
Failed	19%	21%	2%

Fig. 3.1 The *traditional* resolution of all projects and DTP projects from FY 2007 to 2016 within the CHAOS database; traditional resolution is OnTime, OnBudget, and OnTarget

Resolution by Modern Measurement			
Resolution	All	DTP	Delta
Successful	29%	28%	-1%
Challenged	52%	51%	-1%
Failed	19%	21%	2%

Fig. 3.2 The *modern* resolution of all projects and DTPs from FY 2007 to 2016 within the CHAOS database

project management process *and* the end results of a project. We have seen many projects that have met the triple constraints of OnTime, OnBudget, and OnTarget (see Fig. 3.2), but the customers were not satisfied with the outcome and on many occasions refused to use the new system. This is evident in the data, which shows an 8% decrease in the success rate and a 6% increase in the challenged rate from 2007 to 2016.

Using the *modern* measurement of OnTime, OnBudget, and with a satisfactory result, DTPs had the same lower success rate as the overall projects in the CHAOS database. Our research shows 29% of 50,000 projects in the CHAOS database were successful, while 28% of DTP projects were successful. However, using both *traditional* and *modern* metrics, DTPs failed 2% more than the projects overall.

Project size has always been a major element in the CHAOS research and DTPs are no exception. It was clear from the DTP data that project size is a major determinant if the project will be successful or returns value. Only 5% of the very large or grand DTPs were OnTime, OnBudget, and with a satisfactory result. The Standish Group revised the size scale from a pure labor cost based to a category a couple of years ago to reflect the changing labor costs and other factors, small is generally a same team for a few months, while it is a very large team over a few years. In many cases larger projects never return value to an organization. The faster the projects go into production, the quicker the payback starts to accumulate. On the other hand, 57% of small DTPs came in OnTime, OnBudget, and with a satisfactory result. Strangely, moderate-sized DTP projects had a 58% chance of a successful result with fewer failures.

3.3 Microprojects to Support Digital Transformation

For many years we have been recommending microprojects or small projects. Yes, less can really be more in the long run. We have seen a major uptick in the use of microservices, which is basically a microproject. However, while this does increase success rates, microprojects come with a problem of coupling the services together.

In order to overcome the coupling problem of microservices/microprojects, we are investigating a promising new method called Normalized Systems (Mannaert and Verelst, 2009; Mannaert et al., 2016), which—among others—is also used by the Dutch Tax and Customs Administration (DTCA) in some of their projects. With the Normalized System method, we can even create smaller projects, and we have termed these types of projects nanoprojects or nanoservices.

The Dutch Tax and Customs Administration levies, collects, and checks taxes on behalf of the government. They are also responsible for paying benefits in the form of financial support for rent, for children, and for health insurance. They monitor goods coming into the country. Besides dealing with issues relating to security, health, the economy, and the environment, the DTCA also levies excise duties. The DTCA is one of the largest government authorities in the Netherlands. Every year about 29,000 employees collect some €232 billion in tax revenues and distribute more than €12 billion in benefits.

Business processes and communication with citizens, enterprises, and other governmental agencies are highly automated. Handling these volumes is only possible because of a high level of computerization. A lot of interconnected IT systems have been built and used to support these business processes for many years.

However, this high level of computerization also has some drawbacks. Continuous requirements from legislation and business, lack of maintenance, and shrinking budgets have led to a complex IT landscape with a lot of point solutions and technical debt. As a result, it takes a lot of effort to guarantee business operations, and fulfilling new demands has a big impact and takes more and more time and money. Structural improvements of the IT landscape—better, cheaper, and more agile—are almost impossible.

In 2014, the Court of Auditors (CoA, 2014) concluded that the Dutch Tax and Customs Administration is not in control of its IT. This includes all aspects of IT, from applications to development tools and from hardware platforms to cost of maintenance and operations. An important finding of the Normalized Systems approach as used by DTCA is that the participation of the ecosystem thus including the client, executive sponsor, and especially the users of a continuous delivery pipeline releases approach is essential.

3.4 Complexity and Absorption

Complexity is one of the main reasons for project failure. Figure 3.3 shows the resolution of DTPs by complexity from FY 2012 to 2016 within the CHAOS database using the *modern* definition of success. The results show that 35% of easy projects were successful. Very complex projects had both the highest challenged (60%) and failure (31%) rates.

The reader should bear in mind, though, that inside of every complex problem are simple solutions. Complexity is often caused by size, conflicting goals, large budgets, and project sponsor egos. Complexity creates costs and confusion. The Standish Group believes that the Normalized Systems methodology has great

DTP Size by Resolution

Resolution/Size	Successful	Challenged	Failed	Total
Grand	5%	52%	43%	100%
Large	12%	58%	30%	100%
Medium	17%	57%	26%	100%
Moderate	58%	35%	7%	100%
Small	57%	35%	8%	100%

Fig. 3.3 The resolution of DTPs by size from FY 2007 to 2016 within the CHAOS database

promise in reducing complexity, since each nanoproject focuses on a narrow set of features and requirements, which users find easier to understand and absorb. The faster you can introduce smaller changes, the more acceptable they are and the easier to implement.

Decreasing complexity by reducing combinatorial effects in software and processes provides for reducing complexity as much as possible in the face of more complex requirements. Reducing complexity supports absorption. Lehman (1980) states "software does not age with use, but deteriorates by the need to adapt it to the changing technology and new user requirements." He identifies three kinds of information systems: static, parameterized, and environmental (the most recent kind). The problem, according to Lehman, is the so-called E-Systems, which must be adjusted to the changing environment, for example, new legislation or cloud technology. DTPs are very close in attributes to E-Systems.

The Standish Group's *The CHAOS Manifesto* report introduced Absorption Theory, which includes continuous change, decreasing complexity, and maintaining familiarity. Absorption Theory is the ability of the organization to successfully grasp business and technical changes without disruption. Absorption Theory stems from the laws of Lehman (1980), who was the chairman of the Department of Computing at the Imperial College of London. From 1974 onward Lehman worked on eight laws of software evolution. Lehman suggested there need to be continuing applications and systems growth in order to maintain user satisfaction. Applications and systems growth will cause a decline in quality as well as increase complexity. These eight laws are the foundation for the three broad concepts: continuous change, decreasing complexity, and conservation of familiarity.

Example: A client outsourced a modernization project to their offshore software development group. After some startup and learning, the offshore group started delivering microproject deliverables at a pace of two to ten deliverables per week. The onshore (client) team had no idea or processes for testing and releasing these kinds of continuous delivery projects (their normal QA, test, release cycle was measured in months not days). After 9 months of deluge releases from the offshore team, they only managed to put one (1) deliverable into production and then cancelled the modernization project. A lesson learned is "Make sure your customer is prepared for drinking from the fire hose before pointing it at them." Absorption at a higher delivery pace requires three broad concepts that can help continuous change, decreasing complexity, and conservation of familiarity. These concepts increase the ability of the organization to successfully grasp business and technical changes without disruption.

Continuous Change: This helps to keep software modern and useful, reducing the need for large mass absorption. Continuous change means adding new functionality and/or changing current functionality. The most important activity to keep software modern and useful is to remove old software. At the same time, newly manufactured software is typically replaced after 7 years of use and adjustments, under the pretext that the old software has deteriorated. However also these "bypass operations" of implementing—next to the current old system—a new system can fail, thus leading to the uncomfortable situation that "The oldest software in the world is not in a museum, but rather is still used daily on the mainframes of large organizations."

Conservation of Familiarity: This is maintaining the user experience while making changes that are intuitive. Many projects fail during implementation because users fail to absorb the new system. Lehman points out that as an application evolves, all people associated with it, for example, developers, sales personnel, and users, must maintain mastery of its content and behavior to achieve satisfactory evolution. The aging, growth, and increasing complexity of software diminish that mastery in practice; this loss of mastery of applications is mostly felt by large organizations such as banks, insurance companies, and government agencies. Here the old adage holds true: "most companies/projects don't die from starvation; they die from indigestion."

Two of the major overriding attributes that determine the chances of a DTP success or failure are size and complexity (also see Fig. 3.4). Size is determined primarily by labor effort. Labor effort is determined by the cost of normalized labor, number of persons, and the overall size of the team. We also consider the number of functions, lines of code, and other factors to determine size.

Determining complexity is more complex. We use about 25 project attributes to determine complexity, such as the number of stakeholders, diverse user profiles, and innovation descriptions, not to mention diverse locations. Complexity ranges from very complex to very simple. A few years ago, The Standish Group created the Size-Complexity Matrix, as shown in Fig. 3.5, as a way to determine the estimated likelihood of success based on both a rating system and a color code. This matrix is based on more than 100,000 projects collected over 20 years. Green means the

DTP Resolution by Complexity			
	Successful	Challenged	Failed
Very Complex	9%	60%	31%
Complex	13%	58%	29%
Average	27%	54%	19%
Easy	35%	46%	19%
Very Easy	34%	48%	18%

Fig. 3.4 The resolution of DTPs by complexity from FY 2007 to 2016 within the CHAOS database

Size-Complexity Matrix						
		COMPLEXITY				
		Very Simple	Simple	Average Complexity	Complex	Very Complex
SIZE	Small	100	250	400	550	625
	Moderate	175	325	475	625	775
	Medium	250	400	550	700	850
	Large	325	475	625	775	925
	Grand	400	550	700	850	1000

Fig. 3.5 The Size-Complexity Matrix provides guidelines for categorizing a project

Size Guidelines		
Size Description		Size
Under $1 million labor	6 or less team members/months	Small
$1 million to $3 million	7 to 12 team members/months	Moderate
$3 million to $6 million	13 to 24 team members/months	Medium
$6 million to $10 million	25 to 50 team members/months	Large
Over $10 Million	Over 50 team members/months	Grand

Fig. 3.6 Guidelines on how to measure the size of a project

project has a good chance of success, yellow means the DTP will most likely be challenged, and red means the project has a very good chance of failure.

In order to assess the risk and effort, the Size-Complexity Matrix uses a 5-point scale for both size and complexity. The lowest-point project is a simple, small project and has 100 points. The largest and most complex project has 1,000 points. Green means low risk and effort, yellow means medium risk and effort, and red means high risk and effort.

It is easy to create your own Size-Complexity Matrix estimate using the following tables and guidelines. Guidelines on how to measure the size of a project are provided in Fig. 3.6. The left side of the table uses labor cost. Standish uses labor effort as a major ingredient to measure size; therefore, when selecting the project size in the table, use normal US labor rates. The right side of the table uses team size. You can take the average of both tables or select the highest or lowest table. Remember these are guidelines, not rules.

The complexity guidelines are more complex. You need to assign points and add up the points based on the attributes of the project per the complexity guidelines table. The higher the points, the more complex the project. We use two dimensions to complexity: environment and scope. If none of the attributes apply, then the project is very simple. If you score fewer than 3 points, the project is simple. If you

Fig. 3.7 Guidelines on how to measure the complexity of a project

Complexity Guidelines	
Environment	Points
Diverse User Base	1
Multiple Team Locations	1
Multiple Stakeholder Locations	1
Uncooperative Peers	2
Uncooperative Stakeholders	3
Scope	Points
Many Requirements - Large scope	1
Ambiguous Basic scope	1
Fuzzy Undefined Requirements	1
Diverse and Multifaceted Objectives	2
Breaking New Ground	3

score 4–7 points, the project is average, while at 5–9 points, the project is complex. If you score above 10 points, the project is very complex. There are a couple of ways you should use this matrix. First, determine the project forecast in terms of size and complexity. Then think of it in terms of your DTP experience as a role model. We had the benefit of 100,000 detailed projects to draw on as our role models, both good and bad (Fig. 3.7).

3.5 Digital Transformation Success Factor: The Executive Sponsor

The single most important person involved with a project and ultimately responsible for its success or failure is the project executive sponsor. The Standish Group's CHAOS database consistently shows that project improvement and success are dependent on the skills of the project sponsor. The larger and more complex the project, the more the skills of an executive sponsor can make a difference between success and failure. For example, as illustrated in Fig. 3.8, the Standish Group 2016 CHAOS database shows that greater than 50% of successful very large, complex projects had a highly skilled project sponsor. On the other hand, over 60% of failed very large, complex projects had a moderate to poorly skilled project sponsor. Be advised that the project sponsor, depending on his/her skills, can make or break any project regardless of its size.

Before beginning any DTP project, Standish Group recommends that the organization find and appoint a skilled and responsible project sponsor. The purpose

DTP Resolution by Project Sponsor				
	Highly Skilled	Skilled	Moderately Skilled	Poorly Skilled
Successful	36%	33%	20%	11%
Challenged	11%	43%	29%	17%
Failed	10%	23%	40%	27%

Fig. 3.8 The resolution of DTPs by the skill level of the project sponsor from FY 2007 to 2016 within the CHAOS database

of the book *The Good Sponsor* (Johnson, 2016) is to act as a guide and to help project sponsors understand their roles and responsibilities and to improve their skills. The book outlines the ten attributes of a good sponsor. The Standish Group has an assessment test to determine the skill level of a project sponsor. A DTP requires a project sponsor who is either very skilled or at least skilled. Both the book and the assessment provide exercises to help project sponsors improve their skills.

Another important factor for a successful DTP is the team's emotional maturity. In project management speak, emotional maturity is the soft skill. The organization needs to be skilled at emotional maturity to have a healthy project ecosystem for a DTP. Emotional maturity supports and promotes the skills to be self-aware, socially aware, self-managed, and able to manage relationships, among other skills. In many ways, emotional maturity is the group dynamics of emotional intelligence. Emotional maturity is all about communicating what people are going to do and when they are going to do it, and making sure they do it. The resolution of DTPs by the emotional maturity skill level within the CHAOS database is shown in Fig. 3.9.

For example, the team needs to continually provide updates to all stakeholders on what has been accomplished. This can be done in formal meetings or published updates. Weekly or biweekly updates during the heavier times in a project are

DTP Resolution by Emotional Maturity				
	Highly Skilled	Skilled	Moderately Skilled	Poorly Skilled
Successful	34%	39%	19%	8%
Challenged	15%	27%	41%	17%
Failed	15%	14%	47%	24%

Fig. 3.9 The resolution of DTPs by the emotional maturity skill level of the project team from FY 2007 to 2016 within the CHAOS database

beneficial. The Standish Group's *Emotional Maturity Research Report* (Johnson, 2013) outlines and discusses the *Five Deadly Sins of project management*, which are arrogance, abstinence, fraudulence, ignorance, and overambition. Overcoming the five sins is the cornerstone of emotional maturity. These five sins are the subject of the book *The Public Execution of Miss Scarlet* (Johnson, 2007). Other traits of emotional maturity include insisting that bad news travels fast. The organization's ability to manage expectations is also important, as are listening skills. Team members must be attentive listeners as well as both realistic and objective. Finally, the team must get good at gaining consensus to achieve buy-in.

Successful projects need smart trained people. We are not just talking about the IT team. Never underestimate the need to have a smart and engaged user community that accepts ownership of the project. After all, they will be living with it when it is completed. Not surprisingly, one of the key project success factors that Standish has identified since the beginning of the CHAOS research is a competent staff. There are five key fundamentals to ensure staff competency. First, identify the required competencies and alternative skills. Second, provide a good continuous training program to enhance the staff skills. Third, recruit both internally and externally to provide a balance of experiences. Fourth, provide incentive to motivate the staff. Finally, ensure the staff is project-focused.

When a project has both teamwork and skilled resources, it can prevail under even the direst of circumstances. To ensure a competent staff, you must match the skills of the team to correspond with the needed skills of the project. Capability is one of the seven constraints we use to prioritize your project portfolio. Constraints are limitations or restrictions. The other six constraints are cost, risk, value, goal, timing, and exclusions. Figure 3.10 shows the resolution of all DTP by capability from FY 2012 to 2016 within the CHAOS database using the *modern* definition of

DTP Resolution by Capability			
	Successful	Challenged	Failed
Gifted	35%	43%	22%
Talented	28%	54%	18%
Competent	27%	52%	21%
Able	21%	53%	26%
Unskilled	13%	63%	24%

Fig. 3.10 The resolution of DTPs by capability from FY 2007 to 2016 within the CHAOS database

success. The results show that projects that had gifted resources had a 35% success rate. Projects that had unskilled people had the highest challenged rate of 63%, and projects with just an able staff had the highest failure rates of 26%. One of the decisions around project priority includes *Do you go forward with a project if you lack skilled capability?* This decision is especially pertinent for large projects with a large staff who have a mix of good and poor resources. This is one of the reasons that small projects have a higher success rate, since small projects are easier to staff with high-performing teams. For example, the DTCA has a small staff of well-trained Normalized Systems engineers which produce tremendous output.

3.6 Optimization

Optimization is another important factor for both success and value. A quote often associated to Benjamin Franklin[1] is that "a penny saved is a penny earned." In this regard, a marginal and less important feature that is not included means the resources can be used for more value-based efforts for the organization. While many might consider every feature or function to have value, some are more valuable than others. However, The Standish Group has shown that only about 20% of features and functions get used frequently, while the other 80% of features are not used very much or not at all. The Standish Group research shows that DTP teams that are skilled at optimization have a higher *modern* success rate.

The Standish Group's optimization process allows you to measure features and functions relative to each other, thus making it clear which ones have the highest value. This enables the team to prioritize more easily and obtain value more rapidly. Figure 3.11 shows that highly skilled DTP teams with optimization have a 25% success rate versus only a 10% failure rate. The reason organizations need to be good at optimization is because of the Constraints Theory. The Standish Group has

DTP Resolution by Optimization				
	Highly Skilled	Skilled	Moderately Skilled	Poorly Skilled
Successful	25%	24%	30%	21%
Challenged	13%	24%	30%	33%
Failed	10%	21%	31%	38%

Fig. 3.11 The resolution of DTPs by the optimization skill level of the project team from FY 2007 to 2016 within the CHAOS database

[1] https://quotes.yourdictionary.com/articles/who-said-a-penny-saved-is-a-penny-earned.html.

identified several constraints to measure and optimize projects: money, time, timing, scope, capability, resources, complexity, risk, goal, and order. Each one of these constraints needs be assessed and balanced for true optimization.

3.7 Agile Digital Transformation

The agile process, such as Scrum, provides an enhanced method to execute DTPs. Figure 3.12 compares the resolution of DTPs from FY 2007 to 2016 within the new CHAOS database segmented by the agile process and waterfall method. The total number of software projects is more than 5,000. The results for all projects show that agile projects were over three and a half times more successful than waterfall projects and waterfall projects had three times the failure rate of agile projects. Other results are also broken down by project size of large, medium, and small. They clearly show that waterfall projects do not scale well, while agile projects scale much better.

The combination of Scrum and Normalized Systems creates a pipeline of nanoprojects. The pipeline works by creating output in a rapid process. Nanoprojects or services come into the pipeline; they get completed in a day to a week, then go into a rapid QA process, and are then sent to a user test group. If the project works, it is implemented and absorbed by the user community. Nanoprojects that fail in either QA or user acceptance are reevaluated and may or may not be reintroduced to the pipeline. One of the most important benefits is the organization can take more risk since the failures are also very small, or nanofailures, that have little impact and cost for the organization.

Fig. 3.12 The resolution of DTPs by agile versus waterfall from FY 2007 to 2016 within the CHAOS database

DTP Resolution by Method			
Method	Successful	Challenged	Failed
Agile	36%	54%	10%
Waterfall	10%	60%	30%

3.8 Project Managers

For many years the project manager was considered the project's linchpin. Money poured into developing education, offering certification, building project management offices, and implementing enterprise project management tools. These investments were in hopes to change the direction to increase project success and improve value. In fact, The Standish Group research shows that improvement in these areas actually caused the opposite effect while increasing the costs of projects and decreasing their value. We are not saying that project managers have no value. However, their value may be overrated. Project managers should have the basic mechanical skills of planning, tracking, and controlling. Project managers should provide an early warning system for projects that are not progressing. In the agile or Scrum world, many consider the project manager unnecessary, since many of the duties are split between the product owner and the Scrum Master.

For DTPs there is a role for a project manager; the role is to be the eyes and ears of the project sponsor. We recommend that the project sponsor be the first person to join the project. The second person should be the project manager. The project sponsor should interview project manager candidates and choose one with whom he or she can work as an assistant. *The Good Sponsor* book (Johnson, 2016) has 25 suggested questions the project sponsor should ask a project manager. In Fig. 3.13 we can see that beyond the basic skills, the project manager does not have a major impact on the success of DTPs.

3.9 Project Types

The type of project has a major effect on resolution. Figure 3.14 shows the resolution of DTPS by project type from FY 2007 to 2016 within the CHAOS database using the *modern* definition of success. Projects using a modernization-in-place technique had the highest success rate at 55%. This is the process the DTCA is using. They are modernizing their application suite by one tax process at a time and implementing

DTP Resolution by Project Manager				
	Highly Skilled	Skilled	Moderately Skilled	Poorly Skilled
Successful	31%	37%	23%	9%
Challenged	36%	43%	18%	3%
Failed	29%	42%	26%	3%

Fig. 3.13 The resolution of DTPs by project manager skill levels from FY 2007 to 2016 within the CHAOS database

Project Type	Successful	Challenged	Failed
Developed from scratch using traditional languages and methods	20%	61%	19%
Developed from scratch using modern methodologies	23%	54%	23%
Developed some components & purchased others	21%	59%	20%
Purchased components & assembled the application	24%	54%	22%
Purchased application & extensively modified	32%	45%	23%
Purchased application & modified	53%	27%	20%
Purchased application & performed no modifications	46%	40%	14%
Modernization	55%	35%	10%

Fig. 3.14 The resolution of DTPs by type from FY 2007 to 2016 within the CHAOS database

it without changing other parts of the application suite or functions. On the other hand, DTPs that were developed from scratch using traditional languages and methods had the lowest success rate of 20%. The results also show that projects that were developed from scratch using traditional languages and methods had the highest challenged rate at 61%. The lowest challenged rate of 27% went to projects of purchased application software with modifications. The highest failure rate of 23% went to projects of purchased software with extensive modifications. Modernization-in-place projects had the lowest failure rate at 10%. Our research into the DTCA using Normalized Systems as modernization-in-place approach is consistent with our findings that this is both a safe method and creates value.

3.10 Goals

The Standish Group has stated for many years that clear goals are achieved when all the stakeholders are focused on and understand the core values of the project. We used to believe that goal clarity and focus were essential to a successful project. However, by measuring success by both the *traditional* and *modern* metrics, we found the opposite to be true. We coded the database with a 5-point scale, from precise to distant, in order to measure the effect on success rates. The results are shown in Fig. 3.15. It is clear from the research that goals closer to the organization's strategy/goal have the opposite effect on higher satisfaction and success rates. The Standish Group uses goal as one of the seven constraints as part of our Optimization Clinic. The Optimization Clinic is the third step in our Value Portfolio Optimization

DTP Value by Goal					
Value	Precise	Close	Loose	Vague	Distant
Very High	11%	13%	29%	26%	21%
High	8%	15%	26%	27%	24%
Average	19%	25%	31%	16%	9%
Low	23%	27%	23%	10%	17%
Very Low	22%	25%	26%	12%	15%

Fig. 3.15 The value of DTPs by goal from FY 2007 to 2016 within the CHAOS database

and Management Service. We also use goal as one of the measurements for our Resolution Benchmark.

The Standish Group is now suggesting that your organization take action over trying to achieve clarity. Many of the most satisfying projects start out with vague goals. The business objectives are dynamic as the project progresses. Project teams should reduce or give up control of the business objectives to encourage and promote innovation. Consider value first followed by goal. We see that many projects that achieve high value are distant from the goal. Therefore, it is imperative that the goal be downgraded to be less important for DTPs.

3.11 Industries

Looking at project resolution by industry provides another view of the CHAOS database. The table on this page shows the resolution of DTPs by industry from FY 2007 to 2016 within the CHAOS database. The results show that retail projects had the highest success rate at 35% using the *modern* definition of success. The results, see Fig. 3.16, also show that government and financial projects had the lowest success rates at 14% and government projects had the highest failure rate at 25%. Considering the industry results highlights the achievements of the DTCA using Normalized Systems.

DTP Resolution by Industry			
Resolution/Industry	**Successful**	**Challenged**	**Failed**
Banking	25%	58%	17%
Financial	14%	62%	24%
Government	14%	61%	25%
Healthcare	29%	54%	17%
Manufacturing	24%	55%	21%
Retail	35%	47%	18%
Services	28%	51%	21%
Telecom	25%	51%	24%
Other	33%	46%	21%

Fig. 3.16 The resolution of DTPs by industry from FY 2007 to 2016 within the CHAOS database

3.12 Factors of Success and Value

In the 2016 CHAOS Report (The Standish Group International, 2016), we combined the success and value tables into one table called *factors of success and value*. The figures reflect our opinion of the importance of each attribute and our recommendation for the amount of effort and investment that should be considered to improve DTP success and value. It is clear to us that creating a "winning hand" requires five elements: a small project using an agile process, with the three skilled areas of project sponsorship, technical staff, and an emotionally mature organization (Fig. 3.17).

It is our tradition to assign points to each factor to highlight its relevance. These points should also be considered as an investment guideline for project management improvement. The Standish Group believes that 80% of your project improvement budget should be spent on these five areas. We also recommend reducing the high overhead of the other six areas to fund these five most important areas. For example, if you are spending $50 million on IT projects, then 2% of the money should be going toward improving the value of those projects. Based on this amount, our recommended breakdown of money to be allocated to each factor is calculated next to the point value on the chart. So, if you want your projects to be more successful, with higher value and greater customer satisfaction, you should carefully consider where you invest your project improvement money.

Factors of Success/Value	Points	Investment
Small Agile Projects	25	25%
Executive Sponsorship	15	20%
Emotional Maturity	15	20%
Talented Staff	10	15%
User Involvement	9	4%
Optimization	8	4%
SAME (Standard Architectural Management Environment)	6	3%
Modest Execution	5	3%
PM/Process Expertise	4	3%
Clear Business Objectives	3	3%
Total Points & Yearly Investment	100	100%

Fig. 3.17 Factors of success and value, adapted from The Standish Group International (2016)

3.13 Conclusions

In summary, we make the following ten recommendations to achieve value and success for a DTP:

1. Pick a skilled project sponsor to head the project. The project sponsor's job is outlined in *The Good Sponsor* book. The book identifies the ten major attributes of a good project sponsor. There is also an online self-assessment test that will score the skill level of the project sponsor.
2. Test the team for emotional maturity. The Standish Group offers an emotional maturity test kit that includes self-improvement.
3. Create a small team of talented or gifted individuals whose skills match the project's technical and business requirements. A small group of talented or gifted individuals can produce more features and functions in less time than a group of mediocre staff.
4. Create a pipeline of small stepping-stone deliverables. This will create rapid feedback, quick adoption, or quick rejection. If rejected, find and fix the problem quickly and reintroduce the new deliverable.
5. Use an agile methodology such as Scrum to execute the project and pipeline.
6. Optimize stepping-stones by value, complexity, cost, and capability. The Standish Group's optimization process can help you examine and optimize your deliverables.

7. The project sponsor should interview at least three project managers to provide assistance and progress information. *The Good Sponsor* book has 25 suggested questions that the project sponsor can ask the project manager.
8. Make quick decisions. Decision latency is a major cause of project stress, delays, and failures. The Standish Group has many examples of where a quick response was much better than a drawn-out response.
9. Watch out for project saboteurs. A project saboteur is a person who does not want the project to succeed and will take action or refrain from taking action to sabotage the project.
10. Only use trustworthy vendors and, even then, keep them on a short leash. Demand rapid deliverables, not promises.

We make these recommendations based on the cases collected in the CHAOS database, conversations with our case adjudicators, interviews, and workshops with CIO/IT executives that have implemented DTPs. The Standish Group has a long-standing policy to only use our primary research. We pride ourselves that our independent research is pure with no outside influence.

As our research only was of an exploratory nature, more research in the matter of value and success for a DTP has still to be done.

Chapter 4
From Product-Oriented Insurance Company to Customer-Centric Service Provider

Ralph van Vliet and Joppe Ter Meer

Abstract Insurance company Aegon encountered a complex transformation challenge: transforming from a division-based, product-oriented company to a modern service provider who operates as a coherent organization, knows its customers, and excellently serves them in a personal way. This challenge forced Aegon to a disruptive approach: a totally new service concept, based on social media, radical new business processes, and a fast switch to a cloud-based Salesforce application landscape. Much emphasis was laid on management style, professional skills of the workforce, and a customer-centric attitude. A comprehensive transformation program, starring the Douma persona's family, led to a fundamental new business model.

4.1 Introduction

4.1.1 Disruptive Environments Necessitating Transformation

The world is more dynamic than ever for insurance companies. The way insurance companies do business, the way they connect and engage with their customers, and the way customers wish to connect with them are rapidly changing. More than ever, it poses a challenge to hold and regain customer confidence. Furthermore, technological developments follow rapidly on each other while legislations change fast. New and more direct distribution models become more dominant and customers expect valuable products and top service.

R. van Vliet
Share a Byte, Bodegraven, the Netherlands
e-mail: rvanvliet@novius.nl

J. T. Meer (✉)
Novius/RHDHV Digital, Amsterdam, the Netherlands
e-mail: jtermeer@novius.nl

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_4

To be successful in such a challenging environment, insurance companies need to be customer centric. This means possessing the capabilities of knowing the customer, offering transparent products, and delivering adequate services. Especially in this era, digital customer interaction and services should rather be a rule than an exception. In order to act in these challenging times, agility is key.

4.1.2 Introducing Aegon

These changes also apply to the case of insurance company Aegon the Netherlands, who faced a complex transformation challenge in order to become a customer-centric service provider. Aegon's mission—*to help people achieve a lifetime of financial security*—forms the basis of the company's strategy. Set against the rapid external changes, this required Aegon to transform the organization by shifting from the originally product-based company to the desired customer-driven organization.

Before the transformation, Aegon was organized through a number of solid "business lines" (pension, life, mortgage, damage, income and banking), supported by various departments, including finance and IT. The culture of Aegon was typified by high involvement of employees, strong entrepreneurial spirit, and large degree of autonomy.

Aegon can be characterized by two sides: innovative and future-oriented. Innovations were continuously being developed and successfully put on the market such as online banking platform Knab, Facebook insurer Kroodle, and target group-oriented labels like Onna-onna, while on the other hand, Aegon has been in a stable position and had a comfortable position in the (Dutch) market for decades. The core activities of the business lines consisted of selling and administering products, largely through insurance brokers rather than developing long-term customer relationships. In addition, these business lines worked independently from each other, without a central customer strategy. It is therefore unsurprising that change activities from the past that involved multiple business lines proved to be tough.

In conclusion, some basic capabilities (innovative and future-oriented) were already present. Nonetheless, Aegon had to come from a long way in order to become a digitally enabled, customer-centric organization according to its strategy. Therefore, Aegon was up to a challenging transformation, comprising of changes to the organizational setup, a new way of working, and a major impact on and improvement of its (digital) capabilities.

4.1.3 Core Question

It was clear that Aegon was up to a challenging task to create sufficient support and urgency to realize the transformation. Therefore, this case answers the questions: *What kind of (digital) capabilities has Aegon developed to transform toward a*

Table 4.1 Key terms as used in this chapter

Customer The actual customers of Aegon on which the transformation was centerd. Out of simplicity arguments, we do not distinguish between the private and business customers in this chapter.
Business line Also business units, of which Aegon comprised of: Pension, Life, Mortgage Damage, Income and Banking products.
Contact domain The part of Aegon that is predominantly occupied with customer contact, across the various business lines of Aegon. The contact domain has been the main focus of the transformation program Douma.
Business Transformation Framework A method that helps organizations design, develop, plan and control a (digital) transformation that is needed to achieve its business goals.
Douma program he program aimed at transforming the organization towards a better service provider. In this chapter also referred to as the "transformation program". Douma represents the typical family being customer of Aegon.

customer-centric service provider and *how has Aegon realized this?* Before we jump into these two questions, the next section discusses the *history* and *ambition* of Aegon in more detail.

Apart from the definitions as discussed above, this chapter makes use of the terms as shown in Table 4.1.

4.2 History and Ambition: From Product Orientation to Customer Centricity

4.2.1 Historical Background of Aegon

From the beginning, Aegon's core activities consisted of selling and administrating products rather than developing long-term customer relationships from one customer strategy. In the previous years, the "product-oriented" business lines had considerable freedom in designing their processes and information, without a central (customer) strategy. As a result of these independent business lines, information flow was largely top-down oriented within these business lines, hindering cross-organizational communication. Due to this organizational setup, business lines had especially autonomous contact with customers (and advisors) (see Fig. 4.1). In worst cases, this led to a situation where a customer "had to die four times" to end his/her products, collections, and/or payments in four separate business lines.

In addition, core (IT) systems in the contact domain were outdated and fragmented across the six isolated business lines. Therefore, managing customer data was an almost impossible task. Although customer data was synced via a central database (core system), each business line had its own policy in registering new

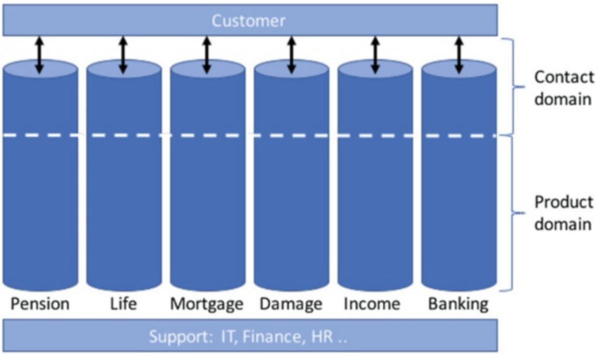


Fig. 4.1 Product-oriented business lines; business lines had especially autonomous contact with "their" customers

customers and enriching customer data. For years, Aegon did not want to face the replacement of the core system, despite the fact that customer data more and more became a vital aspect of Aegon.

Moreover, former business transformations across multiple business lines had proven to be tough. Working according to a company-wide "program management standard" and adhering to a tight governance were never strongly developed in the entrepreneurial culture of Aegon. The logical step was to keep projects small and manageable. Such approach would, however, not work for the management of a company-wide transformation as intended by Aegon.

4.2.2 Ambition: Toward a Customer-Centric Service Provider

As a result of the more growing need to transform, the management board of Aegon set its standards high: it is Aegon's ambition to be regarded as *a trusted partner for financial solutions at every stage of life in all its markets*. That means being recognized by customers, business partners, and society as a company that puts the interests of its customers first in everything it does. In order to do so, Aegon needs to serve diverse and evolving needs across the customer life cycle ("right time, right solution"). This requires a development of digitally enabled long-term customer relationships by providing guidance and advice. Furthermore, Aegon needs to be able to identify additional financial security needs at every stage of customers' lives.

An almost perfect world, but not unreachable and at the same time necessary to achieve the desired benefits for customers and improve the organization, is described as follows: providing significantly better services and professionalism of the organization, putting the interest of the customers first, and realizing a robust infrastructure for CRM and master data management. To realize this ambition, Aegon faced an unprecedented transformation.

At the request of the management board of Aegon, consulting firm Novius[1] was asked to conduct a preliminary research (later called "blauw en wauw" (blue and wow)) to determine the approach and directions for improving the contact domain and develop the outline of the transformation program.

4.3 Approach for Transformation: "Blue" and "Wow" Directions

The preliminary research was carried out with the aim to study scenarios for the development of the contact domain and set up the transformation program. The research largely had two directions, being implemented simultaneously so both "streams" could reinforce each other:

1. *The "blue" structured approach*—With use of the Novius[2] Business Transformation Framework (BTF), Aegon developed the vision, guiding principles, and the architecture designs of the new organization.
2. *The "wow" innovative and unstructured approach*—This focused on achieving knowledge, exploring new possibilities, developing prototypes, and instigating enthusiasm among Aegon employees for possible solutions.

On the one hand, the "blue" direction was executed at Aegon with the use of the Business Transformation Framework (BTF). The BTF (see Fig. 4.2) is a method that helps organizations design, develop, plan, and control a (digital) transformation that is needed to achieve the business goals (Stoop et al., 2016). This method provides the foundation and tools to draw up a business transformation plan. At a substantive level, working with the BTF imposes coherence between *strategy and objectives* and *business transformation portfolio* as well as identifying the impact on the four different levels of the organization: (1) "Customer & Services," (2) "Processes & Organisation," (3) "Information & Applications," and (4) "IT infrastructure & facilities." The BTF can only generate the maximum effect when it is regarded as a way of thinking and acting. Therefore, the method provides input for program management, portfolio management, and enterprise architecture.

[1] Adviesgroep Novius is a Dutch consulting firm which operates in the field of comprehensive digital and business transformations. Adviesgroep Novius supports clients with the use of propositions centered around major digital breakthroughs: data-driven way of working, smart digital processes, optimal customer experience, agile organizations, and digital business models. On 23 October 2020, Adviesgroep Novius is officially part of Royal HaskoningDHV Digital.

[2] Novius has been actively involved in the Douma program, first of all by supporting transformation planning and enterprise architecture within Aegon. In order to support this, Novius applied their Business Transformation Framework and Novius Architecture Framework. Furthermore, Novius took on various roles during the Douma program, e.g., in the role of enterprise architect, as a trusted advisor to the CEO and on management level to secure all culture-driven interventions in the organization.

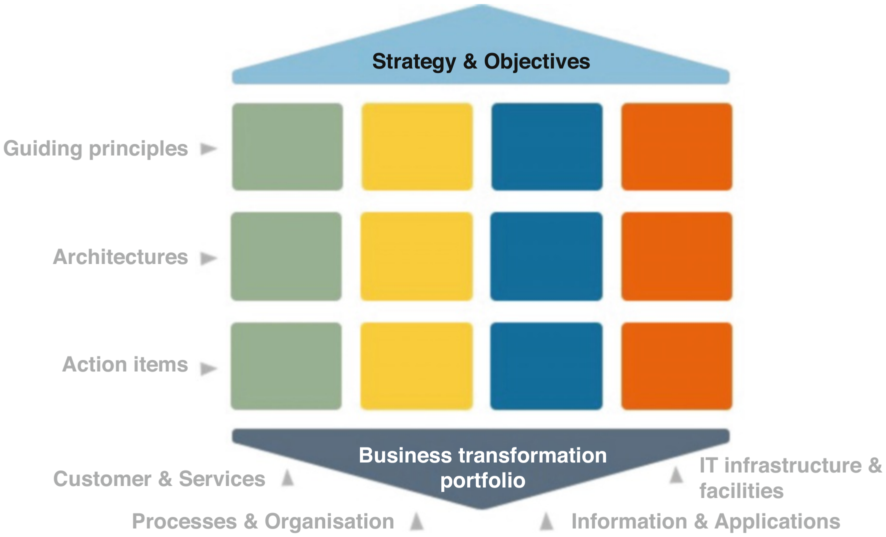


Fig. 4.2 The Novius business transformation framework (BTF)

In the case of Aegon, the BTF gave insight in the underlying causes of the appointed problems in Sect. 4.2.2, i.e., product-oriented and autonomous business lines, fragmented and outdated core systems, and poor program management. These underlying causes included lack of a common customer contact strategy, unclear responsibilities, and working from business line starting point rather than customer perspective. From an IT point of view, the system landscape in the contact domain was too complex, outdated, and fragmented across business lines. The core systems slowed down the business.

On the other hand, the "wow" direction simultaneously brought enthusiasm and energy to the organization, resulting in concrete examples of new digital capabilities for customer interaction. The wow initiatives instigated the belief that Aegon could bring these technologies to the customer and thereby seriously improve customer experience. Furthermore, reference visits and developing creative movies caused great excitement among Aegon's experts and direct stakeholders. One of the movies created also led to the naming of the transformation program. The transformation program was named Douma, referring to the typical Dutch family being customer of Aegon and representing the roots of the organization. The family persona as used in the Douma program, played a central role in the transformation program, where it provided a concrete image for which to develop the new customer services.

As a result of this preliminary research, the necessity for Douma became apparent, vivid, and concrete. It became clear that if Douma wanted to be successful, Aegon needed to excel in two critical dimensions: the *what* of the transformation (transformation capabilities) and the *how* of leading the transformation (leadership capabilities). In the next section, the transformation capabilities are outlined in order to become a customer-centric service provider, in other words the "what" of the transformation.

4.4 Capabilities to Realize Transformations ("What")

What kind of capabilities has Aegon realized to switch from a product-oriented insurance company to a customer-centric service provider? Implementing a variety of capabilities proved to be a 2-year transition toward a new integral way of working, specifically focused on customer contact. This chapter discusses five major transformations of Aegon to develop and implement these capabilities in order to successfully support the overall switch to a customer-centric service provider.

4.4.1 Transformation 1: Redesigning Services Around "Life Events" of the Customer

Instead of providing services for each of the business lines separately, services had to be organized and redesigned around "life events" of the customer. Life events are impactful events in the life of the customer, for example, the death of a family member. In case of such an event, the customer expects a "streamlined" administration process in which the insurer adjusts the appropriate policies and, if applicable, gives advice on payments that can be released as a result of the life event. During the Douma program, Aegon wanted to be noticed as a reliable insurer, providing relevant and integrated services. Starting from life events, a framework was developed to show the coherence with services, processes, people, competencies, and eventually IT. This framework shown in Fig. 4.3 proved to be a baseline to visualize the impact on various other aspects of Aegon.

Life events	Services	Processes & Governance	People & Competences	Functions	IT architecture
The life of customers is balanced around events in which insurances, pensions and banking services play a big role: giving birth, getting married, moving, retire, pass away.	Aegon proactively delivers services for each of these life events, such as product information, personal details, offers for a new insurance, damage report.	These services are the result of processes in which multiple parties are involved; Aegon governs and monitors the complete cycle from response to initial customer request until actual delivery of services.	The customer-to-customer (or end-to-end) processes are executed by competent employees who possess the appropriate skills and have the right attitude and behavior.	Processes and employees are supported by functions from the relationship domain, such as the management of customer data, the right customer view, the contact history, case management and management information.	The information function from the relationship domain is part of an overall IT architecture and collaboratively works with front-office media (telephone, portals, mobile, ...) and back-offices systems in the various business lines.

Fig. 4.3 Visualizing the impact of Douma on all aspects of Aegon

The framework ensured manageability of the complex situation and installed a common language to collaborate across business lines. At the same time, the framework helped in formulating guiding principles and to think in terms of customer needs and customer experience. The redesign of various services was much needed and ensured some quick wins for the organization.

4.4.2 Transformation 2: Standardizing Process (Working Methods) Around the Handling of Customer Requests

To effectively provide services around these life events also required uniformity in the way customer requests were handled. Case management was introduced to support this standardization in working methods and ensure all employees had the same information about a customer. Cases were focused on requests of customers and necessitated the collaboration with employees from different business lines, eventually assuring predictability and timely handling of customer requests.

Case management made it possible to store all sorts of information, ranging from customer documents to used articles, in a particular case, e.g., a customer. This made sure that all employees had the same information and could easily work across business lines. Also, new employees could directly get a total "view" of the case. Despite the "transferability" of cases, every case had one owner, making it possible to control for a very important KPI: on time handling of requests.

The integrated way of working was supported by one central system in which case management received a central place: Salesforce. The fact that approximately 1,700 employees of Aegon in the Netherlands (more than one third of all Dutch employees) had direct contact with customers showed the necessity for a central case management-based system. Soon, Salesforce yielded an enormous amount of generated customer information across business lines.

4.4.3 Transformation 3: Shifting the Organization Based on "End-to-End" Customer Processes

Working around the life events and standardizing processes while creating one central view of customer information were major capabilities to develop, but were not enough. In order to let the customer truly experience one organization instead of isolated business lines, Aegon needed to fundamentally shift the organization to end-to-end customer processes. Every customer process needed to have one process owner responsible for both the end result and customer interaction throughout the complete process, from request to receiving the end result by the customer.

Also of major importance was the shift of ownership of customer processes that exceeded a single business line. Responsibility for these processes was carried over from front-office to back-office. Up to the transformation, no one was formally responsible for the end result of "chained" processes and, so, the performance of

Aegon toward the customer. This trajectory started by allowing customer process ownership to arise on its own. In this way, there was no need for a major organizational shift, but this turned to be a step-by-step organical change. It eventually resulted in an implementation strategy to roll out the concept throughout Aegon.

4.4.4 Transformation 4: Generic KPIs Focused on Result for the Customer (Strategic/Tactical)

Integrating customer processes also required uniformly shared Key Performance Indicators (KPIs) to control for. Every business line was supposed to report on the same KPIs:

1. *Net Promoter Score*—This was measured from customer evaluations.
2. *First Time Right*—Has the customer received a fitting response/solution the first time?
3. *Process time/throughput time*—Has the customer received a fitting response/ solution within the agreed time interval?

Apart from defining KPIs, this also meant setting norms along the customer processes, adjusting dashboards, designing a reporting functionality to be able to easily report and analyze, and training managers to control based on the new dashboards.

4.4.5 Transformation 5: Designing and Implementing an Integrated CRM Solution

Apart from the deployment of Salesforce as the integrated solution to manage customer interaction, all related application and infrastructural work needed to be arranged. This required installing a distinct “pipeline” for data traffic to Salesforce as cloud solution, migrating customer data and past customer contact information. Securing storage of customer data in the cloud was top priority. Even as challenging was integrating Salesforce in the various workflow applications and insurance policy administrations. Eventually, two-way interfaces were realized, ensuring registration of customer requests, status changes, and contact moments wherever these were exchanged.

Thus, making customer interaction work comprised of more than implementing a CRM tool. It required professional IT capabilities and the ability to collaborate with procurement, legal, and security experts to ensure every aspect of a successful implementation of Salesforce. But by choosing a cloud-based CRM solution, Aegon exceeded the long existing boundaries within the insurance sector.

4.5 Leadership Capabilities ("How")

Securing and embedding such transformations could only succeed through advanced leadership capabilities, in other words the *how* of driving transformation. The "how" and "what" have been implemented parallel to each other. Shifting the organization to "end-to-end" customer processes and redesigning customer services needed to be embedded in the organization content-wise but even as much affected the attitude and behavior of all employees. Despite the scope of the Douma program and the complexity in changing the information systems, this did not pose the biggest challenge. This was the "internalization" of the new vision of the customer-centric service provider, the translation into innovative design choices and the mindset change of employees. Therefore, the management board of Aegon decided to set up a program in which business and IT would realize the transformation together. Figure 4.4 shows the milestones, at a global level, of the Douma program (2013–2015).

This chapter focuses on the business aspect of the transformation. Nonetheless, the transformation process is as applicable to IT as well as to the business itself. The following subsection describes the key choices leadership of Aegon had to make in order to not only realize the transformation but also to sustain it. In other words, how could the leadership of Aegon prevent the organization from falling back into old behavior? These key choices could be positioned as a set of choices determining the design and implementation of the Douma program and toward the successful transformation of Aegon.

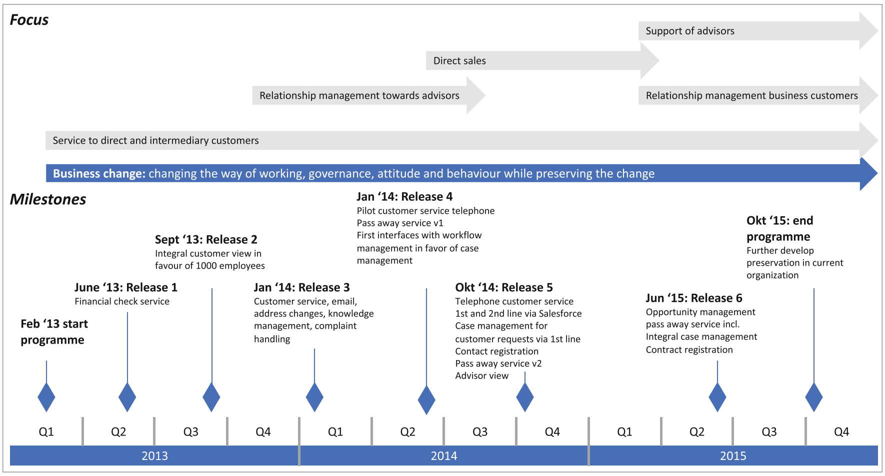


Fig. 4.4 Milestones of the Douma program

4.5.1 Key Choice 1: What Approach Do You Choose to Transform?

The transformation Aegon stood for was very complex. The transformation had an effect on almost all business lines and aspects of the organization while integrating new technology in the existing application landscape. Some of the fundamental choices Aegon had to make lied in the suitable approach to achieve success in such complexity. As well as how to effectively utilize a Agile/Scrum way of working, since these methods where not yet used widely within the organization at the start of the program. How much expertise would Aegon have to bring in and how much experimenting would Aegon allow?

Program management of Aegon adopted the term “controlled trekking,” a trekking to symbolize the uncertainty what Aegon would find on its “route” but nonetheless controlled, with a clear vision and dot on the horizon. The first steps in every cycle were planned but leaving all subsequent steps intentionally blank. Along the route, Aegon made sure that this approach could not affect pace, capacity, and effectiveness.

Furthermore, the business had significant influence on prioritization of the road map, as the core team had a responsible program manager and manager responsible for both business and IT. The business took on the role as product owner. Collaboration between program manager, business, and IT was essential for a balanced performance, support, and decisiveness. The approach needed to be further explored and became based on five beliefs how the Douma program had to be deployed.

4.5.1.1 Belief 1: Encouraging Trust Instead of Investing in the Business Case

Aegon’s CEO, also the executive program sponsor, was truly convinced of the need to transform to a customer-centric service provider and took a prominent role in providing trust and motivation to achieve the desired results. Also due to the involvement of the CEO, necessary commitment of the top was arranged from the start.

4.5.1.2 Belief 2: Frequently Delivering Business Value

The core team was convinced that frequent delivery of business value, also directly from the start, was a success factor. This also meant developing services while the base functionalities were not yet fully worked out. It ensured growing support in the business from day one.

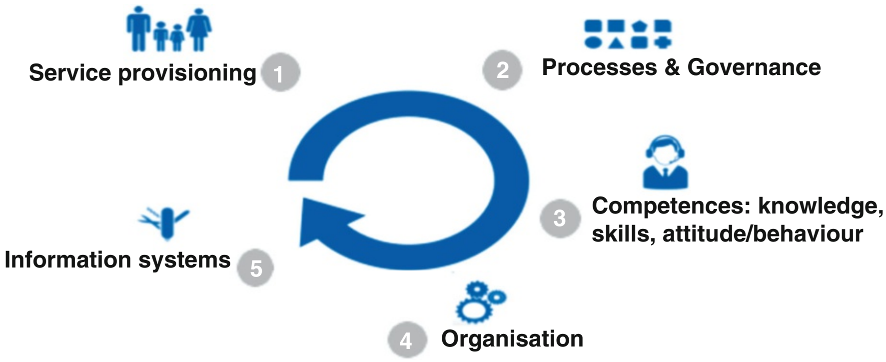


Fig. 4.5 Douma integrating IT and business aspects

4.5.1.3 Belief 3: Implementing CRM Is a Business Initiative, Not an IT Program

Another conviction was in scoping of the program, which had to be positioned as a business initiative that put the interest of the customer and business first. Therefore, the CRM implementation became managed in an integrated way and not approached from an IT perspective. This is visualized in Fig. 4.5.

4.5.1.4 Belief 4: Embracing Agile/Scrum and Encouraging Support

The Agile philosophy and Scrum methodology were at the time of the transformation not widely embraced within Aegon. Choosing for this new way of working implied a serious investment of time and money, which the program MT also did, especially in the first year. Agile and Scrum contributed to the continuous alignment between business and IT and a much more clear role division.

4.5.1.5 Belief 5: Equally Focusing on IT Skills as well as Attitude and Behavior

IT skills and knowledge were considered equally important. The employee should know how to work with the new tools but also why. In every customer interaction training, the why of the transformation was discussed. Eventually, the employee should be able to acknowledge the customer's feeling, helping the customer with the right tone of voice, in a proactive, right, and timely manner. This should also prevent from having to repeat the customer's request in the future.

4.5.1.6 Belief 6: Ensure that the Program Activities Will Be Continued in the Line Organization

To truly anchor the switch toward a customer-centric organization, the program MT put in place a new department, customer interaction management (CIM), which continued the program activities in the line organization. This activity made clear that this transformation was to be maintained and embedded in the organization and that it would sustain once the Douma program was over. The CIM showed to be the link between marketing and business development (customer focus) and operations.

4.5.2 Key Choice 2: How Do You Organize Ownership—Including Governance?

In order the secure transformation, organizing ownership was a precondition. But were the responsibilities in the Douma program clear and how to distinguish between the "right" ownership for a specific aspect? How to involve stakeholders in the shift of ownership across parties?

Organizing ownership in complex transformations has many aspects. One important aspect is involvement, here in particular the management involvement in Douma. During the transformation, the management board showed adequate involvement and communicated the new "customer-centric" vision to all stakeholders. Most challenging however was the involvement of management in the business lines. The program organization, and the involvement of the business lines in the transformation program, is visualized in Fig. 4.6.

4.5.3 Key Choice 3: How Do You Combine Design and Development at the Same Time?

A good combination of design and development was key to transformation success, because it can lead to solutions, that can rely on a solid basis and company support. This starts with a solid understanding of both concepts, which are explained below in more detail.

Design is mostly about shaping the content. The focus is on a systematic analysis and a future description of the organization and its information systems. This provides insight, overview, and quality. Therefore, design focuses more on the "hard" side of the transformation. *Development* is mostly about initiating a transformation process, in order to connect people within the organization with the goals of the transformation and to effectively and enthusiastically contribute to achieving the results. Development also includes creating learning experiences and increasing competences so that people are empowered to take ownership.

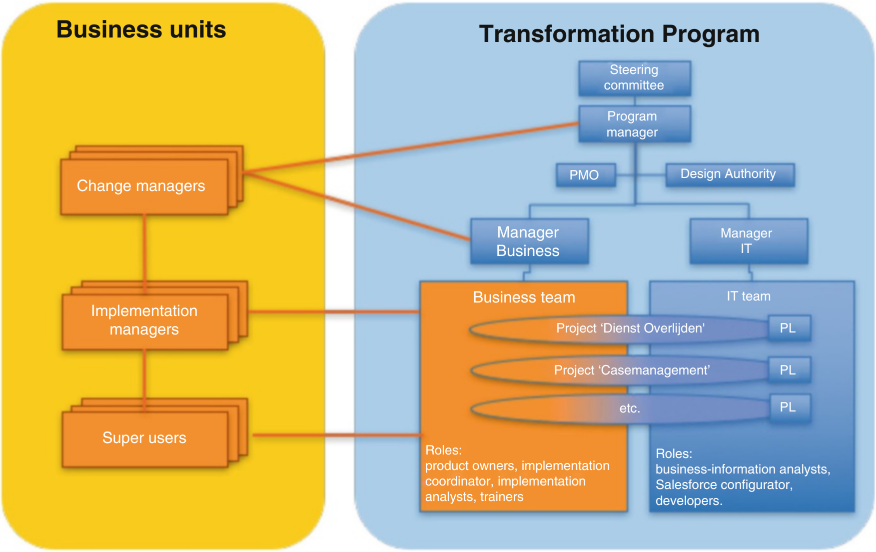


Fig. 4.6 Interaction between business lines and program organization

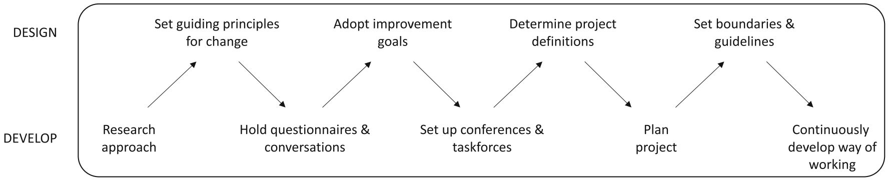


Fig. 4.7 Combining design and development

Development thus focuses more on the "soft" side of the transformation. Figure 4.7 illustrates the design and development concepts and how they influence each other in achieving transformation.

In this case, the combination between design and development was made in various forms. How did Aegon put this in practice? The design and development paradigms perfectly represent the two directions of the Aegon transformation process, respectively, the blue and wow approach. Key principle is not about deploying both directions separately, but by letting them interact. In general, without enthusiasm and energy, the new principles, goals, and guidelines will not be adopted by the employees, but without principles, guidelines, and a clear goal to actually guide employees, the "wow" initiatives might be experienced as an "empty shell."

4.5.3.1 Guiding Principles as Dot on the Horizon

At the start of the program, a number of guiding principles were designed together with the stakeholders. These principles guided the transformation program. The guiding principles helped in defining scope and setting requirements and literally guided the decision-making process of the steering committee.

4.5.3.2 Developing Basic Functionalities and Services Simultaneously

From the start, services were redesigned and in parallel, basic functionalities were created. Simultaneously, complex integrations with existing information systems came by. Nonetheless, quick results were preferred because of the effect and commitment it would yield on customers and employees. The risks of rework were taken for granted. Within 6 months, a customer view with more than three million customers was ready, and within 3 months the first service (*financial check service*) went "live."

4.5.3.3 Empowering Employees by Developing Their Change Competences

The complexity of the transformation rested on the competences of employees. Thus, employees got to work on improving their set of competences, for example, business analysts who were challenged to define inspiring visions and service descriptions as a starting point for business analyses or by creating a uniform approach for business analyses and design to work across business lines. Product owners were given the task of stakeholder management, where a design authority was to oversee the various design choices made across business lines. All employees took on an Agile way of working, although commonly unknown up till that moment.

4.5.4 Key Choice 4: How Do You Implement a Generic Working Method Across Multiple Business Lines?

The previous key choices have elaborated the decision for a transformation approach, how to effectively create ownership and how to combine design and development within a transformation program. The final key choice that the, which management considers is the way how to implement a generic working method across multiple business lines. As Aegon's six business lines have been isolated for long and cross business-line collaboration was a driver of program success, this key choice was of particular importance to Aegon.

4.5.4.1 Implementing Is About Inserting and Sustaining

Implementation was more than only "inserting" the transformation. Inserting is about creating the employees' "will and skill" for change, while "sustaining" the transformation is about "doing and keep doing" what was intended. The Douma implementation team had the core responsibility to insert the transformation, but for sustaining the transformation, this was up to the line organization in the form of change managers. They were to monitor the adoption of the new way of working and the transformation's vision in its own business line. Clearly, situations arose where the uniform way of working proved incompatible with the business lines' processes. Based on impact analyses, the program's MT would then decide to either accept the consequences or adjust the way of working.

4.5.4.2 Organizing the Right Roles and Collaborations

Aegon made the key roles in the program clear. It distinguished between the roles and responsibilities within each business line as well as in the overarching transformation. The manager of the business had a central position in the transformation, accompanied by an implementation team who would generally manage the implementation throughout Aegon and deal with business line exceeding issues. Within the various business lines, implementation managers were also active, together with an operational manager to sustain the transformation in each business line.

However, collaborations between business lines and the Douma program were not that easily set up. The business lines were so used to being self-sufficient that sharing insights and good practices did not occur spontaneously. The program had a facilitating role to create a more collaborative way of thinking and acting. By organizing weekly implementation of managers' meetings, transition meetings, and activities focused on embedding the transformation in the organization, this collaboration was also enhanced in a more formal sense. It was up to the program to set up collaborations across business lines who had mutual benefits.

4.6 Outcomes of the Transformation

Aegon has put a major step toward becoming a customer-centric service provider. But as one can imagine, does it take years before such a transformation has been truly adopted by the organization? Nevertheless, Aegon has made an undeniable switch and has this integrally secured in their governance and working methods.

Below, the results and effects that Aegon achieved with this program are discussed, together with the key lessons that other organizations can learn from this case.

4.6.1 Results

The following results show the key findings of the transformation:

- Aegon has reached out to all customer groups, ranging from customer relationship management to sales support to advisors.
- Aegon has implemented the first business line overarching services such as "financial check," "payment delay," "death," "complaints," and "change of address."
- Aegon has created a powerful but easy to use CRM functionality, including one Aegon-wide customer view, advisor view, integrated opportunity management, case management, contact recording, reporting, and support for phone/email/chat and social.
- Aegon has ensured the same information is available in all customer contact channels, ranging from telephone to email, web, and chat.
- The customer-oriented thinking is adopted by 1,700 users across all 6 business lines, from the contact centers and business lines to the staff departments. Awareness has raised for the value of registering the right customer information visible for the customer and colleagues.
- Various trainings related to customer interaction have been added to the HR portfolio, such as "customer skill," "Salesforce knowledge," "case management," and "steering with reports and customer communication."
- Aegon has positioned "customer interaction management" in a newly added business function.

4.6.2 Effects

As results are more visible, the most important can be considered—how the transformation of Aegon has been perceived by customers, advisors, and employees. What can be seen now what Aegon has achieved on the longer term?

More than ever before, *customers* now receive an answer or solution that fits their situation, as employees now have more information available about the customer's situation and the status of requests. Status information is also increasingly available. The ultimate impact of the program on the customer is visible in the Net Promoter Score (NPS). This is obviously influenced by many other factors, but it can be no coincidence that the NPS showed the biggest improvement in years. The improvements toward advisors are also noticeable because Aegon employees are better informed while more relevant information, such as leads, is shared with advisors.

Because customer information is located in one place, *employees* experience that they have acquired a powerful tool. Aegon is the first insurer who records "social identities" and "social cases" in a customer view. As a result, customers can be recognized faster, requests can be anticipated upon, and consequently customers can be assisted on time. This makes it more intuitive to serve customers “right the first time” and to work together across business lines.

The *managers* experience a positive twist too; they have gone from product-oriented mindset to prioritize and control from a customer perspective. With the new dashboards, it has become possible to control and improve the level of quality of the customer service. In addition, a great deal of what happens—also failures and bottlenecks—are more transparent, resulting in faster improvements for Aegon.

The Douma program was not supported by a very detailed and fully worked out business case. As mentioned, it was started from the belief that Aegon should become more relevant for customers. Nonetheless, the higher NPS also benefited Aegon financially. The program proved to have more financial benefits, as the improved lead and opportunity management created more sales opportunities than ever before. And because customer requests are increasingly handled right the first time, less rework is necessary. Finally, systems have been phased out and processes optimized, with major cost reductions as a result.

4.7 Conclusion

The whole transformation program proved to be a complex task and required drastic changes in many areas, from services, processes, and IT to governance, competences, and attitude from employees. Overall, a solid balancing of the structured blue approach and the innovative and open wow approach proved to be a key success factor of the program. To advance on that, this chapter shows ten concrete lessons learned from the transformation program:

1. *Hold on to your success factors and preconditions.* Key is to monitor these success factors tight and report progress to initiate and maintain enthusiasm.
2. *Let a major program set the pace of transformation, but ensure the organization can keep up.* Key is to balance the pace of the program with the organization who has to adopt the transformation. Regularly assess whether the stakeholders are still involved.
3. *Identify the ability to change (capacity and potential) and align this between the steering committee and executive sponsor.* With Aegon, the product-oriented focus (isolated business lines) was more persistent than originally thought. Along the way, the program was regularly confronted that coherence was lacking across the business lines, despite being crucial for the success of the program. Take sufficient time to deal with such obstacles that potentially threat its success.
4. *Let the line organization in charge to truly create ownership.* Ownership will never exist if management takes over once things go wrong. Let teams indicate their struggles and let them make mistakes. Realizing such a transformation requires patience.

5. *Hold on to the chosen approach, especially at times of resistance.* A major program such as Douma naturally gives rise to resistance. Such programs know various difficult stages. The major challenge is to keep attention on success but be aware of negative opinions and listen, as they might prove a basis for small improvements.
6. *Let C-level play a prominent role in such programs.* Aegon's CEO was chair of steering committee and thus implicitly showed how important the Douma program was for Aegon.
7. *Make a transformation program tangible.* In Aegon's case, Douma was not chosen for nothing. It had strong ties to Aegon's history and represented a family to develop new services for.
8. *Effective collaboration determines the organization's power to execute.* Although it seems obvious, collaboration between all stakeholder groups (not only business and IT) is of utmost importance. By leaving enough room, it then becomes a "shared adventure."
9. *A well-thought-out and balanced governance structure is worth investing in.* Aegon not only put in place a balanced program MT but also created ownership for the transformation at change managers, sponsorship with the CEO, and power to implement with the use of implementation managers.
10. *Create mandate to oversee design choices across business lines.* As the focus on the sole business line was a potential threat for implementing the same service in various ways, a design authority was to keep overview of the use of design choices. The design authority was led by the program architect.

Chapter 5
How Blockchain Technology Affects the Performance of Financial Services

Martin Smits, Hans Weigand, Joost de Kruijff, and Jerom de Valk

Abstract This chapter explores how the application of blockchain technology may enhance business network performance in financial services networks. We apply a recent detailed definition of "blockchain technology" that distinguishes between 12 aspects of blockchain technology and 11 aspects of the application of blockchain technology, based on enterprise ontology theory. We combine the ontology-based definition with business network theory and apply it on the literature-based case of the Australian Stock Exchange. We conclude that blockchain technology implementations affect the structure of business networks (since the technology influences which firms (do not) participate), the transactions and network processes, as well as the communication logic, content logic, consensus logic, and contract logic. We summarize our observations in five simple, practical questions to help predicting blockchain technology success.

5.1 Introduction

Recently, blockchain technology emerged with the potential to fundamentally change businesses and industries (Swan, 2015; Williams et al., 2016). Blockchain technology enables secure exchange of value over the Internet, without the need for a third party like banks or insurance firms. The technology works as a shared ledger system where (in its classical form) every node possesses a complete copy of the

M. Smits (✉) · H. Weigand
Tilburg School of Economics and Management (TiSEM), Tilburg University, Tilburg, the Netherlands
e-mail: M.T.Smits@tilburguniversity.edu; H.Weigand@tilburguniversity.edu

J. de Kruijff
Tilburg University, Tilburg, the Netherlands
e-mail: J.C.deKruijff@tilburguniversity.edu

J. de Valk
PwC, Amsterdam, the Netherlands
e-mail: jerom.de.valk@nl.pwc.com

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_5

ledger and agrees on changes. Application of the technology results in records (in the ledger) that are immutable, driven by consensus, secure, and most important: a single, automated, source of truth.

Blockchain, or distributed ledger, technology can be used for any form of registry, inventory, and exchange, as explained by—pseudo identity—Nakamoto (2008). The technology enables new ways of governance in decentralized organizations that operate without human intervention over an IT-enabled business network. These new organizational forms may result in huge cost savings and transparency gains (Swan, 2015; Smits & Hulstijn, 2020). Transparency gains also seem promising from an audit and regulatory point of view. Blockchain applications have led many to compare this technology to the Internet, with accompanying predictions that blockchain will shift the balance of power away from centralized authorities in the fields of communications, business, and eventually even politics and law (Wright & De Filippi, 2015).

A key question that remains is how blockchain technology applications enhance business network performance. A lot of uncertainty exists about the blockchain concept and its ramifications for businesses and industries (Pilkington, 2015). The research objective of this explorative chapter is to identify the impact of blockchain technology on business networks and how such impact can be assessed. We focus on the impact on financial services networks, and our research question is "how can blockchain technology enhance the performance of financial services networks?".

We use Smart Business Network theory (van Heck & Vervest, 2007) as the explorative perspective for the impact of blockchain technology. We choose this theory since it explains how technology affects the network structure, processes, and performance. The theoretical relevance of this paper is that it contributes to the Smart Business Network body of knowledge by elaborating some unclear aspects of the theory. We also contribute to the understanding (the ontology) of blockchain. This chapter is also interesting for practitioners since it provides insight in how blockchain design choices and implementations affect the business network and its performance.

The remainder of this chapter is organized as follows: In Sect. 5.2, we first define and provide an ontology for blockchain technology. In Sect. 5.3, we summarize Smart Business Network theory (van Heck & Vervest, 2007) and hypothesize how blockchain technology affects business networking (Sect. 5.4). In Sect. 5.5, we use the theory to explore the impact of blockchain technology in the Australian Security Exchange (ASX) case. Finally, we answer the research question and discuss the relevance of blockchain-enabled business networks in the financial services industry.

5.2 Blockchain Technology Definition and Ontology

Blockchain technology has been defined as “a distributed ledger technology that can identify participants, automatically execute transactions, and provide a platform to support advanced functions and business logic known as smart contracts” (Williams et al., 2016). Distributed ledger technology is a relatively new phenomenon but based on a number of established technologies in novel ways:

1. *Blockchain*—a secure record of historical transactions, collected into blocks, chained in chronological order, and distributed across a number of different servers to create reliable provenance
2. *Digital signatures*—unique digital keys used to authorize and check transactions and to identify the initiator
3. *A consensus mechanism*—rules and techniques to ensure that participants recording and processing transactions agree on which transactions are valid
4. *A digital currency*—in some implementations, a cryptographic token that represents actual value. Bitcoins are one example, but ultimately central banks could create digital fiat currencies as well.

Bitcoins are one example, but parties can create other tokens as well.

Note that these definitions describe blockchain technology not only as the technology itself but also as the application of the technology to execute transactions in a business network setting. To clarify blockchain technology, de Kruijff and Weigand (2017) define blockchain technology in more detail following enterprise ontology theory (Dietz, 2006). Enterprise ontology theory describes an enterprise as a heterogeneous system consisting of three layers (see Fig. 5.1). Applied to blockchain technology, the datalogical layer describes blockchain transactions at the technical level in terms of blocks and code. Second, the infological abstraction describes the blockchain transactions effectuating open ledger system. Finally, the economic meaning of the transactions is described at the essential (or business ontology) layer.

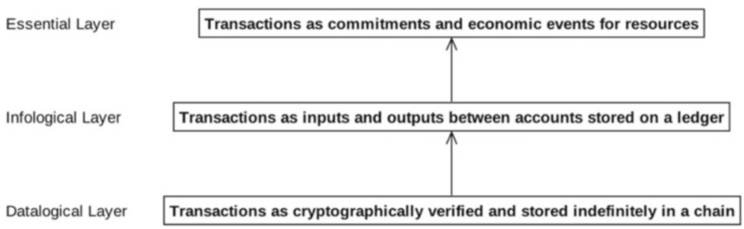


Fig. 5.1 Blockchain technology defined at the technical level (datalogical layer), the data level (infological layer), and business implementation layer (essential layer)

At the datalogical level, de Kruijff and Weigand (2017) define blockchain technology by the following 13 classes:

1. *Actor*—A virtual ID (for any individual or organization) that owns a wallet.
2. *Wallet*—A wallet initiates transactions on the blockchain and receives the transaction output.
3. *Transaction*—A request to the blockchain nodes that contain an input, amount, and output (blockchain) or custom data like code (altchain). Verified transactions provide a proof that there was authorization to interact with the system.
4. *Node*—An entity in the blockchain network that either proofs (public transactions) or validates (hybrid or private transactions) and subsequently adds it to a block with a unique hash. The hash will be used as input by the next transaction. Nodes receive rewards for every successful transaction that is added to the block.
5. *Miner*—An anonymous node (e.g., server) that cryptographically proofs a public transaction to be valid using a proving mechanism like proof of work, proof of resource, proof of state, proof of activity, etc.
6. *Mining mechanism*—To mine transactions in public blockchains, altchain, or sidechain.
7. *Validator*—A nonpublic node that (cryptographically) validates hybrid or private transactions based on validation mechanisms like byzantine fault tolerances or double spending.
8. *Validating mechanism*—To validate transactions in nonpublic blockchains, altchains, or sidechains. An example of a validation mechanism is a byzantine fault tolerance mechanism.
9. *Block*—A transaction container with a unique block header, which cryptographically commits to the contents of the block, a timestamp, and the previous block header.
10. *Uncle*—A block that is very close to being the "correct" next block in the blockchain. By mining and rewarding for uncles, the proofing process becomes heavier and more reliable.
11. *Cousin*—A block that is very close to being the "correct" next uncle in the blockchain. By mining and rewarding for cousins, the proofing process becomes heavier and more reliable.
12. *Runtime (or cryplet)*—Enables secure interoperation and communication between blockchain middleware and third-party clouds like Microsoft Azure, Amazon AWS, and others.
13. *Middleware*—Software included in the blockchain and enables third parties to interact with blockchain records to provide services like identity management, data analytics, smart contracts, and connections to widely used cloud software like Office 365 and Exchange.

At the infological level, the blockchain contains the following seven classes (de Kruijff & Weigand, 2017):

1. *Chain*—the highest level of abstraction for a combination of blocks.

2. *Mainchain*—a digital ledger that contains the block headers of all blocks that are digitally signed and containing validated records of ownership that are irreversible, depleting the necessity for the reconciliation of data. A blockchain that is deployed as a service contains middleware and a runtime (or cryplets).
3. *Blockchain*—refers to the Mainchain implemented according to the Bitcoin codebase.
4. *Altchain*—refers to a mainchain implemented according to an alternative codebase, like Ethereum, Tendermint, Eris, or List. Nowadays, over 600 altchains or alternative digital currencies exist.
5. *Sidechain*—a chain that allows for the transfer of assets between the sidechain and the mainchain. The benefit of a sidechain is that it can store assets and data that cannot be saved (or is too expensive) on the mainchain and may increase the transaction speed significantly by using pre-mined mainchain addresses.
6. *Drivechain*—a sidechain that provides a two-way peg (2WP) that allows transfers of a cryptocurrency from a mainchain to another mainchain (and vice versa) requiring low third-party trust.
7. *PeggedSidechain*—a sidechain that enables assets to be moved between multiple mainchains, thereby illuminating counterparty risk, enabling atomic transactions (transaction happens all together or not), enforcing firewalled chains, and making chains independent from each other.

At the essential (business) level, the blockchain contains the following four classes (de Kruijff & Weigand, 2017):

1. *Digital ledger*—maintains a continuously growing list of transaction records called blocks. Each block contains a timestamp and a link to a previous block.
2. *Account*—sends and receives value to and from a transaction.
3. *Transaction*—is an end-to-end mainchain transaction as depicted in the datalogical ontology.
4. *Journal*—is list of transactions.

Summarizing, the technology itself is defined by 12 concepts (datalogical level), and the implementation of the technology is defined by 11 concepts: 7 concepts for "chain" (the infological level) and "digital ledger, account, transaction, and journal" (the business level). This ontology is intended to understand applications of blockchain technology to automate transactions in business networks. We now proceed and summarize business network theory as additional lens to analyze the business network effects of a blockchain technology application.

5.3 Smart Business Networks

Rather than viewing the business as a sequential value chain, firms in a Smart Business Network make linkages that are novel and create better than usual results. Business networks that effectively use technologies and outperform competing networks are known as Smart Business Networks (van Heck & Vervest, 2007). The notion of "smartness" is reflected in the ability to quickly realize "scenarios in which business is conducted in a flexible business network that executes transactions automatically by firms that participate in the network" (van Heck & Vervest, 2007, p 29). The critical organizational capabilities for a Smart Business Network are:

1. The ability to connect and disconnect with (new) firms
2. Effective automated selection and execution of business processes in the network
3. Establishment of standards and embedded operating logic within the business network

Not only the "connect capability to other firms" but also the "disconnect capability" must be fully supported: firms that decide to leave the network must be able to disconnect (and reconnect to other networks). If connected to a network, and if selected to participate in a business transaction, firms must be able to interoperate ("pick, plug, and play") with other firms in the network using Enterprise Service Bus technology and BPEL-like service compositions. The idea is that the business logic of the network is enabled by a Networked Business Operating System (NBOS) layer, rather than be scattered over the many different companies (in the information layer) with their own silos. This NBOS layer (or logic layer) allows process execution and management "from a distance" from the underlying application systems.

Figure 5.2 illustrates smart business theory by distinguishing three network layers (we renamed the layers originally introduced by van Heck and Vervest, 2007). The bottom layer is the physical layer, representing the logistic processes between

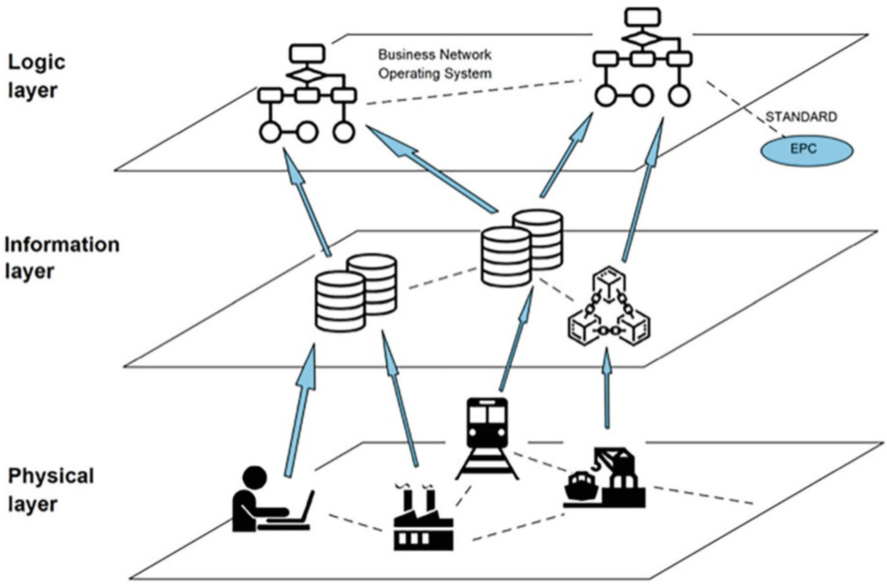


Fig. 5.2 The smart business network model (adapted from van Heck and Vervest, 2007)

the firms (actors) in the network. The information layer represents the transactions between firms, stored in information systems (within firms) or shared ledgers (shared between firms). Note that these shared ledgers may include all transaction data, or only limited data sets, like some financial transactions, personnel, resources, etc. Smartness of the network may refer to finding the right amount of data (not) to be shared in shared ledgers. Finally, the third layer (logic layer) includes the business logic (NBOS, BPEL, ESB, smart contracting logic, etc.).

Smart Business Network theory hypothesizes that the "performance of smart business networks depends on effective use of a shared technological infrastructure." This infrastructure consists of a network platform with a Networked Business Operating System (NBOS) that serves information sharing over and with network partners (van Heck & Vervest, 2007, p. 30), in particular network transactions to enable network logistics. Organizations not only rely more and more on the business networks they participate in but also move from traditional relatively stable and slow-moving business networks to an open digital platform "where business is conducted across a rapidly formed network with anyone, anywhere, anytime despite different business processes and computer systems." The business is no longer a self-contained organization working together with closely coupled partners: "it is a participant in a number of value networks where it may lead or act together with others." Firms in the network act individually according to the rules of the network: "network participants become a smart insect in a goal seeking swarm" (van Heck & Vervest, 2007). Note that a business network can be a network of divisions or departments within one large firm, or an interorganizational network in which (many) firms participate. Obviously, blockchain technology may provide solutions for storing and executing transactions that can be completely self-governed or automated. This change is on the information layer, but also effects on the other layers may be expected, as elaborated in the next section.

5.4 Impact of Blockchain Technology on Smart Business Networks

We now hypothesize effects of blockchain technology (following the datalogical, infological, and essential layers defined in Sect. 5.2) on the information, physical, and logics layers in business network theory (as defined in Sect. 5.3).

5.4.1 Information Layer

The information layer (the middle layer in business network theory) is where data on transactions are stored in either internal (not shared) information systems of individual firms or in (distributed) shared ledgers. Where transactions between organizations used to be stored by each organization internally (represented by separate data silos in the information layer), transactions can also be stored now only once externally in a blockchain ledger. We use the word "transactions" here, in line with the smart business network model (van Heck & Vervest, 2007), to refer to the interactions between organizations, with economic transactions of sales and payment at the core. This includes orders and order commitments (commitment-type transactions), as well as payments and deliveries (execution-type transactions).

Internal transactions within the company can be stored in a private (local) blockchain. When transactions are stored (internally or externally) in an irrevocable way in a blockchain, this not only eliminates duplications (data redundancy) but also the related inconsistencies. Another effect of the externalization of data into the shared ledger is mitigation of data heterogeneity. Standards and data ontologies will still be needed, but their reach and effect at the network level will be much stronger, as they are not only used for exchanging data but also for storing the data.

5.4.2 Physical Layer

The physical layer represents the firms (including intermediaries) and logistics involved in the network. From an organizational perspective, blockchain-enabled transactions will affect the intermediaries. In particular, intermediaries supporting information exchange or trust will be threatened, but this may depend on the type of service offered by the intermediary. Search intermediaries may not be affected. Trust intermediaries may be affected if the basis for trust shifts to blockchain security. Information exchange intermediaries may be affected because blockchain aims for single point of storage.

Transactions are related to movements of goods at the physical layer. An important development at this layer is the *Internet of Things* (IoT). More and more business resources are equipped with IoT devices and in this way integrated into the Internet. Following the IoT reference model of Bauer et al. (2013b,a), we can say that *physical entities* get a representation by means of *virtual entities*. The physical entity is extended with a device that contains software resources. The virtual entity, with a unique URI, has a service interface to its software resource.

A second relevant development is *servification*. As has been argued in the service science literature, there is an evolution from a goods-dominant logic to a service-dominant logic. This not only means that the services sector grows in economic significance but also a shift from the emphasis on control (ownership) of resources to the use of resources (access right). In this development, there is, for instance, less a need to own a car, if you can have a car, or a taxi service, when you need it. These two developments reinforce and are reinforced by blockchain technology: blockchain transactions can be used to transfer money (Bitcoin) but also to transfer access keys for digital products (software and e-books). In the same vein, it can be used to transfer ownership rights on registry goods like houses and ships. Obviously, the transaction cannot transfer a physical resource. However, when physical entities are accessible by IoT services, they become more like digital goods. When the goal of the transaction is to deliver a service, then logistic movements become less important than access rights.

It remains to be seen to what extent ownership transfers can be turned into services and to what extent the access to these services can be mediated by IoT. Perhaps blockchain transactions cannot govern all exchanges on the logistics, physical level. Still, it is to be expected that blockchain transactions will (not only record but) govern a large amount of economic exchanges. This will have an effect on operational efficiency (less human effort in the loop) and control efficiency (external control by IT replacing internal control). Together with the savings at the information layer, this will cause significant savings in transaction costs that in turn may also affect the institutional structure (Electronic Market Hypothesis).

5.4.3 Logic Layer

Blockchain transactions can be embedded in smart contracts that are executed automatically by the blockchain infrastructure. At this moment, the smart contracts are still in their infancy, but in principle, there is no computational limit to their scope, and a smart contract could take on the orchestration role that the NBOS plays in the Smart Business Network model. However, as stated in the above, the NBOS typically assumes an orchestrator function—technical and organizational. These roles will be less prominent and may even disappear in a blockchain-enabled environment where the goal is self-governance.

We use the 4×4 model (Birch et al., 2016) to analyze the new logic layer in a blockchain-enabled Smart Business Network. The 4x4 model distinguishes four types of logic:

- *Communication logic*—This logic determines the logic for communication between participants in the network. Blockchain technology is mostly used in three different forms: public, hybrid, and private. For all three forms, three properties are of particular relevance: the number of nodes, access to read the blockchain, and access to write the blockchain (Brennan & Lunn, 2016):

  1. *Permissionless Public Blockchain (Public)*—Distributed; anyone can read and write on the blockchain, as long as they meet certain criteria and follow the specified rules. This type of blockchain is entirely distributed and a single source of truth and has entirely trustless integrity. A well-known example is the Bitcoin blockchain.
  2. *Permissioned Public Blockchain (Hybrid)*—Decentralized; only permissioned entities may write the ledger, but anyone may view the content. This results in greater accountability and transparency. This form shows great potential in the financial services sector.
  3. *Permissioned Private Blockchain (Private)*—Decentralized; only permissioned entities can read and write on the blockchain. This form is mostly used

in experimental settings where R&D is the main purpose of its existence. A well-known example is the R3CEV[1] consortium.

- *Content logic*—The types of assets that are distributed over the network. On a blockchain, many different types of assets can be transferred, like cryptocurrencies, letters of credit, or stock bonds. This means that token value can be simply information or representative of extrinsic value or have intrinsic value. It is also possible to configure multiple kinds of assets on a single blockchain.
- *Consensus logic*—To ensure that only legitimate transactions are added to the blockchain, the participating nodes in the network use voting to confirm that new transactions are valid. A new block of data will be added to the blockchain only if miners in the network reach consensus as to the validity of the transaction. Consensus can be achieved through many different voting mechanisms. The most common is proof of work, which depends on probability through the amount of processing power donated to the network (Wright & De Filippi, 2015).
- *Contract logic*—Also defined as the automation logic; the way that transactions are animated to trigger events. Using blockchain technology, parties have the possibility to confirm that an event or condition has in fact occurred without the need for a third party. A well-known application is a "smart contract": a computable contract where the determination of performance and enforcement of contractual conditions occur automatically, without the need for human intervention (Wright & De Filippi, 2015).

Each of the types of logic can be modified to optimize the operating logic and to achieve different business objectives (Birch et al., 2016).

5.4.4 Propositions

Based on the literature in the previous sections, we hypothesize that *blockchain technology ultimately affects business network performance* by causing changes in:

1. The information layer by introducing general ledgers
2. The business operating logic (logic layer) by enabling several forms of business logic
3. The physical layer by affecting the business network structure and business network processes

Thus these lead to the following propositions:

- *Proposition 1*—The use of blockchain technology leads to changes in the business operating logic.

[1] https://en.bitcoinwiki.org/wiki/R3.

- *Proposition 2*—The use of blockchain technology leads to changes in the structure of business networks.
- *Proposition 3*—The use of blockchain technology leads to changes in network processes.

We now use the theory and propositions above to explore the impact of blockchain technology in the Australian Security Exchange (ASX) case. We selected this case because much information is available via desk research. We use (ASX, 2016) and several other documents, indicated below, as our sources for the ASX case. Since the aim of our study is exploratory, we do case-based research (Yin, 2009).

5.5 The ASX Case

The World Economic Forum distinguishes six categories for using blockchain technology in the financial industry: payments, insurance, deposits and lending, capital raising, investment management, and market provisioning (World Economic Forum, 2016). In order to analyze the impact of blockchain technology, we selected the Australian Security Exchange (ASX) case. The ASX case focuses on market provisioning and investment management, more specifically, on equity post-trade, meaning the processing of security trades after the trade execution at stock exchanges.

The creation of integrated and efficient capital markets is among the most important and ambitious projects currently running worldwide. Developed, effective, and reliable capital markets are needed for many important reasons like the financing of economic activities and providing attractive alternatives for investors' savings. An important aspect is the finalization of securities transactions ("clearing and settlement"), where the buyer ultimately receives the purchased securities and the seller the agreed amount of money. Today, the equity post-trade process consists of a couple of main steps which are briefly illustrated in Fig. 5.3. The parties involved in this process cooperate in a business network with the aim to process the incoming transactions efficiently, effectively, and safely (Toppen et al., 1998; AFME, 2015).

5.5.1 *Introduction to the ASX Case*

ASX collects, manages, transforms, and disseminates a wide range of data on the Australian equity and derivatives markets, clearing houses, and settlement systems. Figure 5.4 shows how 2 million investors (firms) are connected to 77 brokers, 37 clearing participants, and 98 settlement participants, all supported by ASX to enable the trading, clearing, and settlement processes. The ASX business network uses the CHESS (Clearing House Electronic Sub-register System) for a variety of services. Over the years ASX and CHESS have been tailored to the needs of a variety of

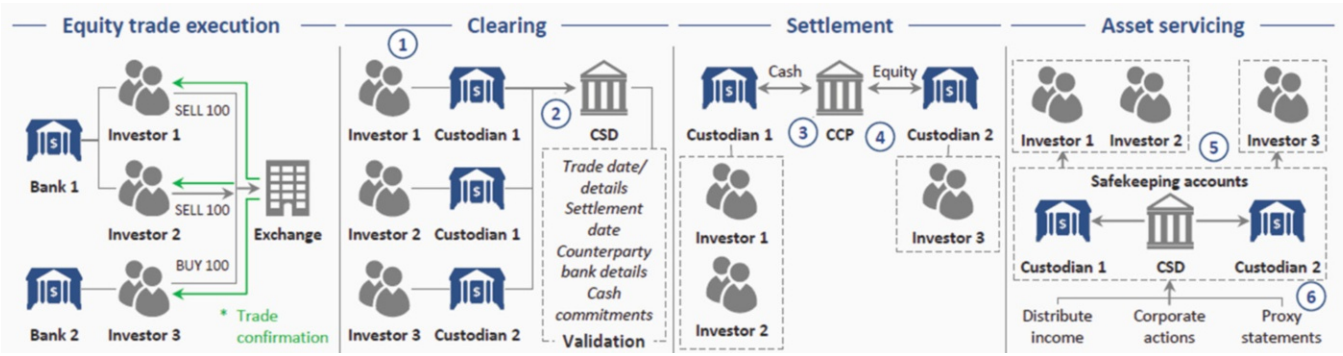


Fig. 5.3 The securities transaction process, consisting of equity trade execution, clearing, settlement, and asset servicing (World Economic Forum, 2016)

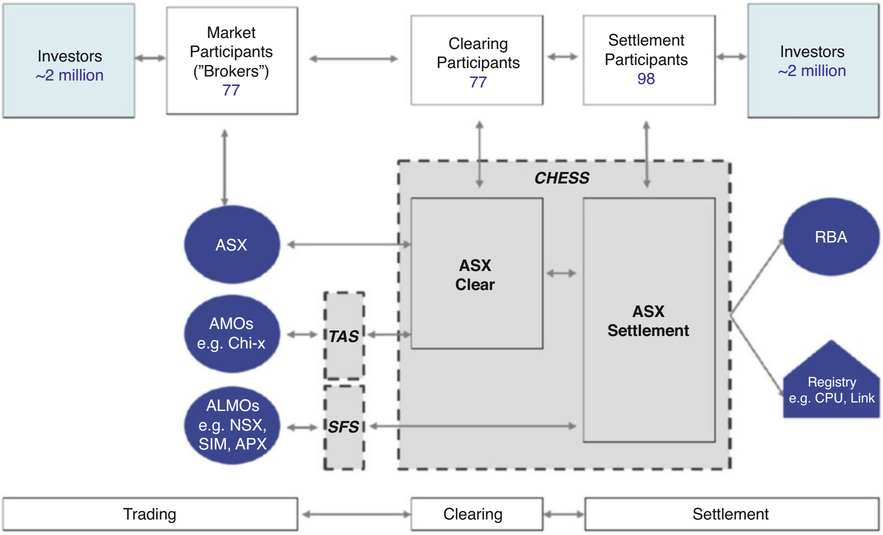


Fig. 5.4 Overview of the Australian Stock Exchange business network, processes, and systems (extended by the authors; based on AFME, 2015)

market intermediaries and end-clients. Over the years, developments have taken place to embrace the needs of new and existing consumers for faster and more detailed information or to open up new distribution arrangements via third-party distributors who cater to both retail and institutional users. Today, CHESS is the core system that performs clearing, settlement, and asset registration processes for the Australian capital market. The costs of CHESS rise, also because of the growing amount of data being generated and the specific demands for more detailed and lower latency information feeds. Therefore, ASX is planning for the replacement of the system and has requested bids to replace/enhance CHESS by blockchain technology.

5.5.2 *The Current Equity Trade Process*

The equity post-trade process starts after a security has been traded (sold/bought) at the exchange. The post-trade process starts with clearing of the trade. It is the process of ensuring that the terms of a contract, initiated by investors (1), are fulfilled by the settlement process. By using Securities Settlement Systems (SSS), custodian banks send their trade details to the Central Securities Depository (CSD) on behalf of their investors (2). After confirmation, the trade proceeds to a Central Counterparty (CCP), which acts as a buyer to the seller and vice versa (3) and (4) and executes the transfer of money and securities. Also, the CCP "nets" the transactions across all trades between a certain time period (i.e., 24 hours) as to minimize the number of required transactions. This way, counterparty risk is transferred to the CCP from the actual parties involved in the trade. After the transfer, securities and cash are stored in safekeeping accounts managed by the custodian and the CSD (5). As various servicing processes are offered, third parties are able to work directly with the CSD to ensure that custodians and, ultimately, investors are engaged (World Economic Forum, 2016; AFME, 2015). The current financial services architectures are pretty mature in terms of functioning and security. However, processes remain very complex, with over ten ledgers necessary to finalize trades.

According to World Economic Forum (2016), significant problems exist in the current process:

- Duration of the settlement: Investors are able to see traded securities in their account almost directly after receiving confirmation. However, settlement can take up to 3 days after the execution (t+3). As a result, actions that investors can take between the execution and settlement are limited.
- Inconsistent data: As a result of frequent trades, counterparty details change often; CSDs manually validate a number of transactions prior to settlement; this leaves room for error.
- Counterparty (financial) risk: Custodians must account for the possibility that a counterparty is unable to settle when due. This risk is covered by paying membership fees to CCPs over every trade. This way, CCPs are funded to cover losses in case of defaults.
- Costly intermediaries: Investors involve third parties as intermediaries to provide asset servicing.

In January 2016, ASX selected Digital Asset Holdings LLC as a technology partner to develop a working prototype of an equity post-trade platform using blockchain technology. This initial phase of work was completed in mid-2016. The prototype concluded that blockchain technology:

- Has successfully met initial capacity tests for ASX's security, scalability, and performance requirements for a replacement system when deployed in a permissioned private form.
- Offers potential benefits for investors, regulators, and intermediaries through improvements and innovations in operational processes. This will be explained in more detail later.

ASX has commenced a next phase of work to determine the specific business requirements for the blockchain-based system that will replace CHESS and to further develop the prototype to an industrial-scale post-trade platform using blockchain technology. ASX expects to make a final decision toward the end of 2017, in consultation with stakeholders, on implementing a blockchain-based replacement.

5.5.3 Changes to the Business Network Processes

Blockchain offers opportunities to change the equity post-trade processes:

- *Clearing Process*—This process ensures that all prerequisites for settlement are in place. This process validates the existence of sufficient securities and funds and stores trade details required for the settlement (e.g., the securities settlement date, identification codes, and settlement venue). Today, a significant part of this process is still done manually (World Economic Forum, 2016). Blockchain will help to further automate the process by using smart contracts. Using the new system, a trade executed at the exchange enters the post-trade process through an Authorized Market Operator (AMO) (i.e., custodian) with certain specifications and is directly linked to a smart contract which performs the necessary checks. The information needed to perform the checks is provided by the internal information systems which are automatically being kept up to date through blockchain (push) technology. When the transaction is cleared successfully, it passes to the settlement phase.
- *Settlement Process*—Settlement is the actual transfer of ownership of securities from a seller to a buyer and payment by the buyer to the seller. This process is usually conducted by a Central Counterparty (CCP). Using the new blockchain technology-based system, this process can be automated. Netting of trades is an important design topic of the new system. Netting is the process of collecting initiated trades as to determine the net amount of securities and cash to be transferred among AMOs after a particular timeframe (i.e., daily). Using blockchain technology, transactions could theoretically be settled directly after the execution of the trade (T+0). In case of the ASX, this would require the network to settle 884 K transactions per day. Allowing for a netting period of 2 days (T+2) would result in a 98.5% reduction to only 13 k transactions per day (Brennan & Lunn, 2016). To reduce the amount of traffic on the network, ASX has decided to set T+2 as the default settlement period. However, the new system will also offer shorter settlement periods by offering AMOs the ability to settle trades daily (T+1). This way, AMOs are provided with the ability to further reduce market risks (counterparty risks).
- *Asset Servicing Process*—ASX also provides registry services. Today, the registry processes contain manual activities and therefore are prone to error. The new system will use blockchain technology to automate this process using smart contracts to list the transactions in the ledger which is distributed among AMOs. This results in an automated single source of truth agreed on by all participants in the network. This benefits AMOs, who no longer have to keep track of their own version of the register.

It is evident that, in this case, blockchain technology can automate a significant part of the equity post-trade processes. This also means that human resources will shift activities from executing the processes to facilitating and monitoring the processes.

5.5.4 Changes to the Physical Layer (Business Network Structure)

The composition of the network in terms of participants will change as a result of the blockchain system. The new system uses ISO 20022 standards to enable global interoperability. It is expected that non-Australian organizations may aim to enter the network, resulting in a larger network and (see Sect. 5.5.3) more daily transactions. Since the ISO 20022 standards are not exclusively linked to blockchain technology, it cannot be concluded that blockchain technology causes this effect directly. Because blockchain enables implementing ISO 20022, we conclude that blockchain enables this effect indirectly. Changes to the network structure resulting from disintermediation are expected to be minimal, for most of the equity post-trade processes are performed by ASX.

Changes may occur also to the relations among the firms in the network. First of all, the density of the network will be increased significantly as AMOs will participate in a distributed network as a consequence of using blockchain technology. This is in contrast to the current structure, which is highly centralized around ASX. The new structure can be seen as a single clique in which every node is directly connected to other nodes, instead of indirect linkages via ASX. As a result, fragmentation and structural holes are reduced. This results in improved connectivity among the AMOs.

5.5.5 Changes in the Shared Business Operating Logic

This chapter addresses the implications of ASX's blockchain solution to the business operating logic of the network. As described in Sect. 5.2, we use the model of Birch et al. (2016) to structure the results.

- *Communication logic*—The blockchain-based solution will be a private permissioned form in order to enhance network security by allowing only selected entities to read and initiate transactions. Increased network security results from blockchain encryption of messages. By using this form of encryption, post-trade processes remain invisible to the public. Investors and issuers will be able to monitor and control their assets only through an Authorized Market Operator (AMO), or via FS regulators (i.e., ASIC, ARPA, and RBA) and the ASX. These selected firms will have access to personal and commercial data on connected service providers, investors, and market activity. These data are needed to manage risks of financial system integrity and stability. Also, the control function of the network remains at the ASX.

  Firms need to comply with several requirements in order to be licensed to participate in the network. Key requirements are "compliance to regulatory standards" and "adoption of predefined operating logic." Another key requirement is complying to the ISO 20022 standard. ISO 20022 is a globally interoperable messaging and data standard for the financial services industry. The key advantages of ISO 20022 include global interoperability and the use of uniform, reusable messages. Complying to ISO 20022 also means that the new blockchain-based system will enable global integration.

- *The content logic*—The content that will be transacted over the network will remain in the form of standardized messages. Messages in the current system are primarily "notified trade" and "netted trade" messages. Securities only exist digitally and will remain in the ASX register. This means that there are no significant changes expected to the content logic.
- *The consensus logic*—Because of blockchain technology, consensus will be reached when the nodes in the network agree on a transaction. Every Authorized Market Operator (AMO) (i.e., custodian) functions as a miner in order to be able to initiate transactions and to stay up to date. This is in contrast to CHESS, which functions as a central processing node. As a result, data is to be pulled from this central database by individual AMOs and (manually) converted to their information systems.
- *The contract logic*—The use of standardized messages within the network makes an excellent case for the use of smart contracts. Smart contracts can be used to automate processes within the network. One of the most important improvements to the contract logic is the ability to place holding locks over securities to ensure that they cannot be transferred unilaterally. This capability can also be used to configure conditions in the smart contracts that guarantee that particular securities are held for a specific purpose. For example, securities are only to be transferred to a counterparty which has a certain credit rating. Smart contracts could also be used to govern dividend, interest remittance, rights issues, and other services. However, today it remains unknown which of these functions will be embedded in the new system.

5.5.6 Improving Equity Post-trade Network Performance

Finally, we use the ASX case to assess how blockchain technology may ultimately affect network performance. Network performance is typically measured in terms of productivity, timeliness of information, operating costs, flexibility, and reduced workflow (Straub et al., 2004).

5.5.6.1 Enhancing Network Performance by Changing Business Operating Logic

Communication logic is configured in a private permissioned blockchain, meaning that only licensed parties can read and write on the network. As a result, the security of the network will increase, but there is no evidence found that this leads to an increase in network performance in terms of productivity, timeliness of information, operating costs, flexibility, and reduced workflow. *Content logic* will also not lead to increased network performance, since the logic will not be changed by implementing blockchain technology. Increased network performance is expected to result from the *consensus logic* and *contract logic*.

Consensus logic is changed since participants in the network will validate the initiated transactions via blockchain logic and since participants are kept up to date automatically. In this way, consensus logic results in increased timeliness of information and lower operating costs since information no longer needs to be pulled manually from ASX. Many parties within and outside the blockchain network will benefit from the increase in timeliness of information. Investors will benefit via the consolidated—real-time—view of holdings. Issuers will benefit via the increased transparency of shareholder base. Regulators will benefit via improved market transparency and clear order trails in blockchain ledgers.

The *contract logic* also enhances business network performance in the way that the use of smart contracts forces AMOs to use standardized formats. This reduces operating costs since data no longer needs to be converted to the standards of their own local information systems.

5.5.6.2 Enhancing Network Performance by Changing the Physical Layer

Performance of ASX's equity post-trade network is also enhanced by changes in the physical layer (network structure). By using blockchain technology, a distributed network structure is created in which information is shared with every participant. This increases the density of the network and spans structural holes. As a result, an ordered architecture of nodes emerges which leads to more information being shared. Moreover, information symmetry is also increased since every participant shares the same version of the ledger, which results in increased timeliness of information.

5.5.6.3 Enhancing Network Performance by Changing Network Processes

ASX estimates that implementation of the new blockchain-based system will lead to annual savings for AMOs of A$ 4–5 billion. These savings partly result from changes in network processes as discussed above. Smart contracts automate large parts of the clearing and settlement processes, resulting in lower operating costs and shorter workflows as less processes have to be executed manually. Moreover, smart contracts execute the processes much faster and thus reduce settlement times. As a result, counterparty risk is reduced which leads to a reduction in capital holdings to cover for potential financial losses

Summarizing the sections above, it can be concluded that the ASX case provides empirical support for the assumptions that blockchain technology enhances network performance via changes in the business operating logic, network structure, and network processes. Network performance is enhanced in terms of operating costs, timeliness of information, and workflow. No evidence has been found that indicates that blockchain technology enhances business flexibility.

5.6 Conclusion

The aim of this chapter was to explore how blockchain technology applications may enhance network performance and affect the structure of the financial industry. We focused on a financial services network and analyzed the securities transactions network in the Australian Stock Exchange case. We now answer our research question "how can blockchain technology enhance the performance of financial services networks?". Obviously, more research and cases are required, but the ASX case allows making a couple of observations.

Paraphrasing the well-known Carr debate around (see *Harvard Business Review* around 2003), we conclude that "Blockchain technology doesn't matter." Following the Carr debate, we also conclude that is not about the technology itself, but about how it is applied in business.

The technology itself is described by 12 concepts (the datalogical level). The technology essentially establishes irrevocable transactions, safe computations, externalized databases, externalized data processing, and externalized control. Additionally, blockchain technology can be combined with IoT technology where tight coupling of the real and virtual world occurs.

The implementation of the technology is described in 11 concepts: 7 concepts for "chain" (the infological level) and "digital ledger, account, transaction, and journal" (the business level) (Sect. 5.2).

Following business network theory, the implementation of blockchain technology affects business operations and the business network structure (the physical layer), the information layer, as well as communication logic, content logic, consensus logic, and contract logic (the business logic layer) (Sects. 5.3, 5.4, and 5.5).

Based on our ASX blockchain technology case, we now see the following key (simple, practical) questions that may help to predict the success of blockchain technology implementations:

1. Who starts the blockchain application (and seeds the first block)?
2. Which other firms participate in the blockchain, and is it closed (private blockchain) or open to other firms (public or hybrid blockchain)?
3. Who (in the network) decide(s) on the four types of logic to be applied in the blockchain?
4. Which transaction data are stored in the blockchain? More specifically, which data are shared, and which data remain proprietary-owned by the firms in the network?
5. How is the blockchain linked to the other (internal and interorganizational) information systems and applications in the business network?

These five questions and the business-technology perspective given in Sects. 5.2–5.5 provide a blockchain analysis tool to evaluate, design, and, ultimately, predict the impact of blockchain technology on business networking.

More research is needed on how blockchain technology implementations may affect:

1. Transactions
2. Pooled dependence versus sequential interdependence between business activities
3. Complete versus incomplete contracts
4. Self-governance vs. vertical governance

More research will help to refine and add to the five questions listed above. The questions may result in specifying success factors and performance indicators of blockchain technology implementations. Furthermore, we expect these questions, success factors, and performance indicators will help to predict success of applying blockchain technology in the financial—and other—industries.

Chapter 6
Analysis and Design of Digital Value Co-creation Networks: Insights from Digital Platforms

Michael Blaschke

Abstract While many *digital value co-creation networks* (DVCNs) fail to coevolve with the changes inside and outside the network, the few surviving ones have contributed to the emergence of the world's most valuable companies such as Apple, Alphabet, or Amazon. This chapter systematically reviews six research articles to investigate how organizations may analyze and design surviving DVCNs—here defined as socio-technical networks of actors to co-create digital service.

This chapter summarizes six studies, undertaken as part of the *Val*ue *Co*-creation *La*nguage (ValCoLa) project at the University of St. Gallen, to investigate how network orchestrators facilitate efficient and effective value co-creation processes—among themselves, third parties, and end users—striving for lasting networks. Module 1 reviews three articles to develop the conceptual foundation of DVCN analysis and design. In turn, module 2 reviews three additional studies that investigate digital platform cases to identify value co-creation capabilities for digital platform survival. The results indicate that DVCNs should not be misconceived as deterministic systems. Instead, they should be analyzed and designed in a way that reconciles a copresence of (i) heterogeneous third parties that foster emergent innovation and (ii) network orchestrators that control value co-creation.

6.1 Introduction

This chapter investigates how advanced communication and information technology (IT) transform the structure of value creation to dynamic value networks—inverting organizations inside out. This chapter rests on three fundamental assumptions. First, the conceptualization of economic exchange undergoes a groundbreaking reorientation from a traditional goods-dominant (G-D) to a service-dominant (S-D)

M. Blaschke (✉)
Kearney Digital, Zurich, Switzerland

Institute of Information Management, University of St. Gallen, St. Gallen, Switzerland
e-mail: michael.blaschke@kearney.com; michael.blaschke@unisg.ch

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_6

logic to promote value creation in networks of actors. These networks comprise, among others, network orchestrators, independent third parties, and end-customer organizations (Vargo & Lusch, 2016). Therefore, we employ the term *value co-creation* to refer to any process of value creation through reciprocal resource integration among actors. The work, as reported in this chapter, was also part of the ValCoLa project as mentioned in Sect. 1.3.

Second, IT plays an increasingly dominant role in such value co-creation processes in that it expands networks of actors beyond their temporal, organizational, and spatial boundaries (Lusch & Nambisan, 2015). To account for IT in value co-creation networks, this chapter investigates *digital value co-creation networks* (DVCNs)—socio-technical networks of actors to co-create digital service. Third, DVCNs are not purely technical in nature, but occur due to the complex, interorganizational interplay of human actors and IT artifacts (Breidbach & Maglio, 2016). Therefore, we employ the term *information system* (IS) to refer to any socio-technical system where human actors interact with IT artifacts to process information (Alter, 2008).

Overall, this chapter introduces value co-creation as a new perspective on how organizations may best analyze and design their networks, while acknowledging the limits of direct control in emergent networks (Ghazawneh & Henfridsson, 2013). The development of this perspective is primarily based on S-D logic due to its distinctive and penetrative conceptualization on how value is *co*-created in emerging, networked business environments (Lusch & Nambisan, 2015). From an S-D logic viewpoint, DVCNs should not be misconceived as deterministic and directly controllable systems. Rather, DVCNs should be analyzed and designed in a way that delicately reconciles a copresence of (i) heterogeneous third parties that leverage generative innovation and (ii) powerful network orchestrators that control the network to some degree.

6.1.1 Problem Setting

A pivotal question for any organization to ask is how to create value: Should it produce its own output, or should it orchestrate the output of third parties (Parker et al., 2017)? Contemporary software firms, such as Apple and Alphabet, increasingly favor orchestration. They build up and orchestrate interconnected IS collectives with third parties to leverage generative innovation. For example, an analysis of one million mobile applications shows that any given mobile application includes as median 10 third parties, and about 18% of mobile applications rely on even 20 third parties (Binns et al., 2018). Therefore, we argue that organizations must manage value creation that occurs *externally* just as carefully as they manage the value they create *internally* (Parker et al., 2017). Interestingly, even firms that produce physical products add a digital layer for third parties to augment the physical product (vom Brocke et al., 2017).

This production-to-orchestration shift is driven by improvements in network connectivity and computing power (Loebbecke & Picot, 2015). Indeed, earlier improvements in transportation technology changed the locus of value creation from vertically integrated organizations to ones organized around a nexus of supplier networks. The current shift toward network orchestration (Henfridsson & Bygstad, 2013), however, goes farther still. Organizations harness a global network of actors they have never even met. These third parties can connect through digital technology to innovate on top of an orchestrator's core set of resources, thereby creating highly valuable products and services for end users (Lyytinen et al., 2016). To account for such digital and multi-actor value co-creation processes, we promote DVCNs. While research has started investigating these networks (Barile et al., 2016), two lasting problems hamper an effective analysis and design of DVCNs.

6.1.1.1 Problem 1: Conceptual Foundation of DVCN Analysis and Design

Accounting for the orchestration logic, we deem value co-creation a timely, opportune, and novel perspective on ISAD to inform DVCN analysis and design (Haki et al., 2019). However, IS research has insufficiently organized and introduced the value co-creation concept as a perspective on ISAD. McColl-Kennedy et al. (2012) have provided a catalogue of 27 *different* definitions of value co-creation emphasizing an equivocal understanding and conceptual ambiguity. Unambiguous definitions of value co-creation, its constituent concepts, and their inherent relationships are not available.

Moreover, existing value co-creation conceptions are limited in their phenomenological coverage of DVCNs. That is, first, DVCNs' constituent actors co-create value through digital rather than physical interfaces (Bitner et al., 2000), an IS research priority (Lusch & Nambisan, 2015). While value co-creation research traditionally sheds light on value co-creation at physical touch-points (Bitner et al., 2000; Breidbach & Maglio, 2016), IS nowadays enable actors to synergistically integrate resources through digital interfaces (Davis et al., 2011; Giebelhausen et al., 2013). For instance, Lusch and Nambisan (2015) call for a reflection of IS as both facilitator and initiator of value co-creation processes. Second, DVCNs' constituent actors integrate *distributed* rather than *co-located* resources (Breidbach & Maglio, 2016). Third, DVCNs' constituent actors co-create value in multilateral rather than bilateral relationships (Akaka et al., 2012; Akaka & Vargo, 2014). IS research insufficiently captures these three characteristics in its phenomenological coverage of DVCNs. This limited coverage can hamper an effective DVCN analysis which, in turn, is likely to translate into ineffective DVCN design.

Finally, value co-creation has been conceptually discussed for over a decade (Galvagno & Dalli, 2014; Ranjan & Read, 2014). However, detailed empirical account and analysis of value co-creation processes in general, and those occurring in DVCNs, are limited. We consequently know little how network orchestrators and their third-party communities *effectually* interact and synergistically integrate resources with one another and with end user organizations.

This limitation is specifically true for the role of IS in value co-creation. IS advances have substantially transformed value co-creation, and these technology-driven advances also need to be taken into consideration when attempting to explore value co-creation (e.g., Sarker et al., 2012). In effect, an investigation of the mechanisms and processes of how DVCNs' constituent actors integrate resources with end user organizations via IS-enabled value co-creation is required. These three outlined limitations, we argue, hamper an efficient and effective analysis and design of DVCNs.

Problem 1 *Limited ontological clarity, phenomenological coverage, and empirical account of value co-creation hamper an efficient and effective analysis and design of DVCNs.*

6.1.1.2 Problem 2: Value Co-creation Capabilities for DVCN Survival

While few DVCNs thrive, such as those of Apple (iOS), Alphabet (Android), and Microsoft (Windows), many others die in the long run (Loukis et al., 2016). A prominent example of DVCN failure is Microsoft's operating system Windows Phone whose support was discontinued in 2017 with a 0.2% market share (Gartner Group, 2017).

We lack a solid understanding of why few DVCNs survive, while most others die (Reuver et al., 2018, p. 7). Therefore, we spotlight *DVCN survival*—defined as a DVCN's state of continued existence (Josefy et al., 2017). Seeking to promote DVCN survival, we particularly focus on value co-creation capabilities that a network orchestrator should possess in relation to third-party communities and end user organizations. *Co-creation capability* refers to an ability to repeatedly perform or achieve certain actions or outcomes that relate to a firm's capacity for co-creating value (Grant, 1999). For instance, a hospital's capability in cardiovascular surgery is dependent on integrating the specialist knowledge of surgeons, anesthetists, radiologist, operating-room nurses, and several types of technicians. Similarly, we argue that a DVCN's capability in surviving is dependent on its orchestrator's ability to efficiently and effectively integrate multiple actors' distributed resources.

Problem 2 *Limited understanding of value co-creation capabilities for DVCN survival contributes to the discontinuation of many DVCNs.*

Table 6.1 Overview of modules

Module	Summary
Module 1. Conceptual Foundation of DVCN Analysis and Design *Studies A, B, C*	▪ **Focus**: Digital Value Co-creation Networks (DVCNs) ▪ **Sub-Research Question 1**: How does a value co-creation perspective on ISAD inform the analysis and design of DVCNs? ▪ **Conceptual Foundation**: Service-dominant (S-D) logic as a theory about value creation in networks of actors ▪ **Overview**: Module 1 ontologically organizes the *value co-creation* concept (Study A) as a complementary perspective on *ISAD* (Study B) to inform *DVCN analysis and design* (Study C). ▪ **Relation to Module 2**: Module 1 conceptually informs Module 2
Module 2. Empirical Analysis of Digital Platform Survival *Studies D, E, F*	▪ **Focus**: Digital Platforms ▪ **Sub-Research Question 2**: How do digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival? ▪ **Conceptual Foundation**: Value co-creation capabilities (independent variable) to effect digital platform survival (dependent variable) ▪ **Overview**: Module 2 empirically examines *digital platform dimensions* (Study D) and organizes these dimensions' characteristics in a *digital platform taxonomy* (Study E) to propose *value co-creation capabilities* for digital platform survival (Study F). ▪ **Relation to Module 1**: Module 2 empirically exemplifies Module 1

6.1.2 *Research Objective and Research Questions*

Motivated by the problem setting as outlined above, the research objective of this chapter is to propose value co-creation capabilities for DVCN survival. The development of these capabilities is split into two interrelated modules (see Table 6.1).

6.1.2.1 Research Question: Value Co-creation Capabilities for DVCN Survival

Network orchestrators are viewed as a central point of gravity within their DVCN. For instance, the firms Alphabet and Apple have become the cornerstones of their mobile communication networks around the mobile operating systems Android and iOS. Relying on digital technologies, orchestrators facilitate the integration of resources among multiple, varied, and interdependent actors. These actors and their relations evolve over time in varied patterns and rates of change (Tiwana et al., 2010).

Therefore, emergent value creation in DVCNs is contingent on a network orchestrator's capability in forming a critical mass of diminutive resource sets (Grover & Kohli, 2012; Tan et al., 2015). In case of malfunction, these specificities can substantially impede DVCN survival (Reuver et al., 2018). Therefore, we argue that DVCN survival is contingent on how different actors with various roles dynamically evolve to *jointly* create value for specific end user needs (Sarker et al., 2012). Thus, DVCN survival is contingent on its orchestrator's organizational capabilities to ensure effective value co-creation (Friend & Malshe, 2016; Grover & Kohli, 2012). This chapter, therefore, focuses on the following:

Main Research Question (RQ):
What value co-creation capabilities promote DVCN survival?

The main RQ is broken down into two sub-research questions (SRQs). Therefore, this chapter is split into two modules, each of which investigates one of the two SRQs, respectively (see Table 6.1). Based on the findings of these two modules, this chapter answers the main research question (problem 2).

Notably, module 2 focuses on the specific empirical context of digital platforms. The subtitle of this chapter, i.e., *Insights from Digital Platforms*, reflects this research design. *Digital platform* refers to a set of digital core technologies augmented by peripheral third-party derivatives and associated organizational arrangements (Reuver et al., 2018). We choose digital platforms as DVCN exemplars as they (i) facilitate the integration of resources in networks of actors (Sarker et al., 2012); (ii) become increasingly valuable when more third parties add their derivatives (Parker et al., 2017); and (iii) mediate the operations of the most valuable companies (FXSSI, 2019).

Different exemplars of DVCNs are in-store self-service technology, e-commerce, online support, email, or IT-mediated projects (e.g., consulting projects) (Breidbach & Maglio, 2016). The discussion presented in Sect. 6.6 generalizes module 2's findings from the specific digital platform context to the chapter's general DVCN phenomenon.

6.1.2.2 SRQ1: Conceptual Foundation of DVCN Analysis and Design

First, an ontologically sound conception of value co-creation is required as conceptual foundation for the analysis and design of DVCNs (SRQ1). This foundation is required as IS research has not ontologically organized and introduced the value co-creation concept as a perspective on IS analysis and design, resulting in both limited phenomenological coverage and empirical account of DVCNs (problem 1).

SRQ1 aims to introduce the *value co-creation* concept as a timely, opportune, and novel perspective on *IS analysis and design* (ISAD) to inform *DVCN analysis and design*. This research question draws on and integrates in a rich body of ISAD literature (Iivari et al., 2000) according to which *IS analysis* aims to specify IS requirements based on human actors' common understanding about a real-world domain (Recker et al., 2011). In turn, *IS design* employs techniques to translate IS requirements into logical IS designs that fulfil the IS requirements (Gregor & Hevner, 2013).

This chapter introduces value co-creation as a perspective on ISAD to inform the analysis and design of DVCNs. ISAD ensures IS designs in line with a business environment's requirements. Since ISAD approaches follow the currently dominant business logic, the rise of a novel business logic requires revisiting and advancing extant ISAD approaches. S-D logic represents such a novel logic of business. Its core concept, value co-creation, emphasizes joint value creation among a variety of networked actors. Module 1 employs S-D logic and value co-creation to provide a novel perspective on ISAD. To do so, module 1 discusses ISAD's main research streams and the role of S-D logic and value co-creation in these streams.

Sub-research Question 1 (SRQ1):
How does a value co-creation perspective on ISAD inform the analysis and design of DVCNs?

6.1.2.3 SRQ2: Empirical Analysis of Value Co-creation Capabilities for Digital Platform Survival

While most researchers and practitioners have an intuitive understanding of value co-creation, this intuition is reflected in a plethora of highly diverse discussions of value co-creation in the current literature (Galvagno & Dalli, 2014; Ranjan & Read, 2014). Moreover, most studies focus on general ideas, whereas only a few explicitly offer empirical account for the general phenomenon of value co-creation (Sarker et al., 2012). Thus, an empirical analysis of value co-creation capabilities for digital platform survival (SRQ2) is required to mitigate the limited detailed empirical account and analysis of DVCN instances in IS research (problem 2).

SRQ2 promotes *digital platforms*, contemporary exemplars of DVCNs, for this empirical analysis. This research question aims to examine digital platforms' *dimensions* and organizes these dimensions' characteristics in a digital platform *taxonomy* to derive *value co-creation capabilities* for platform survival. Residing in the firm-external end of platform research (Ceccagnoli et al., 2012; Reuver et al., 2018; Parker et al., 2017; Sarker et al., 2012; Tiwana, 2015), the chapter rests on the premise that *digital platform survival* is contingent on the availability and contribution of a critical mass of actors within each of the relevant actor roles (Grover & Kohli, 2012; Parker et al., 2017; Tiwana, 2015). Each of these actor roles offers complementary resources to the respective network to serve a wide range of end users and to satisfy various requirements (Wareham et al., 2014).

Sub-research Question 2 (SRQ2):
How do digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival?

Concisely stated, the purpose of this chapter is (i) to understand what value co-creation capabilities promote DVCN survival (RQ), (ii) to understand how a value co-creation perspective on ISAD informs DVCN analysis and design (SRQ1), and (iii) to explain how digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival (SRQ2).

6.1.3 Overview of Research Design

Below, we briefly describe the overall research design of this chapter. While this section outlines the relation between the two modules and summarizes their respective methodology, data, analysis, and key results (see Fig. 6.1), Sect. 6.3 provides an in-depth account on the two module's research design.

6.1.3.1 Module 1: Conceptual Foundation of DVCN Analysis and Design

The outlined diversity in value co-creation research may be ascribed to a concurrent development of distinct value co-creation discourses, such as service science and S-D logic. These not only have different understandings of what value co-creation is

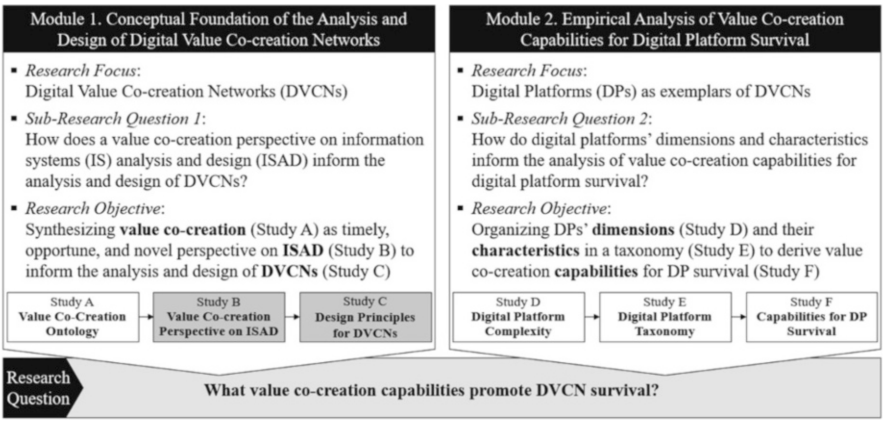


Fig. 6.1 Overview of research design

but also come with their own and incompatible theoretical assumptions. Following calls for research in IS (Alter, 2008) and marketing research (O'Shaughnessy & O'Shaughnessy, 2009), module 1 (i) develops an unambiguous and multidisciplinary *value co-creation glossary* of its constituent concepts and (ii) integrates these concepts and their relations into a value co-creation ontology for ISAD. Moreover, while value co-creation studies to date focus on theoretical, philosophical, and conceptual underpinnings of value co-creation (Ranjan & Read, 2014), very few studies embrace value co-creation as a vantage point to study various IS phenomena, such as DVCNs (Ceccagnoli et al., 2012; Grover & Kohli, 2012; Sarker et al., 2012). Module 1 is therefore concerned with synthesizing value co-creation (Study A) as a timely, opportune, and novel perspective ISAD (Study B) to inform the analysis and design of DVCNs (Study C). To integrate different value co-creation conceptions, module 2 reviews Study A, which combines an assortment of complementary views into a single ontology.[1]

In reviewing three studies, module 1 promotes an overarching value co-creation ontology as a necessary conceptual foundation for subsequent applications in ISAD and in DVCN analysis and design. Second, it motivates the chapter's overall proposition: ISAD should account for the transformative role of IT in inverting organizations from inside-out to nonlinear value networks. Module 1's conclusion is that DVCN analysis and design is constrained by a mere *theoretical* reflection of S-D logic and value co-creation in ISAD. This motivates module 2, which analyzes how *empirically* investigating digital platforms might overcome these limitations.

6.1.3.2 Module 2: Empirical Analysis of Value Co-creation Capabilities for Digital Platform Survival

With the aim of empirically analyzing value co-creation capabilities for DVCN survival, module 2 argues that digital platforms represent contemporary and common DVCN exemplars that are ideally suited to empirically analyze such capabilities. This has two related sub-aspects. First, although digital platforms widely differ in their configurations, digital platforms' dimensions and characteristics to disentangle different digital platform configurations are under-researched. To bridge this gap, this module employs a dedicated method for taxonomy development in IS to systematically derive a taxonomy of digital platforms (Nickerson et al., 2013). Specifically, the adopted platform architecture perspective captures the configuration of digital platform's components (Cennamo et al., 2018). Regarding data, module 2 reviews 46 digital platform studies to not only derive dimensions and characteristics but also to scrutinize digital platform instances studied in research to inform our taxonomy.

[1] Ontologies are used to structure and codify knowledge about the concepts, relationships, and axioms/constraints pertaining to a domain (e.g., value co-creation) (Kishore et al., 2004b). Ontologies for ISAD are prominently discussed in IS research as ontology-driven IS (Fonseca & Martin, 2007; Guarino, 1998) independent of the domain of interest (Fernández-López et al., 1997, p. 36).

The resultant taxonomy facilitates a more pronounced understanding and grouping of digital platforms as configurations of certain dimensions and characteristics.

Second, module 2 opts for a single-case study approach owing to the inherent and multifaceted complexity of digital platforms rooted in a multitude of actors (static complexity) that coevolve in varied patterns and rates of change (dynamic complexity). We choose a single-case study as capabilities for digital platform survival are poorly understood and as digital platform literature is heterogeneous and young (Thomas et al., 2014). Regarding data, this module analyzes an established business-to-business (B2B) digital platform for enterprise software that has grown globally since its launch in 2012. This context is particularly relevant for this chapter, as the platform owner has experimented with several *other* digital platforms before 2012. Thus, experience with several previous attempts to implement digital platforms is included in this platform owner's value co-creation capabilities.

6.2 Research Foundations

This section introduces the research foundations of this chapter. It reviews the current state of research in related areas. Module 1 draws on, integrates in, and extends three discourses—DVCNs, ISAD, and value co-creation—that constitute the research foundation for module 1. In turn, module 2 draws on, integrates in, and extends three different discourses—value co-creation capabilities, DVCN survival, and digital platforms—that constitute the research foundation for module 2. Table 6.2 synthesizes these six discourses that underlie the two modules.

Notably, in briefly sketching these discourses to examine their underlying logic, the employed citations are merely illustrative; a structured review of each discourse would be a substantial and worthwhile research project.

Table 6.2 Overview of research foundations

Module	Discourse	Definition
Module 1. Analysis and Design of DVCNs	DVCN	Socio-technical network of actors to co-create digital service
	ISAD	The process of systematically identifying requirements of a real-world domain to reflect these requirements in IS development
	Value Co-creation	The process of resource integration incorporating different actor roles in a network of actors
Module 2. Value Co-creation Capabilities for Platform Survival	Co-creation Capability	An organization's ability to repeatedly perform or achieve actions or outcomes that relate to an organization's capacity for co-creating value
	Platform Survival	A digital platform's state of continued existence
	Digital Platform	A set of digital core technologies augmented by peripheral derivatives, and associated organizational arrangements

6.2.1 Module 1: Analysis and Design of DVCNs

Module 1 rests on the *DVCN*, *ISAD*, and *value co-creation* discourses in investigating conceptual foundations of the analysis and design of DVCNs.

6.2.1.1 Phenomenon of Interest: Digital Value Co-creation Networks

DVCN research moves from the traditional view of value co-creation that emphasizes dyadic one-to-one encounters to a more encompassing view of multi-actor constellations within which actors are connected through digital technology over time and space (Akaka et al., 2012; Beirão et al., 2017; Barile et al., 2016). S-D logic refers to this as *service ecosystems* (Vargo & Lusch, 2011).

Service ecosystems are characterized as "spontaneously sensing and responding spatial and temporal structures of largely loosely coupled, value-proposing social and economic actors interacting through institutions, technology, and language to (1) co-produce service offerings, (2) engage in mutual service provision, and (3) co-create value"(Vargo & Lusch, 2011, p. 185).

In such service ecosystems and through digital technology, IS enable distributed actors to not only exchange but synergistically integrate their resources.

To account for such digital and multi-actor value co-creation processes, this chapter promotes DVCNs as investigated IS phenomenon.

DVCNs can be characterized as a "heterogeneous and dynamic pool of actors and tools that need to be dynamically identified and mobilized for effective cognitive and social translations across a diverse set of actors in the absence of hierarchical control and presence of high levels of knowledge heterogeneity" (Lyytinen et al., 2016, p. 59).

Against this backdrop, we designate *DVCN* as socio-technical network of actors to co-create digital service. DVCNs label a specific class of service ecosystems with three characteristics. They (i) are inextricably intertwined with and mediated by digital technology, (ii) configure digital service in *value co-creation* processes, and (iii) are constituted of multi-actor *networks*. These networks include orchestrators, third parties, and subcontractors all of which (re)form to meet an end user organization's requirements.

6.2.1.2 Topic Focus: Information Systems Analysis and Design

This chapter focuses on the analysis and design of DVCNs. Thereby, it engages in IS analysis and design, one of the most classical fields of IS research lying in the core of IS research (Iivari et al., 2006; Siau & Rossi, 2011; Sidorova et al., 2008; Wand & Weber, 1993). ISAD systematically identifies requirements of a real-world domain to reflect these requirements in IS development (Recker et al., 2011; Wang & Wang, 2012).

IS analysis aims to gather, analyze, specify, and document IS requirements based on a common understanding that stakeholders have about a real-world domain (Recker et al., 2011). It denotes "a number of activities in the early stages of IS development [. . .] to identify and document the requirements for an IS to support organizational activities" (Iivari et al., 2006, p. 510). In turn, IS design translates these requirements into logical IS designs to fulfil the requirements imposed by the real-world domain (Gregor & Hevner, 2013). It denotes "the process of defining the system architecture, components, modules, interfaces, and data for a software system to satisfy the requirements specified during systems analysis" (Iivari et al., 2006, p. 510).

At least four constituent ISAD research streams can be distinguished (Iivari et al., 2006, p. 510). The first stream, *ISAD activities and processes*, comprises all analysis and design activities and processes included in IS development endeavors. Exemplary ISAD activities are problem identification, requirement engineering, conceptual modeling, design specification, construction, and evaluation (Sonnenberg & vom Brocke, 2012). This stream is supported by the second stream on *ISAD instruments* that is concerned with developing artifacts (e.g., approaches, methods, techniques, and tools). The third stream, *research methods used for ISAD*, sheds light on research methods (e.g., experiment) for constructing ISAD artifacts. The fourth stream, *ISAD theory*, is concerned with the underlying theoretical basis of ISAD artifacts.

6.2.1.3 Conceptual Foundation: Value Co-creation in Service-Dominant Logic

This chapter draws on S-D logic as a theoretical lens to conceptualize both value co-creation and emerging networked business environments in which DVCNs are devised. Two reasons justify this choice. First, S-D logic sophisticates the value co-creation concept through establishing a holistic, unified, and precise theoretical foundation as a distinctive, yet complementary, filter on extant debates (Vargo and Lusch 2017; Vargo et al. 2010). This well-defined theoretical basis informs and guides the DVCN analysis and design. Second, promoting value creation in networks of actors, S-D logic offers a penetrative conceptualization of the emerging business environments in which DVCNs are devised (Lusch and Nambisan, 2015).

Two reasons justify this choice. First, S-D logic sophisticates the value co-creation concept through establishing a holistic, unified, and precise theoretical foundation as a distinctive, yet complementary, filter on extant debates (Vargo & Lusch, 2017; Vargo et al., 2010). This well-defined theoretical basis informs and guides the DVCN analysis and design. Second, promoting value creation in networks of actors, S-D logic offers a penetrative conceptualization of the emerging business environments in which DVCNs are devised (Lusch & Nambisan, 2015).

S-D logic is rooted in marketing research since its inception by (Vargo & Lusch, 2004), followed by further amendments (Vargo & Lusch, 2008, 2016) that specify *value co-creation*—the process of resource integration incorporating different actor roles in a service ecosystem. Having emerged as S-D logic's core concept (Edvardsson et al., 2011; Galvagno & Dalli, 2014; Payne et al., 2008; Ranjan & Read, 2014), value co-creation is pivotal for S-D logic as it integrates all related constructs, namely, actor, resource, service, institutional arrangement, and service ecosystem.

In also explicating their relationships, value co-creation underscores that all actors in social and economic exchange integrate resources and engage in service exchange, all in the process of synergistically and reciprocally co-creating value (Vargo & Lusch, 2016, p. 3). Value co-creation is represented by the reciprocity of exchange, as well as by the existence of shared institutional arrangements that facilitate this exchange in the given service ecosystem (Vargo & Lusch, 2016). Actors integrate resources through service exchange that is configured by institutional arrangements through which service ecosystems endogenously emerge.

6.2.1.4 State of Research in the Field

IS research underscores the significance of analyzing and designing DVCNs (Reuver et al., 2018; Lusch & Nambisan, 2015; Tilson et al., 2010). Only little research strives for a distinctive conceptual foundation for DVCN analysis and design. Ontologies are the first step in devising conceptual foundations of a research domain (Kishore et al., 2004a). Four constituent streams of research on ontology in IS can be differentiated (Fonseca & Martin, 2007; Kishore et al., 2004a; Sharman et al., 2004). The first stream, *ontology theory in IS*, is concerned with the theoretical basis of ontologies in IS. The second stream, *ontology research methods in IS*, sheds light on IS-specific research methods for constructing ontologies. The third stream, *ontology for IS*, is concerned with constructing IS-related ontologies to reflect them in ISAD. This chapter draws on and integrates in the third stream. Lastly, the fourth stream, *ontology of IS*, employs ontologies as a lens to evaluate theory in IS.

Previous research sheds insufficient light on a value co-creation ontology for ISAD. There are few studies aiming at one or a few fragmented aspects of value co-creation in ontology development, such as an ontological foundation of S-D logic (Fragidis & Tarabanis, 2011), core value ontology (Gailly et al., 2016), service system ontologies (Lemey & Poels, 2011), and service science ontology (Lusch et al., 2008). Nevertheless, hitherto, no such work on a value co-creation ontology from an S-D logic perspective is available. Existing ontologies that relate to S-D logic or value co-creation either (i) engage in ontology development for value co-creation, but not from an S-D logic perspective, or (ii) engage in an S-D logic perspective but do not focus on value co-creation.

Since ontologies are a direct foundation of ISAD (Verdonck et al., 2015), several ISAD techniques reflect and use certain aspects of value (co-)creation, for instance, *e3Value* (Gordijn & Akkermans, 2003) for value modeling, *i** (Yu & Mylopoulos, 1994) and *KAOS* (Matulevicius et al., 2008a) for goal modeling, as well as *ArchiMate* (Lankhorst et al., 2017) and CIMOSA (Kosanke, 1995) for enterprise architecture modeling.

Nevertheless, these ISAD techniques do not systematically account for the concept of value co-creation—particularly not as conceptualized in S-D logic. Consequently, no ontology is available that can be used as a foundation for a value co-creation coverage in ISAD.

6.2.2 Module 2: Value Co-creation Capabilities for DVCN Survival

Module 2 rests on the *DVCN survival*, *value co-creation capabilities*, and *digital platform* discourses in investigating value co-creation capabilities for digital platform survival.

6.2.2.1 Phenomenon of Interest: DVCN Survival

Understanding why some organizations survive while others fail is a central question of strategic management research (Josefy et al., 2017). Indeed, many consider survival the *quintessential* indicator of organizational performance. This chapter builds upon the management literature's construct *organizational survival* (Josefy et al., 2017) to spotlight the notion of DVCN survival. *DVCN survival* here denotes a DVCN's state of continued existence. In turn, *DVCN failure* denotes a DVCN's dissolution. DVCN survival can be measured by the *level of continuity* of its (i) operations, (ii) ownership, and (iii) solvency (building on Josefy et al., 2017, p. 773). On one end of the spectrum are continuation of the DVCN's operations or market persistence, continuity of ownership, and continued solvency. On the other end are discontinuation of the DVCN's operations (including dissolution or market exit), discontinuity of ownership (including sale or acquisition), and insolvency or bankruptcy. Operations capture the continuation or cessation of DVCN activity in a given market environment. Continuity of ownership addresses mergers and acquisitions. Solvency reflects an aspect of a DVCN's financial performance. In contrast to the other two dimensions, a DVCN's continued solvency may be more consistently categorized as DVCN survival, whereas bankruptcy can be clearly demarcated as DVCN failure.

6.2.2.2 Topic Focus: Value Co-creation Capabilities

Our investigation of DVCN survival particularly focuses on pivotal capabilities that network orchestrators should possess to effect DVCN survival. In this chapter, *value co-creation capability* refers to the ability to repeatedly perform or achieve certain actions or outcomes that relate either directly or indirectly to a firm's capacity for co-creating value (Grant, 1999). More specifically, we focus on value co-creation capabilities—i.e., capabilities in relation to third-party communities and end user organizations. This focus rests on the premise that network orchestrators must attract an entire ecosystem of third parties that are capable of serving as development, sales force, or consulting partners to configure digital service for end users (e.g., Ceccagnoli et al., 2012; Sarker et al., 2012). To account for these multiple, varied, and interdependent actors co-creating value in ecosystems (Ceccagnoli et al., 2012; Han et al., 2012; Sarker et al., 2012), we specifically focus on value co-creation capabilities.

6.2.2.3 Empirical Context: Digital Platforms

A recent and rapidly growing discourse in IS research seeks to understand the omnipresent digital platforms in today's industries (e.g., Reuver et al., 2018; Kazan et al., 2018; Parker et al., 2017). Prime examples of digital platforms are social media (e.g., Facebook and LinkedIn), operating system (e.g., Android and iOS), payment (e.g., PayPal and Apple Pay), and peer-to-peer (e.g., Uber and Airbnb) platforms. Prevalent IS research views these digital platforms as a set of digital core technologies augmented by peripheral platform derivatives and associated organizational arrangements (Reuver et al., 2018). Following this definition, thriving digital platforms are contingent on attracting third parties to add their platform-augmenting derivatives (Parker et al., 2017). Embracing this logic, prominent digital platform owners such as Apple (iOS), Alphabet (Android), and Microsoft (Windows) are among the most valuable companies (Statista, 2017).

Digital platforms serve as central point of gravity for arising digital ecosystems (Parker et al., 2017). The concept of platform dates back to the late 1990s (Ciborra, 1996; Kim & Kogut, 1996), where research started to focus on technology as a mediating factor on innovation activities of *two* cooperating actors (Rochet & Tirole, 2003). In the mid-2000s, research started to investigate platforms as a mediator for networked, *multilateral* innovation activities (Gupta et al., 2007). Due to the pervasiveness of digital technology in supporting collaboration across spatial and functional borders, IS research began to study platforms not only as an integrator for different actors but also as a central form of organizing technological innovation (Yoo et al., 2012). Bringing together a network of actors (Reuver et al., 2018), digital platforms can also be characterized as a "building block, providing an essential function to a technological system—which acts as a foundation upon which other firms can develop complementary products, technologies or services" (Gawer, 2011, p. 2).

6.2.2.4 State of Research in the Field

IS research amplifies the significance of DVCN survival (Constantinides et al., 2018; Reuver et al., 2018). Only little research, however, strives for a distinctive theorization on DVCN survival. More specifically, DVCN research sheds little light on value co-creation capabilities that network orchestrators should develop, maintain, and use to effect DVCN survival. For instance, Scherer et al. (2015) draw on value co-creation to examine whether a shift from personal to digital, in-store self-service channels affects dyadic customer-firm relationships. Sarker et al. (2012) employ the relationship between a software vendor with its licensed third parties to examine three modes of dyadic value co-creation within a distributed business-to-business context. Breidbach and Maglio (2016) investigate how multiple actors in complex B2B service systems with multiple consulting firms co-located at one client's location to co-create value contingent on IS. Lusch and Nambisan (2015) identify the role of digital infrastructures in enabling distributed actors to become ambidextrous as a research priority of IS. Ghazawneh and Henfridsson (2013) theorize inherent control-emergence tensions in facilitating DVCN survival. They attest a delicate tension in boundary resource design between maintaining DVCN control and, at the same time, stimulating emergent third-party development.

The outlined related work faces limitations that this research seeks to overcome. First, related work draws on extreme cases, such as Apple's iOS platform. However, in increasing generality rather than plausibility, it would be useful to conduct studies using a more representative case selection technique (Yin, 2009) for generality across empirical settings beyond an extreme case. Second, related work does not offer testable value co-creation capabilities for DVCN survival. Third, related work does not explicitly integrate the perspective of third parties in investigating DVCN survival. Finally, partly because of the theoretical shortcomings outlined above, research on DVCN survival has so far not developed prescriptive design knowledge. From a design science research perspective (Gregor & Hevner, 2013), an empirical understanding of the design practices of today's DVCNs would thoroughly inform design theories. To conclude, while little, yet existent, related IS research covers DVCNs in the form of conceptual models and research agendas (e.g., Eaton et al., 2015; Lusch and Nambisan, 2015), DVCN survival remains under-researched.

6.3 Research Design

This section details the content of the two modules as discussed in this chapter (for an overview, see Table 6.1 and Sect. 6.1). Within each module, we discuss how the corresponding SRQ is reflected in the constituent studies, respectively. In turn, Sect. 6.4 describes the results of the individual studies.

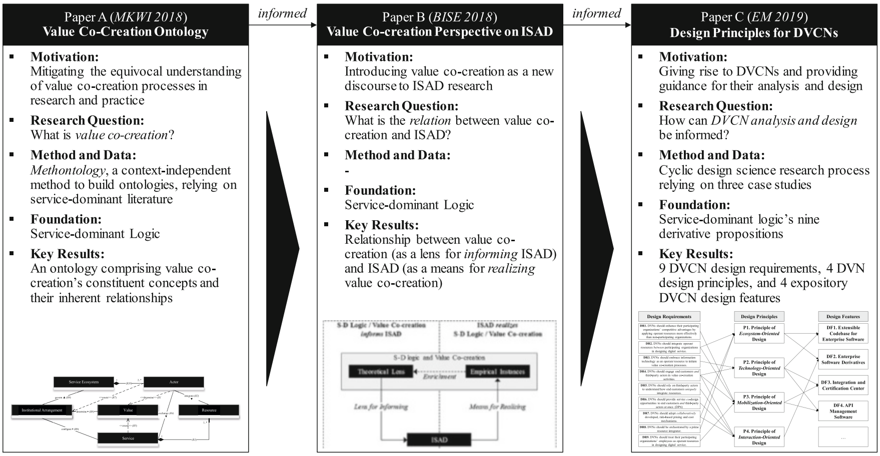


Fig. 6.2 Overview of studies in module 1

6.3.1 Module 1: Conceptual Foundation of DVCN Analysis and Design

Considering the highly diverse understandings and conceptualizations of DVCNs in current literature (see problem 2), module 1 (see Fig. 6.2) aims to understand how a value co-creation perspective on ISAD informs the analysis and design of DVCNs (SRQ1). Primarily, this requires understanding the value co-creation concept itself (What is value co-creation? What are its constituent concepts? How are these concepts related?). Furthermore, it requires understanding how value co-creation and ISAD interrelate (How does value co-creation inform ISAD?). Ultimately, we aim to understand DVCN analysis and design (How can DVCN analysis and design be informed?). SRQ1 therefore has three aspects (see table):

- What is *value co-creation*?
- What is the *relation* between value co-creation and ISAD?
- How can *DVCN analysis and design* be informed?

Aspect (i) An ontology offers a pragmatic approach to structure and codify knowledge about the concepts, relationships, and axioms/constraints pertaining to the value co-creation domain (Kishore et al., 2004b).

To systematically develop a value co-creation ontology, Study A opts for *Methontology*, a step-by-step and well-structured methodology to build ontologies independent of the domain of interest (Fernández-López et al., 1997, p. 36). *Methontology* has been frequently used in different disciplines such as computer science, IS, and law (e.g., Corcho et al., 2005; Fernández López et al., 1999) to guide ontology development. It creates a pool of 64 papers required to uncover S-D logic's concepts underpinning value co-creation. Building on Cooper (1988) taxonomy of literature reviews, the authors choose a review that integrates and generalizes value co-creation research. They include relevant studies on value co-creation from the perspective of S-D logic that are published in the leading marketing and IS journals.

Aspect (ii) Study B chooses a theoretical approach in promoting value co-creation as a relevant conceptual basis to advance ISAD research. To organize ISAD research, the authors distinguish four ISAD research streams considering influential ISAD scholars (e.g., Necco et al., 1987; Frank, 1999; Iivari et al., 2000; Wand and Weber, 2002) as well as studies published in three special issues on ISAD (Briggs et al., 2004; Parsons et al., 2005; Iivari et al., 2006). The study argues that value co-creation opens new pathways to ISAD, and, in turn, ISAD leverages the realization of value co-creation. The authors highlight the pivotal role of theory-informed ISAD approaches to afford a sound basis for developing advantageous ISAD methods and ISAD techniques.

Aspect (iii) The value co-creation ontology guides a derivation of DVCN design requirements based on which Study F derives DVCN design principles. Moreover, the study presents expository DVCN design features that illustrate specific technical ways to instantiate the proposed design principles.

The authors organize the outlined results in a tripartite organizing structure of interrelated DVCN requirements, principles, and features. To this end, they adopt Sonnenberg and vom Brocke (2012, p. 392)'s cyclic design science research (DSR) process and its extension by Abraham et al. (2014). This process (i) incorporates an effective design-evaluate-construct-evaluate pattern and (ii) includes multiple evaluation episodes throughout a single iteration of a DSR process. These episodes ensure a continuous assessment of the progress achieved in devising the design principles (Abraham et al., 2014) and (Sonnenberg & vom Brocke, 2012, p. 390).

6.3.2 Module 2: Value Co-creation Capabilities for Platform Survival

Module 2 (see Fig. 6.3) aims to understand how digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival (SRQ1). Primarily, this requires understanding digital platforms' architectural dimensions. Furthermore, it requires understanding the constituent characteristics of these architectural dimensions to distinguish digital platforms. Ultimately, building on these dimensions and characteristics, we aim to analyze value co-creation capabilities that promote digital platform survival. SRQ2 therefore has three aspects (see Table 6.3):

- How does socio-technical complexity manifest in digital platforms' *architectural dimensions*?

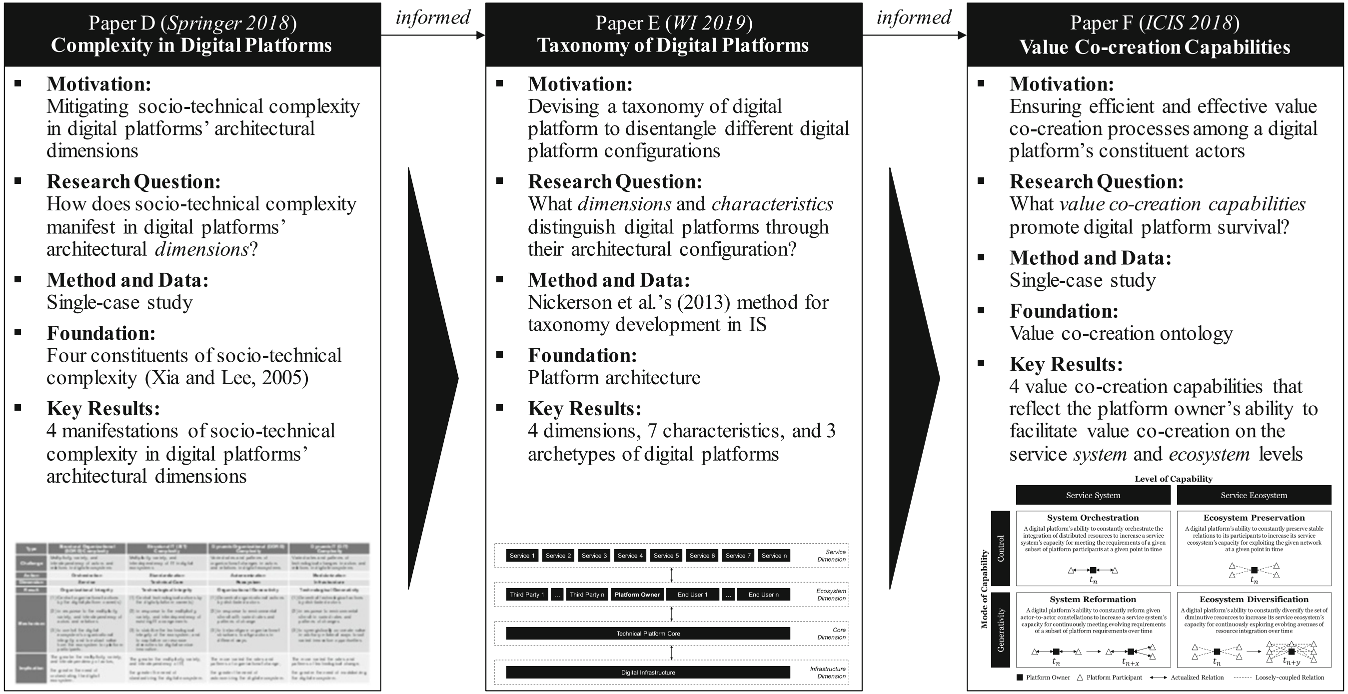


Fig. 6.3 Overview of research design for module 2

Table 6.3 Module 1: Studies A, B, and C in relation to Sub-research Question 1

SRQ1: How does a value co-creation perspective on ISAD inform the analysis and design of DVCNs?			
	Study A: Value Co-creation Ontology—An S-D Logic Perspective	**Study B:** A Value Co-creation Perspective on ISAD	**Study C:** Design Principles for Digital Value Co-Creation Networks
Relation between Studies and SQR1			
(i) What is *value co-creation*?	●	●	◑
(ii) What is the *relation* between value co-creation and ISAD?	◑	●	◑
(iii) How can *DVCN analysis and design* be informed?	○	◑	●
Key Contribution of Study to SRQ1	Value co-creation ontology for ISAD from an S-D logic perspective synthesizing value co-creation's constituent concepts and their relationships as a first step toward reflecting value co-creation in ISAD	Introducing value co-creation as a new discourse to ISAD research to argue that (i) value co-creation provides a novel perspective to ISAD; and that (ii) value co-creation-informed IS designs realize value co-creation	Through an iterative research process, this research derives design requirements and design principles for DVCNs, and eventually discusses how these design principles can be reflected in design features for DVCNs
Legend: ○ No relation to SRQ1; ◑ Partial answer to SRQ1; ● Full answer to SRQ1			

- What *dimensions* and *characteristics* distinguish digital platforms through their architectural configuration?
- What *value co-creation capabilities* promote platform survival?

Aspect (i) Study D opts for an exploratory case study research design (Eisenhardt, 1989; Sarker et al., 2012; Yin, 2009) to study digital platforms in their real-life context through recursive, iterative data collection and analysis steps that eventually help derive explorative insights. The authors choose a single-case study approach owing to the inherent and multifaceted complexity of digital platforms and their respective ecosystems. As digital platforms are poorly understood ((Reuver et al., 2018)) and as the field of platform research is heterogeneous and young (Thomas et al., 2014), developing a theoretical framework and formulating hypotheses upfront is hardly feasible (Eisenhardt, 1989). The investigated digital platform *Helix Nebula—The Science Cloud* is constituted of four leading European IT providers that started cooperating with third parties over a shared digital platform to deliver cloud services to client organizations. Value-destroying high levels of socio-technical complexity in *Helix Nebula*'s platform architecture increasingly inhibited *Helix Nebula*'s survival (Table 6.4).

Aspect (ii) Study E adopts a step-by-step and well-structured method for taxonomy development method in IS (Nickerson et al., 2013). This method has been frequently used in IS research (e.g., Prat et al., 2015; Siering et al., 2017). As an input for Nickerson et al. (2013)'s *empirical-to-conceptual* (E2C) and *conceptual-to-empirical* (C2E) approaches, the authors review digital platform literature (Webster & Watson, 2002) to not only derive dimensions and characteristics from research

Table 6.4 Module 2: Studies D, E, and F in relation to Sub-research Question 2

SRQ2: How do digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival?			
	Study D: Socio-technical Complexity in Digital Platforms	**Study E:** Taxonomy of Digital Platforms	**Study F:** Capabilities for Digital Platform Survival
Relation between Studeis and SRQ2			
(i) How does socio-technical complexity manifest in digital platforms' *architectural dimensions*?	●	●	◐
(ii) Which *dimensions* and *characteristics* distinguish platforms through their architectural configuration?	◐	●	◐
(iii) What *value co-creation capabilities* promote platform survival?	○	◐	●
Key Contribution of Study to SRQ2	Four architectural dimensions of digital platforms and measures to reduce socio-technical complexity within these dimensions	A taxonomy that synthesizes digital platforms' architectural *dimensions* and *characteristics* to differentiate digital platforms	Detailed empirical account and analysis of a digital platform's key value co-creation capabilities to promote digital platform survival
Legend: ○ No relation to SRQ2 ◐ Partial answer to SRQ2 ● Full answer to SRQ2			

(C2E) but also to scrutinize digital platform instances studied in previous research to inform our taxonomy (E2C).

Aspect (iii) Study F adopts an exploratory single-case approach for the same reasons as outlined in aspect (i). The authors study a B2B digital platform for enterprise software that has grown globally since its launch in 2012. The context of this digital platform is particularly relevant for our study for two reasons. First, the owner of this digital platform has experimented with several other digital platforms before 2012. Thus, longtime experience with several previous digital platform attempts is included in capabilities built on digital platform survival in the setup of the studied digital platform from the outset. Second, it represents an exemplary case in that its thriving ecosystem is ideally suited to identify capabilities for digital platform survival.

The studied digital platform, which we refer to as *DP* (a pseudonym), is a platform-as-a-service for creating new applications or extending existing applications in a secure cloud computing environment to ultimately integrate end user organizations' data and business processes. *DP* is managed by its *owner*, which we refer to as *DP-Owner* (a pseudonym), a leading global enterprise software vendor. To complement and market *DP*, *DP-Owner* collaborates within a large digital ecosystem, which we refer to as *DP-Eco* (a pseudonym), with 13,000 build, run, sell, and service *partners* (see Fig. 6.4). Build partners design and develop applications, software, and integrated solutions based on *DP-Owner* technology and its platform. Run partners offer private- or public-cloud-deployed services to their end user organizations based on *DP-Owner* solutions. Sell partners resell *DP-Owner* solutions while managing an entire service's life cycle at the end user organiza-

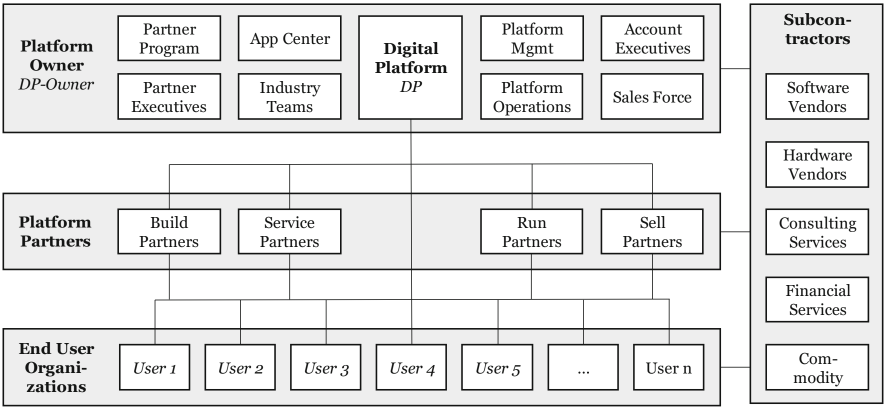


Fig. 6.4 Overview of the studied digital platform's ecosystem

tions, including pre-sales, sales, implementation, and maintenance. Service partners provide consulting and implementation services to end user organizations through the design, implementation, and integration of *DP-Owner* solutions. *DP* partners contribute resources such as industry competence, end user-specific knowledge, close relationships with end user organizations, reach to end user organizations in each geographical location, and human resources capable of serving as sales force, consultants, and augmenting developers.

Further, *DP* has attracted 130 corporate clients as *end user organizations*, most of which are multinational large enterprises. End user organizations typically operate massive arrangements of interconnected systems and technologies that had been introduced over many years and for different purposes. Against this backdrop, end user organizations opt for *DP* to obtain lower-cost state-of-the-art technology, more reliable and versatile technology, finely customized IT solutions, high levels of engagement, faster implementation of IT solutions, and *DP* partners that speak the same language as end user organizations for improved user-partner communication. Finally, *DP-Owner*, *DP* partners, and *DP* end user organizations hire a whole microcosm of *subcontractors* to reduce costs and to mitigate project risks. *DP* subcontractors provide software, hardware, consulting, financial services, and commodities (e.g., electricity for server plants). Figure 6.4 summarizes the actors in the *DP*'s respective ecosystem.

6.4 Results

This section summarizes the relationship between the six studies and the research questions of this chapter (see Table 6.5). It also details the results of the individual studies after a brief synopsis of each study.

Table 6.5 Summary of study contributions to the research questions

Study		RQ	SRQ1	SRQ2
A	*Value Co-creation Ontology—A Service-dominant Logic Perspective*	○	◑	○
B	*A Value Co-creation Perspective on IS Analysis and Design*	○	◑	○
C	*Design Principles for DVCNs—A Service-dominant Logic Perspective*	◑	●	○
D	*Socio-technical Complexity in Digital Platforms*	◑	○	◑
E	*Taxonomy of Digital Platforms: A Platform Architecture Perspective*	○	○	◑
F	*Capabilities for Digital Platform Survival: Insights from a B2B Platform*	●	○	●

Legend:

○ No relation to RQ ◑ Partial answer to RQ ● Main contribution to RQ

Research Questions:

RQ	What value co-creation capabilities promote DVCN survival?
SRQ1	How does a value co-creation perspective on ISAD inform the analysis and design of DVCNs?
SRQ2	How do digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival?

6.4.1 Study A

Synopsis This study starts with the premise that ISAD should account for the reorientation from a G-D logic to an S-D logic of economic exchange (Blaschke et al., 2018). One of the primary steps toward reflecting S-D logic and value co-creation in ISAD is to develop an ontology of value co-creation based on S-D logic's theoretical foundations. To this end, Study A develops an ontology of value co-creation for IS from an S-D logic perspective. Through employing *Methontology*, the authors synthesize the evolving S-D logic literature into a glossary of value co-creation's constituent concepts and represent their inherent relationships. The authors do so in the structure of a computational, semi-formal, and domain-level ontology (Sharman et al., 2004).

Results Drawing on value co-creation ontology's core concepts (see Table 6.2), value is always uniquely and phenomenologically determined by the *actor* being supplied with a service (R1). As such, S-D logic underscores that *value* occurs when an actor integrates its resources (R2). Therefore, a service requires at least one *resource* (R3), and it is exchanged between actors to access, adapt, and integrate resources among themselves for the benefit of another actor or the actor itself (R4). An instance of a designed, offered, and exchanged service creates an instance of value once the service is received and used by an actor (R5). To enable and constrain exchange of service, each service instance is configured guided by *institutional arrangements* (e.g., rules, norms, beliefs) (R6) that are created by actors (R7). Being humanly devised, institutional arrangements also determine value in that they impact an actor's interpretation and determination of what is valuable and what is not (R8). Actors are then connected by their shared institutional logics and mutual value co-creation, both of which govern and evaluate the emergence of the nested and overlapping *service ecosystems* (R9). As such, institutional arrangements are key in fostering cooperative and coordinated behavior among actors. Eventually, service ecosystems are composed of at least two loosely coupled actors (R10).

6.4.2 Study B

Synopsis This research note argues that ISAD should account for novel emergent business logics that require to rethink the way a business environment is conceptualized and, consequently, the way IS as part of such a business environment are analyzed and designed (Haki et al., 2019). This research note builds on S-D logic and its core concept of value co-creation to introduce a new discourse and to outline an agenda for future ISAD research.

Results The authors promote a mutual relationship between value co-creation and ISAD: while (i) value co-creation opens new pathways to analyze and design IS, (ii) ISAD leverages the realization of value co-creation. The authors motivate two main future research questions. First, prospective research should address the question of how the lens of S-D logic and value co-creation *informs* ISAD. This research question is in line with existing IS research that promotes S-D logic and value co-creation as a theoretical lens. Second, prospective research should investigate how ISAD realizes the shift to S-D logic and value co-creation. This research question is in line with IS research concerning the realization of S-D logic and value co-creation through IT. It is also in line with recent calls to better understand the nuances of IS as influential resource in value co-creation (Akaka & Vargo, 2014). These results extend existing IS literature through entangling the value co-creation and ISAD concepts as well as explicating a future research agenda on these entangled concepts.

6.4.3 Study C

Synopsis This study passes from theoretical consideration to actionable design guidance asking "What are the principles for guiding the design of DVCNs that account for the requirements of value co-creation?" (Blaschke et al., 2019b). Owing to S-D logic's distinctive conceptualization on how value is co-created in networks, the authors employ S-D logic as a kernel theory to guide their derivation of DVCN design requirements. Accounting for these requirements, the authors derive DVCN design principles. Ultimately, for an illustrative DVCN instance, expository DVCN design features illustrate specific technical ways to instantiate the proposed design principles. These results are then organized in a tripartite organizing structure of interrelated DVCN requirements, principles, and features.

Results Nine DVCN design requirements and four DVCN design principles result. DVCN design requirements represent the problem space of DVCN design in that they capture generic requirements that any instance of DVCNs should meet to survive (Baskerville & Pries-Heje, 2010; Walls et al., 1992). DVCN design principles embody prescriptive knowledge that bridges the problem space (requirements) and solution space (features) of DVCN design (Chandra et al., 2015; Gregor & Hevner, 2013). Design principles serve as means to convey design knowledge that contributes beyond context-bound DVCN instantiations (Chandra et al., 2015). Design principles also constitute general solution components technologies that can be instantiated into several exemplars of DVCNs (Iivari, 2015).

6.4.4 Study D

Synopsis The digitalization case as reported in this chapter pertains to the digital platform *HelixNebula*. This digital platform suffered from value-destroying socio-technical complexity along four architectural dimensions. Consequently, *Helix Nebula* implemented four consecutive and interrelated actions to counteract socio-technical complexity. First, it modeled its digital ecosystem entailing platform owners, partners, clients, and subcontractors. Second, it agreed on a shared understanding of socio-technical complexity comprising four constituents: structural organizational, dynamic organizational, structural IT, and dynamic IT complexity. Third, it identified manifestations of these constituents in its digital ecosystem. Fourth, it took according countermeasures to reduce these manifestations. While two countermeasures (*orchestration* and *standardization*) reflect the need of maintaining organizational and technological integrity, the other two (*autonomization* and *modularization*) reflect the need of maintaining organizational and technological elasticity in digital ecosystems.

Results First, facing considerable challenges in analyzing its evolving digital ecosystem, capturing all dimensions and characteristics of socio-technical complexity in digital platforms proved intricate. In effect, *Helix Nebula* managers have favored the parsimonious and succinct framework presented in this work conversely. Second, *Helix Nebula* managers adopt an ambidextrous approach to reducing complexity. That is, successful digital platforms balance (i) top-down, central control imposed by platform owners and (ii) bottom-up, decentral *generativity* imposed by platform partners, clients, and subcontractors. *Third*, complexity in digital platforms can pose both *good effects* (enabling, rewarding, value-adding, required, desirable) and *bad effects* (constraining, unrewarding, value-destroying, unrequired, undesirable).

6.4.5 Study E

Synopsis After defining key concepts in the digital platform context, the authors follow (Nickerson et al., 2013) to organize digital platforms' dimensions and characteristics in a digital platform taxonomy (Blaschke et al., 2019a). By instantiating the taxonomy with 34 digital platform instances, the authors derive three digital platform archetypes that capture prototypical configurations of digital platform profiles with similar characteristics. *Archetype* here refers to "a set of structures and systems that consistently embodies a single interpretive scheme" (Greenwood & Hinings, 1993, p. 1055). Platform architecture serves as a focused perspective to capture the configuration of a given digital platform's components. *Platform architecture* here refers to the fundamental organization of a digital platform, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution (Cennamo et al., 2018; Thomas et al., 2014).

Results The findings suggest that digital platforms exhibit characteristics on at least four layered dimensions—namely, *infrastructure*, *core*, *ecosystem*, and *service* dimensions (e.g., Karhu et al., 2018; Kazan et al., 2018). Afforded by the adopted platform architecture perspective, these dimensions reflect the socio-technical and complex architecture of digital platforms (Reuver et al., 2018).

While the *core* dimension appreciates a set of stable technical core artifacts, the *infrastructure*, *ecosystem*, and *service* dimensions capture the dynamic periphery of platform components. The findings further suggest that digital platforms that exhibit similar characteristics belong to one of at least three digital platform archetypes—namely, *orchestration*, *amalgamation*, and *innovation* platforms.

6.4.6 Study F

Synopsis This study identifies four capabilities for digital platform survival. It draws on, integrates in, and extends digital platform research (*targeted literature*) (Constantinides et al., 2018; Reuver et al., 2018). It specifically studies digital platforms' survival (*phenomenon of interest*) through the vantage point of value co-creation (*employed theoretical lens*) (Ranjan & Read, 2014). To answer the research question "Which value co-creation capabilities promote digital platform survival?", the authors study an established B2B digital platform for enterprise software—one that has thrived globally since its launch in 2012. Detailed empirical account and analysis of a digital platform's key capabilities to promote digital platform survival are provided (*resulted insights*).

Four theorized capabilities for digital platform survival—*system orchestration*, *ecosystem preservation*, *system reformation*, and *ecosystem diversification*—are specified based on their respective modes and levels. For each capability, the authors provide an explanation (following a tripartite *challenge-capability-outcome* structure) and empirical evidence. In addition, they further support each capability relying on existing literature. While supporting arguments are merely illustrative to reflect existing discussions in the literature, our set of explored capabilities can be further used as a basis to categorize the existing body of knowledge in the digital platform literature.

6.5 Discussion

The six presented studies' findings, as part of the ValCoLa project at the University of St. Gallen, suggest that DVCN survival is considerably dependent on how efficiently and effectively different actors jointly create value for end users. The chapter at hand, therefore, argues that DVCN survival is contingent on its orchestrator's capabilities to ensure efficient and effective value co-creation processes among the DVCN's constituent actors. The two modules investigate these value co-creation capabilities that promote DVCN survival.

Module 1 develops the conceptual foundations for the analysis and design of DVCNs. We synthesize the value co-creation concept as a perspective on ISAD to inform the analysis and design of DVCNs. Thereby, module 1 answers SRQ1 (*How does a value co-creation perspective on ISAD inform the analysis and design of DVCNs?*). Module 1 results in prescriptive knowledge in the form of a tripartite organizing structure of interrelated and value co-creation-informed requirements, principles, and features for DVCN design. The design requirements are built upon S-D logic's derivative propositions (Lusch et al., 2007)—representing managerial implications of S-D logic. The design principles—namely, *ecosystem-*, *technology-*, *technology mobilization-*, and *technology interaction-oriented design*—represent a general solution for DVCN design that addresses the DVCN design requirements. An expository DVCN case illustrates how to map the Stressgeneric DVCN design principles to specific DVCN design features. The DVCN design principles guide organizations in designing DVCNs that account for the requirements of value co-creation.

Module 2 investigates digital platforms, exemplars of DVCNs, to examine their dimensions and organizes these dimensions' characteristics in a digital platform taxonomy to eventually derive value co-creation capabilities for digital platform survival. Thereby, module 2 answers SRQ2 (*How do digital platforms' dimensions and characteristics inform the analysis of value co-creation capabilities for digital platform survival?*).

Module 2 results in descriptive knowledge in the form of four value co-creation capabilities. While two capabilities (*system orchestration*) and (*system reformation*) reflect the platform owner's ability to facilitate value co-creation processes on the service system level, the other two capabilities (*ecosystem preservation* and ecosystem diversification) reflect the owner's ability to facilitate value co-creation processes on the service ecosystem level. In what follows, we generalize module 2's digital platform-specific findings.

6.5.1 Analytic Generalization

This subsection answers the main research question (RQ: *What value co-creation capabilities promote DVCN survival?*) through an analytic generalization of module 2's digital platform-specific findings. While module 2 investigates *digital platforms* as exemplars of DVCNs, this subsection argues why and how module 2's results apply for DVCNs in general. In distinction to statistical generalization, we adopt *analytic generalization* for this purpose. Referred to as the extraction of a more abstract level of ideas (i.e., DVCNs) from a set of case study findings (i.e., digital platforms), analytic generalization is used for generalizing the findings from a *case study* (Yin, 2013). In this, it resembles experiments, which make no claim to statistical representativeness, but instead assume that their results contribute to a general theory of the phenomenon. Specifically, we answer why and how module 2's findings can be generalized.

Why Generalizable? The following arguments outline what contributes to the generalizability of module 2's findings from 34 digital platform instances (Study E) to a more general theory of DVCNs. First and foremost, digital platforms represent common DVCN exemplars frequently adopted by contemporary organizations. With their set of stable technical core artifacts augmented by peripheral third-party derivatives, and associated organizational arrangements (Reuver et al., 2018), digital platforms are well suited to empirically investigate the more general DVCN phenomenon. We opt for digital platforms as they (i) facilitate the integration of resources in orchestrated networks of actors (Sarker et al., 2012), (ii) become increasingly valuable when more third parties add their derivatives (Parker et al., 2017), and (iii) exemplify the DVCNs of the most valuable companies (FXSSI, 2019).

How Generalizable? Here we outline to what extent the findings of Study F can be generalized to a more general theory of DVCN survival. Study F studies an established digital platform for enterprise software. Its results provide detailed empirical account and analysis of the platform owner's key capabilities (viz., *system orchestration*, *ecosystem preservation*, *system reformation*, and *ecosystem diversification*) to promote digital platform survival. We argue that these four organizational capabilities in general promote the survival of any DVCN exemplar, such as self-service, e-commerce, online support, email, or IT-mediated projects (e.g., consulting projects) (Breidbach & Maglio, 2016). In what follows, we elaborate on why these four capabilities are required for any DVCN to effectively and efficiently co-create digital service.

We begin with two system-level capabilities. As DVCNs are constituted of networked multi-actor settings (re)formed to (re)conform to a given service beneficiary's needs, DVCN survival is generally contingent on a capability to align organizational and technological resources in the DVCN's multiple service systems to meet end user requirements (*system orchestration*). As these service systems change during value co-creation processes among DVCN orchestrators, third parties, and service beneficiaries, DVCN survival is also generally contingent on a capability to continuously reconfigure the DVCN's service systems to meet the evolving requirements of a given subset of end user organizations (*system reformation*).

We continue with two ecosystem-level capabilities. As DVCNs are inextricably intertwined with and facilitated by generative digital infrastructures, DVCN survival in general demands a capability to facilitate the preservation of the DVCN's stability to maintain viable long-term relations with all actors (*ecosystem preservation*).

Finally, as DVCNs are aimed at effective digital service co-created by DVCN orchestrators, third parties, and service beneficiaries, DVCN survival generally demands a capability to continuously capture and enrich the DVCN's diversity to afford an ever-increasing amount of diminutive resource sets (*ecosystem diversification*). Hence, we overall argue that Study F's digital platform-specific findings apply to DVCNs in general.

6.5.2 Limitations and Future Research

The conceptual foundation of DVCN analysis and design (*module 1, SRQ1*) faces limitations. Since Study A anchors its approach in S-D logic with its rigorous, yet specific, interpretation and conceptualization of the value co-creation process, the ontology does not unify alternative and/or complementary conceptualizations of value co-creation. For instance, co-production is potentially abstracted by but not explicitly integrated into S-D logic (Ranjan & Read, 2014). Therefore, future research may holistically unify McColl-Kennedy et al. (2012) catalogue of 27 different definitions of value co-creation. Also being purposefully limited to S-D logic's conception of value co-creation, Study B discusses a future research agenda for the two questions of (i) how the lens of S-D logic and value co-creation informs ISAD and (ii) how ISAD realizes the shift to S-D logic and value co-creation in practice. In turn, Study C's DVCN design requirements, principles, and features are limited in their naturalistic evaluation comprising real tasks, real systems, and real users (Sonnenberg & vom Brocke, 2012, p. 396). We suggest evaluating these design requirements, principles, and features in the contexts of different DVCNs.

The empirical analysis of value co-creation capabilities for digital platform survival (*module 2, SRQ2*) also faces limitations. Study D faces "the huge challenges in studying large-scale complex phenomena" (Tilson et al., 2010, p. 751) potentially limiting the validity of which digital platform configurations benefit or constrain survival. Therefore, future research should more deeply differentiate between *value-adding* and *value-destroying* complexity in theorizing digital platform survival. In turn, while Study E descriptively investigates digital platforms' dimensions, characteristics, and archetypes, it does not prescribe how to effectively configure digital platforms. Prospective research may thus investigate how different digital platform configurations translate into which outcomes. Moreover, Study F is limited in its focus on *value co-creation* capabilities. We provide no specific insights on additional capabilities that promote digital platform survival. Deriving additional capabilities from different perspectives (e.g., business model, competition, or boundary resource perspectives) would complement the set of four capabilities presented in this work.

Lastly, this chapter empirically investigates digital platforms as exemplars of DVCNs; we then answer the *main research question* through an analytic generalization of module 2's digital platform-specific findings (see Sect. 6.5.1). This analytic generalization faces limitations as there are different exemplars of DVCNs beyond digital platforms. For example, DVCNs can have no clearly defined orchestrator (e.g., in open-source initiatives) or shared ownership (e.g., in research consortia or public-private partnerships). In such cases, capabilities to promote survival may unfold differently. Therefore, prospective research is motivated to examine whether alternative DVCN exemplars *do* require alternative capabilities to ensure survival.

6.5.3 Implications

The results of the presented six studies provide insights that extend current discourses on value co-creation and digital platforms. They also offer implications for practice.

Value Co-creation Existing research has mainly focused on classical bilateral relations in value co-creation processes (Aarikka-Stenroos & Jaakkola, 2012). By focusing on DVCN's multilateral relations, this chapter's analysis comprises *many* actors. In addition, value co-creation in S-D logic research mainly remains in its philosophical realm and is limited to theoretical discourses (Grönroos, 2011). Therefore, only limited *detailed empirical* account and analysis of value co-creation's specificities and intricacies are available. Specifically, the chapter demonstrates the pivotal role of end user organizations' feedback in innovating and designing a specific service for any given end user organization's unique context. In empirically witnessing all identified capabilities, we stress the need to account for end user organizations' resources, such as requirements, feedback, experience, knowledge, and skills. Thereby, the role of end user organizations is pivotal in studying digital platform survival. Further, through studying *digital* platforms, module 2 contributes to the emerging discourses on *IS-enabled* value co-creation (e.g., Ceccagnoli et al., 2012; Lusch and Nambisan, 2015; Sarker et al., 2012). Both IS (Lusch & Nambisan, 2015) and service research (Breidbach & Maglio, 2016; Vargo & Lusch, 2017; Wilden et al., 2017) have called for investigating the role of digital technologies in value co-creation. By investigating value co-creation in *digital* platforms, module 2 provides new insights into the previously underserved role of IS in leveraging value co-creation. We demonstrate how digital platforms become means in realizing value co-creation processes and how digital platform survival is contingent on co-creation of value among their constituent actors.

Digital Platforms By investigating digital platform *survival*, this study contributes to a more thorough understanding of what delineates very few vibrant digital platforms to succeed and thrive in the long run, while so many others lie idle. Through spotlighting value co-creation processes as *the* cornerstone of digital platform survival, this research introduces a *multi*-actor perspective to digital platform research. Existing digital platform research predominantly focuses on *either* platform owner or platform partner perspectives (e.g., Ceccagnoli et al., 2012; Sarker et al., 2012). Instead, this chapter offers a multi-actor-role perspective on digital platforms through studying platform *ecosystems* as actor-to-actor assemblages constituted by the platform owner, partners, end user organizations, and subcontractors. Specifically, we highlight the role of end users (a platform's customers), following the seminal premise of the value co-creation concept from an S-D logic standpoint. Study F demonstrates that digital platform survival is contingent on their owners' capabilities to dynamically mobilize and (re)configure *all* actors' resources to satisfy all affected actors' ever-changing requirements.

Practice The six studies reviewed in this chapter guide practitioners in their systematic analysis and design of DVCNs, which is generally a multistage approach that requires repeated checks and rethinking. Adapting to emerging networked business environments is both relevant and complex for managers at network orchestrators. Through reflecting the offered DVCN design requirements, principles, and features (Study C), managers can more clearly analyze requirements and specify according logical DVCN designs. Moreover, managers at network orchestrators are provided with an organizing logic to understand, develop, and apply a set of capabilities (Study F). By applying this logic, they can more clearly define the specific aspects required in realizing value co-creation and in leveraging DVCN survival. This may be especially useful for early design decisions that affect DVCNs' evolution trajectories. Focusing on the identified capabilities, managers might anticipate areas of concern and take appropriate measures. The case narratives of the analyzed DVCN cases themselves serve as a consultable record for managers. Reflecting these capabilities can be valuable for other organizations that may be motivated to develop DVCNs but are unaware of inherent intricacies and required managerial actions.

6.6 Conclusion

This chapter introduced value co-creation as a new perspective on how organizations may best analyze and design their networks of actors in practice while acknowledging the limits of direct control in highly emergent networks (Ghazawneh & Henfridsson, 2013). The development of this perspective is primarily based on S-D logic due to its distinctive and penetrative conceptualization on how value is co-created in emerging, networked business environments (Lusch & Nambisan, 2015). In conclusion, we argue that, from an S-D logic viewpoint, complex DVCNs should not be misconceived as deterministic and directly controllable systems but should rather be analyzed and designed in a way that delicately reconciles a copresence of (i) heterogeneous third parties who leverage generative innovation in the DVCN and (ii) powerful network orchestrators that control the DVCN to some degree.

Acknowledgments The studies presented in this chapter, under the ValCoLa project, have been supported by the Swiss National Science Foundation (SNSF).

Chapter 7
ArchiMate Extension to Value Co-creation: The Smart Airport Case Study

Christophe Feltus, Henderik A. Proper, Andreas Metzger, and Juan Francisco Garcia López

Abstract The design and engineering of collaborative networks and business ecosystems is a discipline that requires an outstanding and upfront attention of the value cogenerated among the parties involved in the business exchanges of these networks. Understanding this value co-creation is undoubtedly paramount, first to adequately sustain the design and the development of the information system that brings about this value, second, to support the communication between the information system designers, and third to allow discovering new co-creation opportunities among the networks companies. In that context, we proposed an abstract language (meta-model) that structures, and provides an explanatory semantics to, the co-creation of value between information system designers, allowing a better definition of the collaboration and of each one of the value propositions. The design of this language is achieved in the frame of the design science theory and accordingly follows an iterative improvement approach based on real case studies from practitioners. This chapter introduces the second iteration of the language based on a real case in a smart airport network.

C. Feltus (✉)
Luxembourg Institute of Science and Technology, Esch-sur-Alzette, Luxembourg
e-mail: christophe.feltus@list.lu

H. A. Proper
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

A. Metzger
University of Duisburg-Essen, Duisburg and Essen, Germany
e-mail: andreas.metzger@paluno.uni-due.de

J. F. G. López
Indra, Alcobendas, Madrid, Spain
e-mail: jfglopez@indra.es

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_7

7.1 Introduction

All engineering steps for information systems (IS) involve a plethora of actors from inside and from outside the company (e.g., software architects, security providers, or consulting companies). These steps include, among others, defining the system requirements, developing software components, testing the system, and deploying appropriate security mechanisms. Traditionally, the relationship between the company and its suppliers aims to generate value in exchange for money. This relationship has been largely investigated through the vector of value exchange and value chains (Vargo & Lusch, 2004, 2008; Wille et al., 2013; Gordijn et al., 2000; Chew, 2016; Proper et al., 2018a). For instance, monitoring a bank information system may be outsourced to a security provider offering a SOC (Security Operation Center) service in exchange of annual fees.

In this chapter, value co-creation (VCC) is investigated as a specialization of value creation and represents the close collaboration between two or more parties to generate value following an ordered set of value co-generation processes inspired by knowledge-intensive business services (KIBS) (Lessard, 2015). The work, as reported in this chapter, was part of the ValCoLa project as mentioned in Sect. 1.3.

Although a plethora of research exists aiming at depicting the fundamentals of VCC, few contributions exist in the area of modeling language for supporting VCC design and deployment. Nevertheless, a common model is needed to facilitate communication among the many different actors. Such a model and modeling language are necessary to describe and to visualize different components of the information system, as well as their underlying relationships and dependencies. As a result, the goal for such a modeling language is to support the process of decision-making and to allow understanding and analyzing the impacts associated to a change of the system architecture on the whole information system.

We propose such a VCC modeling language as an extension of ArchiMate, a standardized enterprise architecture modeling language (Band et al., 2016).

ArchiMate is an open, independent, and non-sector specific language maintained by The Open Group.[1] It supports the description, analysis, and visualization of architectures in an unambiguous way, by structuring the enterprise elements on different layers. These layers cover the business concepts (like the collaboration, the process, etc.) down to the very technical one (like the network, the servers, etc.). In particular, ArchiMate proposes two extension mechanisms that allow extending the model and the language to various fields of interest like IS governance or risk analysis (Grandry et al., 2013). In that context, ArchiMate appears to be appropriate as a language to express the value creation and, by the way, the value co-creation.

To illustrate the designed language extension, a case study related to the development of value co-creation in a smart airport is proposed (Feltus et al., 2018d). The airport systems support the complete operations of the airport, including in

[1] https://www.opengroup.org/ArchiMate-forum/ArchiMate-overview.

particular arrival and departure control system (such as assignment of planes to gates), on-site check-in, baggage handling, and security control. In parallel, the airline management systems support the activities of the airline companies in offering transport services to its customers and in particular support ticketing, online check-in, and passenger management. Both airport and airline systems are essential for supporting the execution of the air transport. These systems continuously provide data to facilitate proactive decision-making based on the real context. In parallel, the operation module uses anonymous passenger data to trace passenger flow.

The remainder of the chapter is structured as follows: In Sect. 7.2, we provide relevant background work on VCC. In Sect. 7.3, we present the value creation model and an extension of ArchiMate to express this value creation. In Sect. 7.4 we present a VCC meta-model and an extension of ArchiMate to express the latter. In Sect. 7.5 we illustrate the extension through a case study in the smart airport domain. Finally, Sect. 7.6 concludes and discusses the proposed approach.

7.2 Background

VCC discipline originates from the marketing theory. It aims to define and to explain the mechanisms for the co-generation of value during business exchanges among companies (Vargo & Lusch, 2004, 2008). Vargo and Lusch (2004); Wille et al. (2013) formalize it using a framework for defining VCC in the perspective of the service-dominant logic (S-DL). According to them, service is the basis of all exchanges and focuses on the process of value creation rather than on the creation of tangible outputs. As a result, a service system is a network of agents and interactions that integrates resources for VCC (Vargo & Lusch, 2004). On that basis, value is proposed by a service provider and is determined by a service beneficiary. According to Grönroos (2008), this interaction is defined through situations in which the customer and the provider are involved in each other's practices. Frow et al. (2015) propose a framework to assist firms in identifying new opportunities for VCC. Therefore, they provide a strategically important new approach for managers to identify, organize, and communicate innovative opportunities. In that matter Cesarotti et al. (2016) explore the opportunity to increase the value cocreated in a service process through improved design using multiple channels and (Herzfeldt et al., 2016) analyze the relationship between cloud service provider profitability and value facilitation.

More recently, Chew (2016) argues that, in the digital world, service innovation is focused on customer value creation, and he proposes an integrated Service Innovation Method (iSIM) for analyzing the interrelationships between the design process elements. At the IS domains level, Gordijn et al. (2000) explain that business modeling is not about process but about value exchange between different actors. Accordingly, Gordijn and Akkermans (2001) propose e3value to design models that sustain the communication between business and IT groups. In Weigand (2009), e3value is extended for considering co-creation. Therefore, the authors define the so-called value encounters which consist in spaces where groups of actors interact to derive value from the groups' resources. The financial case used to illustrate our method is modeled with this e3value language (see Sect. 7.3). In the same vein, Razo-Zapata et al. (2016) propose visual constructs to describe the value co-creation process.

7.3 Value Creation Model and Language

In this section, we present the value creation model and an extension of ArchiMate to express this creation, and the next section will extend the language to the value co-creation (see Fig. 7.1).

7.3.1 Model Elaboration

In this subsection, value is defined according to the following three dimensions (see Fig. 7.2): the nature of the value, the method of VC, and the object concerned by VC (Feltus & Proper, 2017a). In the next subsections, each dimension is conceptualized, modeled, and illustrated with real cases.

At a methodological level, the research that we tackle concerns the improvement of co-creation of value in the field of collaborating companies. Accordingly, we

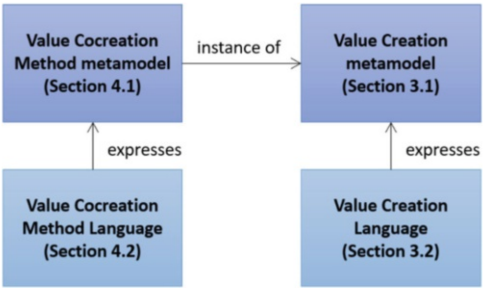


Fig. 7.1 Modeling approach and language extension design (section numbers are relative to this chapter)

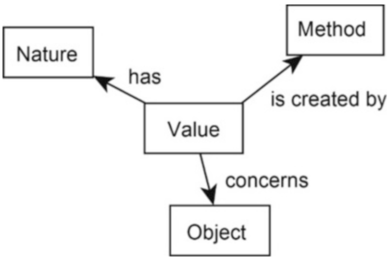


Fig. 7.2 Three value dimensions

have conceptualized and defined a language to support the SPCC process on the basis of the three dimensions of the value creation meta-model. Through this research, we aim to strengthen the organizational capability to improve the design of the SPCC related to the IS. Accordingly, Hevner et al. (2004) explain that the Design Science Research (DSR) paradigm seeks to extend the boundaries of human and organization capability by creating new and innovative artifacts. Practically, provided that we aim to design a new artifact (VCC language) to support the design of the information system value, we acknowledge that this research may plainly be considered in the scope of DSR (Peffers et al., 2007). As advocated by the DSR theory (Hevner et al., 2004; Peffers et al., 2007), the method that we use to design these value dimensions is an iterative approach consisting first of analyzing different instances of the domain under scope, second of extracting the relevant concepts from the instances, and third of designing elementary domain models. For example, to model the nature of the value, we have analyzed some instances of this nature like security, privacy, and quality, we have extracted the more relevant concepts of these domains in Table 7.1, and we have designed the nature of the value model (Fig. 7.3). For the sake of pragmatism, only the last version of the iterations is presented in the next sections.

Table 7.1 Nature of the value

Value reference framework	Nature of the Value examples		
	Nature of the value	Characteristics of the nature of the value	Concerned object
Web Quality Model (Calero et al., 2005)	Quality	Functionality, Reliability, Usability, Efficiency, Portability, Maintainability	Web feature
ISSRM (Matulevicius et al., 2008b)	IS Security	Confidentiality, Integrity, Availability, Non-repudiation, Accountability	Business Asset
HCI (Dix, 2009)	Usability	Learnability, Flexibility, Robustness	Design rules, design knowledge
ReMMo (Feltus et al., 2009)	Responsibility	Accountability (e.g., RACI)	Actor
EA Compliance Model (Foorthuis et al., 2009)	Compliancy	Correctness, Justification, Consistency, Completeness	Acts of software developers
VDML (OMG, 2015)	Generic Value	Factor of benefit, Factor of interest	Business item
Privacy Meta-model (Feltus et al., 2018a; Langheinrich, 2001)	Privacy	Notice, Choice and Consent, Proximity and Locality, Anonymity and Pseudonymity, Security, and Access and Resource	Sensitive Information Hawley et al. (2013)
...			

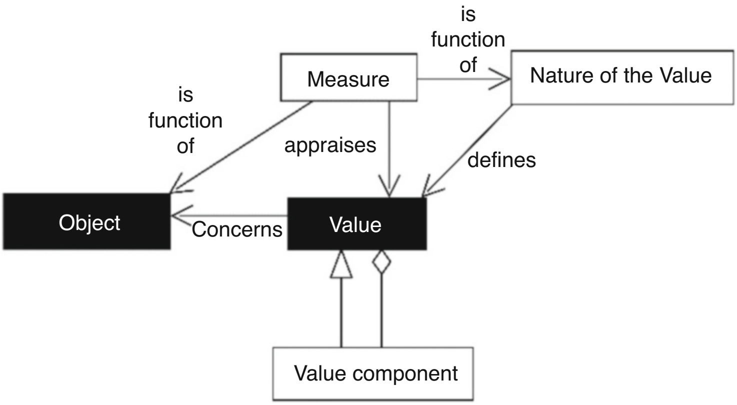


Fig. 7.3 Nature of the value meta-model

7.3.2 Nature of the Value

Value is an abstract concept that expresses a measureable information of a determined nature and which is associated to a well-defined object. According to Zeithaml (1988), value implies some form of assessment of benefits against sacrifices. Most researches that focus on depicting the semantic of value agree on the abstract character of the latter, mostly generated by the different types of existing value nature (Alves et al., 2016). Whatever, two main categories of value nature emerge depending on the context: value at provider side vs. value at customer's side. When value is perceived at the provider side, economists largely argue that the latter is created (manufactured) by the firm and distributed in the market, usually through exchange of goods and money (Smith, 1963). This nature of value has for a long time traditionally been represented by the possession of wealth and money. However, it is also worth to note that considering the provider in the context of the digital society expands this narrow interpretation to the consideration of other value elements, like the information collected on the customers which, afterward, fills the bill of economic increase (Nyman, 2016). On the customer side, value generated by a transaction never refers to money but consists in other wealth, which contributes in sustaining and supporting the customer's own business.

Let us take the example of a SME that outsources the privacy management of its assets to dedicated enterprises, in order to remain being focused on its core business. In this case, the privacy nature of the value is traditionally expressed with well-defined characteristics (e.g., pseudonymity, anonymity, consent, etc. (Table 7.1) that are specifics for privacy). Moreover, two types of value are created by this outsourcing: a direct value (privacy of the assets) and an indirect value (more time for core activities). Over and above that, this transaction happening with a customer being a citizen also contributes to the latter's improvement of his well-being as observed in Korkman (2006) that asserts that value for customer means that after they have been assisted by a self-service process or a full-service process, they are or feel better off than before.

As summarized in Table 7.1, our analysis to understand and to define the nature of the value has been performed by tackling a set of frameworks in different areas like security, quality, compliancy, privacy, responsibility, and so forth. For instance, we have analyzed the information systems security risks management (ISSRM) framework that addresses the IS security (Matulevicius et al., 2008b). ISSRM characterizes security through integrity, confidentiality, non-repudiation and accountability, and availability, and the latter concerns business asset of the company. Moreover, according to Theoharidou et al. (2012), we acknowledge that the abovementioned characteristics also constitute complementary types of value. Based on our review, we have observed that value is an abstract concept defined by a well precise nature with well-determined characteristics, that it is measureable, and that it concerns a well-defined object.

The concepts composing the nature of the value model are:

- *Value*—This concept is defined as a degree of worth that concerns something (Calero et al., 2005; Foorthuis et al., 2009) and that improves the well-being of the beneficiary after it is delivered (Vargo & Lusch, 2004).
- *Nature of the value*—Table 7.1 shows that the nature of the value expresses a domain of interest related to which the value will be delivered (e.g., security of the IS, the cost of a transaction, or the privacy of personal data). As a consequence, the nature of the value *defines* the value to be delivered. In the case of the datacenter that archives the data of the bank customers, the nature of the value generated by the datacenter is the *availability* of the customer's data.
- *Value component*—This concept expresses the different elements that constitute the value, or the pillars that found this nature (e.g., availability, confidentiality, portability, etc.). Hence, the value *aggregates* value components and the latter may also, as a result, themselves be other *types of* value. Regarding the case study, one component of the availability is the *accessibility in real time*.
- *Object*—The object concerned by the value is the element from the information system that has significance and is necessary for a company to achieve its goal, and that is be better off after that value is delivered (e.g., software, process, data). From a modeling point of view, the value is associated to an object with a relation of type *concerns* or objective to be achieved. In the case study, the object concerned by the value is the *customers' data*.
- *Measure*—The measure corresponds to a property on which calculations can be made for determining the amount of value expected from a value creation method. This measure (e.g., the *% of time data is available*) can result from different factors impacting value. This corroborates the statement made in Calero et al. (2005), which argues that the value components are measured by means of estimation methods. Accordingly, there exist an association named *appraises* from the concept of measure to the concept of value and an association named *is function of* between the concept of measure and the type of value and between the concept of measure and the object concerned by the value. The first expresses

that the measure is characterized by the nature of the value and the second that the measure also depends on the object concerned by the value. According to Calero et al. (2005), this measure may integrate qualitative and quantitative elementary performance expressions.

Based on the above definitions, the nature of the value has been modeled in Fig. 7.3.

7.3.3 Method of Value Creation

A method of value creation is a formalized activity which contributes to the generation of value. Traditionally, value is acquired by exchanging goods or services and emerges out of its use (Wille et al., 2013). Methods for value creation are the body of techniques and series of steps necessary to create value. This corresponds, at the corporate level, to a bundle of approaches including processes, audits, controls, decisions, etc. Likewise, as for the nature of the value, in order to depict the elements relevant for the creation of value, we have reviewed a set of value creation methods among a plethora of them (Table 7.2).

The methods that we analyzed so far are the "method chunk" (Ralyté, 2004), the risk-based method (Daneva, 2006), the model-driven approaches (Bénaben et al., 2008; Becker & Klingner, 2013), the process based method (Manuj & Mentzer, 2008), the impact assessment (Becker, 2014), and the method by design (CPDP, 2014). By looking more closely to all of them, we observe that these methods have each a dedicated goal, that they are composed of method elements, and that the latter are organized in ordinate steps. For instance, by investigating the model-driven approach, we notice that it has a goal to improve interoperability of enterprises information systems, that it is composed of models, and that three steps are required for model-driven interoperability, to know models design, models integration, and models instantiation.

Among the other methods reviewed, it is interesting to highlight that one of them (method chunk) has as its primary objective the creation of method themselves (Ralyté, 2004).

As a summary and according to our analysis, the concepts which compose the method of value creation are:

- *Method*—The method is a specific *type of* object that defines the means used by the stakeholder to *create* objects and value. According to Table 7.2, a method is *composed* of a set of activities necessary to achieve a dedicated goal. In the same vein, Sein et al. (2011) explain that the elementary quantitative value expressions (the value components) are aggregated by means of selected aggregation methods and quantitative weights to generate the overall value. The method used to create the availability is *the exploitation of a redundancy system (tools and procedures to guarantee redundancy).*

Table 7.2 Method of value creation

Method reference	Method of Value creation examples			
	Method	Goal of the method	Method elements	Activity
Ralyté (2004)	Method chunk	Method creation	Chunk of existing methods	Decomposition of existing methods into method chunks and definition of new method chunks from scratch
Daneva (2006)	Risk-based	Security strategy development	Risk, Costs, Benefits	Analysis of the methods elements and identification of the options that exist in investment decisions
Bénaben et al. (2008); Becker and Klingner (2013)	Model-driven	Improve interoperability of companies information systems – service modeling and configuration	Model – Meta-model	Models design, model integration and model instantiation
Manuj and Mentzer (2008)	Process-based	Risk management for global supply chain	Process, Step, Dependency	Step-by-step execution in a function of the dependency amongst them
Becker (2014)	Impact assessment	Explore social consequences for social security policies	Scenario, Strategy, Impacts, Implementation	Scenario design, Design of strategies, Assessment of impacts, Ranking of strategies, Mitigation of negative impacts, Reporting, Stimulation of implementation, Auditing and ex-post evaluation
CPDP (2014)	By design	Prevent privacy risk from occurring	Project	Project-by-project approach realization
Aier and Winter (2010)	Enterprise integration patterns	Enterprise integration project	Services, Cluster, Patterns	Services clustering
...				

- *Activity*—The activity is an element of the method that corresponds to a unitary task (e.g., analysis, collection of information, or reporting). The activities *compose* the method and are organized and coherently articulated with each other (e.g., if-then-else, process elements ordination, etc.). This relation is modeled using an iterative association of a type: activity *follows* activity. The articulation of activities corresponds to the aggregation from Feltus et al. (2018b). One particular type of activity consists in *generating* resources, for instance, *acquiring a backup tool, maintaining the backup tool, etc.*.
- *Stakeholder*—A stakeholder is a human, a machine, or an organization that is involved in the creation of value at three levels. First, it *performs* the method that generates value (e.g., the risk manager performs a risk analysis); second, it *generates* resources used by the method; and third it *expresses* the value expected after the execution of the method. For example, the *datacenter* is the stakeholder that exploits the redundancy system and the *bank* expresses that it expects availability of the data.
- *Resource*—This element is a *type of* object from the IS that is generated by a stakeholder and that *is used by* an activity composing the value creation method. Resources are typically information and data (e.g., passenger location) but could also consist in computing resources, funding, manpower, etc. For instance, the *backup software* is the resource used by the exploitation of a redundancy system.

Based on the above definitions, the value creation method has been modeled in Fig. 7.4.

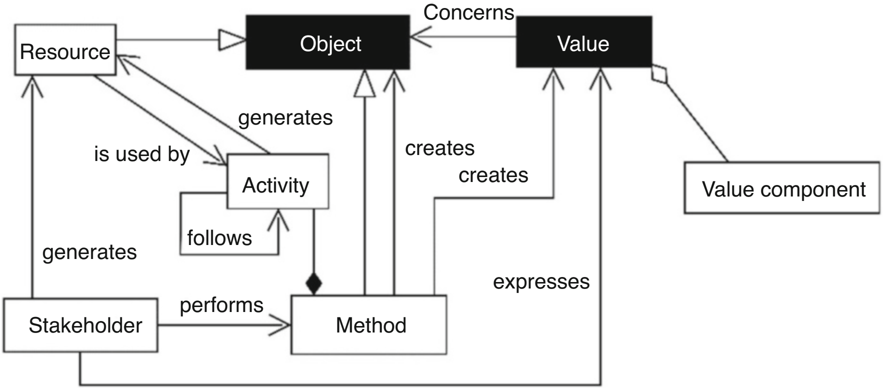


Fig. 7.4 Value creation method meta-model

7.3.4 Object Concerned by the Value

The object concerned by the value corresponds to the elements (mostly existing at the information system level, e.g., information, process, tool, actor) that have significance for a company to achieve its goal. This object exists in a determined environment represented at the information system level by the context, the latter having an influence on the type and the amount of value associated to this object. For instance, a customer browsing history is an object of a data type that has a particular pecuniary value for an airline travel agency which can estimate the value ascribed to a flight ticket for a customer. This value is calculated based on the number of times this flight ticket was viewed on the company website by the customer. At the opposite, this customer browsing history is not an object of value on a drugstore website with fixed prices. Additionally, it is also worth to note that this context has no impact on the nature of the value. For example, privacy in the healthcare sector is defined the same way as in the industry, meaning with the same characteristics.

To collect and to deal with the concepts that are necessary to model the object of value, we assume that each sector of activities, should it be the manufacturing, the finance, or the healthcare sector, for instance, is associated with a specific information system. The latter models the objects composing them and the relationships between these objects, using a dedicated language. In order to focus on the right object of value when defining a business model or when analyzing the co-creation of value, it is important to have an understanding of, and an alignment between, the objects of value of all stakeholders involved.

Sector-specific information systems and enterprise architecture (EA) models and languages are good approaches because they semantically define generic objects and sometimes concrete languages to express the latter. Numerous frameworks have been designed to model IS and EA of various sectors, e.g., CIMOSA (Berio & Vernadat, 2001), HL7 (Dolin et al., 2006), DoDAF (DoD Deputy Chief Information Officer, 2011), etc.

Table 7.3 provides a review of some meta-models and languages to depict the context targeted, the IS under scope, and some examples of objects addressed.

The above frameworks have been systematically reviewed and distilled in order to capture the semantic of their value-related concepts. After that, the meaning of all these concepts has been precisely and methodically compared and synthesized with each other. As a summary and according to this review, the concepts which define the context and the object concerned by the value are:

- *Information system*—The information system encompasses, and *is composed of*, the objects concerned by the value and the stakeholders that benefit from the value created.
- *Context*—The context represents the surrounding of the IS. It includes (1) the constraints on the system in which the value is created and (2) the definition of the borders of this system (e.g., the sector and the sector purpose of the business entity that is concerned by the IS, the rules and regulations related to the sector or the IS, the institutional arrangements, etc.).

Table 7.3 Object concerned by the value

Language+Reference	Object concerned		
	Sector	Information system	Examples of objects
DEMO (Dietz, 2006)	Enterprise	Business Process, Information Systems	Models (Interaction, Business Process, Action, Interstriction, Fact), Actor, Action, ...
ARIS (Scheer & Nüttgens, 2000)	Enterprise	Business process management	Data, Function, Organization, Material, IT resources, or Machine resources, ...
CIMOSA (Berio & Vernadat, 2001)	Production Industry	Industrial information system	Business process, flow, step, function, information, resource and organization aspects, business user, control, capability, ...
HL7 (Dolin et al., 2006)	Healthcare	Clinical document architecture	Organization, Clinical document, Author, Legal Authenticator, Person, product, consumable, ...
DoDAF (DoD Deputy Chief Information Officer, 2011)	Military	DoDAF Meta-Model (DM2)	Guidance, activity, capability, resource, performer, location, information, project materiel, system, service, organization, ...
BSE (Feltus et al., 2015)	Enterprise	Business Service Ecosystem	Service, Capability, Resource, Process, Actor, ...
...			

Accordingly, the context is associated to the information system with an association named *characterizes*. As stated in Li et al. (2017), the context also allows selecting the "performance components [...]" necessary "to define the scope of the performance evaluation problem." Hence, this selection defines a particular context, or viewpoint, for the evaluation of the value. To model this, the concept of context is associated to the measure with a relation named *influence*. Regarding the case study in the financial sector, the context is the *financial regulation*.

Based on the above definitions, the context and the object concerned by the value have been modeled in Fig. 7.5.

Figure 7.6 represents the integrated meta-model of value creation based on the integration of the three value creation dimensions proposed in Figs. 7.3, 7.4, and 7.5. In that figure, the concept of value (co-)creation has been added and consists in a type of value (cf. Sect. 7.3.2).

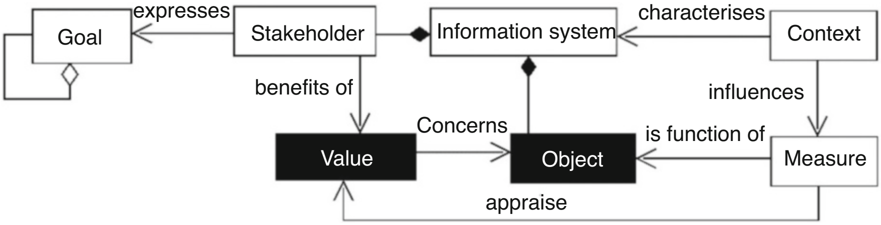


Fig. 7.5 Object concerned by the value

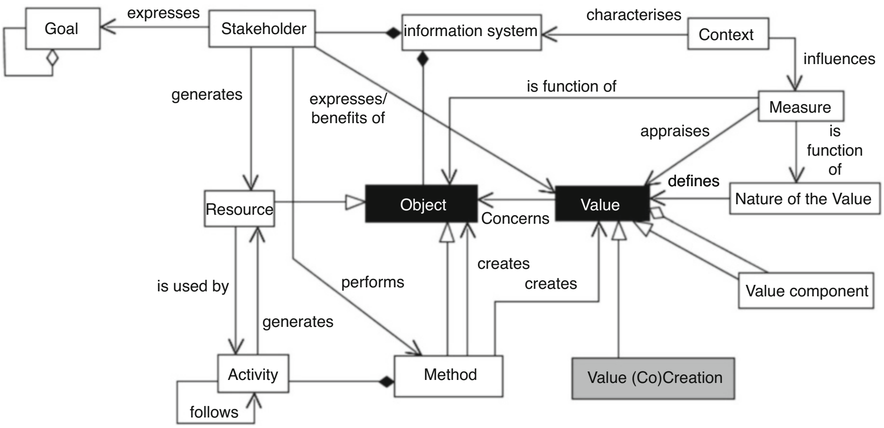


Fig. 7.6 Integrated VC meta-model built on the integration of Figs. 7.3, 7.4 and 7.5

7.3.5 ArchiMate Language Extension to Express Value Creation

In this subsection, we extend the ArchiMate language to the VC domain. Therefore, first we introduce ArchiMate (meta-model), then the ArchiMate language and its extension mechanisms, and finally the extended ArchiMate to VCC.

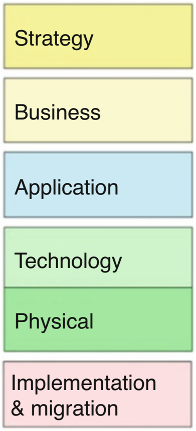
7.3.6 Introduction to ArchiMate

ArchiMate is an enterprise architecture framework (i.e., meta-model and syntax) used by enterprise and IT architects to design business and IT static views and their links, of the corporate architecture (Josey et al., 2016). ArchiMate allows reducing the complexity and proposes means to model and thus better understand the enterprise, and the interconnections and interdependency between the processes, the people, the information, and the systems. Consequently, one objective of ArchiMate is to express and visualize enterprise architecture aspects such as the organizational structure, the business processes, the information processing system, the infrastructure, or the responsibility (Feltus et al., 2012). It permits to ensure uniform semantics of the instantiated models, but it is not really appropriate to enable quantitative analysis.

Another objective of the enterprise architecture is to highlight the creation of business value. For instance, in the ArchiSurance scenario (Josey et al., 2016), the customer needs to "be insured" with the instance "be insured" being a type of "business value." This business value is generated by the business processes which are supported by applications and infrastructures.

Core ArchiMate is structured in six horizontal layers (see Fig. 7.7): strategy, business, application, technology, physical, and implementation and migration. Three of these layers are particularly relevant to express the value co-creation: the *business layer* in light yellow, the *application layer* in blue, and the *technology layer* in light green. All three layers are built with the same type of concepts and the same sort of associations. They are structured according to three aspects (vertical layers). The first aspect concerns the active structure elements which are defined as *entities that are capable of performing behavior*, e.g., a role or an actor. The second aspect regards the behavioral elements which are defined as *units of activity performed by one or more active structure elements*, e.g., a process or a function. The last aspect addresses passive structure elements which are defined as *objects on which behavior is performed*, e.g., a contract or an object. The generic ArchiMate meta-model is shown in Fig. 7.8. This meta-model is also contained in the specification of the ArchiMate standard (Band et al., 2016), albeit with some additional details.

Fig. 7.7 ArchiMate layers



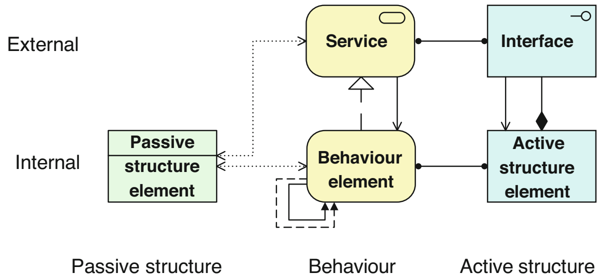


Fig. 7.8 Generic ArchiMate meta-model, adopted from Lankhorst et al. (2017); ©2017 Springer-Verlag Berlin Heidelberg; reprinted with permission

7.3.7 Relevant ArchiMate Symbols

ArchiMate uses a syntax based on symbols and colors, related to the vertical and horizontal layers. Table 7.4 contains ArchiMate elements, definitions, and symbols used during the mapping and integration of both meta-models.

7.3.8 ArchiMate Extension Mechanisms

ArchiMate extension is achieved by integrating its meta-model with the meta-model of the domain that extends it. According to Zivkovic et al. (2007), the integration of two meta-models requires resolving three types of heterogeneities: syntactic, semantic, and structural. For our integration, only the semantic and the structural heterogeneities have been addressed. Indeed, the syntactic heterogeneity aims at analyzing the difference between the serializations of meta-model and, as explained by Parent and Spaccapietra (2000), addresses technical heterogeneity like hardware platforms and operating systems, or access methods, or it addresses the interface heterogeneity like the one which exists if different components are accessible through different access languages. The structural heterogeneity exists when the same meta-model concepts are modeled differently by each meta-model primitives. This structural heterogeneity has been addressed together with the analysis of the conceptual mapping and the definition of the integration rules. Finally, the semantic heterogeneity represents differences in the meaning of the considered meta-model elements and must be addressed through elements mapping and integration rules. Regarding the mappings, three situations are possible: no mapping, a mapping of a type 1:1, and a mapping of a type n:m (n concepts from one meta-model are mapped with m concepts from the other).

After defining the mapping, the concepts are integrated in a single meta-model using both ArchiMate's extension mechanisms: the addition of attribute and the specialization (Josey et al., 2016). Concretely, if no mapping is detected, the

Table 7.4 Relevant ArchiMate symbols

Modelling concept	Definition	Symbol
Value	Value represents the relative worth, utility, or importance of a core element or an outcome. Value may apply to what a party gets by selling or making available some product or service, or it may apply to what a party gets by buying or obtaining access to it. Value is often expressed in terms of money, but it has long since been recognized that non-monetary value is also essential to business.	Value
Meaning	The knowledge or expertise present in, or the interpretation given to, a core element in a particular context	Meaning
Assessment	An assessment represents the result of an analysis of the state of affairs of the enterprise with respect to some driver.	Assessment
Business function	A business function is a collection of business behavior based on a chosen set of criteria (typically required business resources and/or competencies), closely aligned to an organization, but not necessarily explicitly governed by the organization	Business function
Business process	A business process represents a sequence of business behaviors that achieves a specific outcome such as a defined set of products or business services	Business process
Business actor	A business actor is a business entity that is capable of performing behavior	Business actor
Resource	A resource represents an asset owned or controlled by an individual or organization	Resource
Capability	A capability represents an ability that an active structure element, such as an organization, person, or system, possesses	Capability
Driver	An external or internal condition that motivates an organization to define its goals and implement the changes necessary to achieve them	Driver

concept from extension domain is added in the ArchiMate using the first extension mechanism which consists in adding attribute to an existing concept. If a 1:1 mapping exists without conflict between two concepts, both concepts are merged in a unique one, this concept is added into the integrated meta-model, and this concept keeps the name of the ArchiMate concept. If a mapping of type 1:1 with conflict exists between two concepts, this means that one concept from one meta-model is richer or poorer than a concept from the other meta-model and, in this case, both concepts are added in the integrated meta-model using the second extension mechanism of ArchiMate which is the stereotype (specialization).

7.3.9 ArchiMate Extension for Value Creation

In this chapter, the ArchiMate extension mechanism has been applied to the field of value creation. Table 7.5 explains the mapping between elements from the value creation and from the ArchiMate meta-models. Nine VC elements are mapped with ArchiMate elements, and only one VC element (i.e., the *value component*) has no corresponding ArchiMate element. The justification of this last case is that, although the *value component* from the VC meta-model could have been mapped to the *value* from the ArchiMate meta-model, we have preferred to keep the semantic difference among the elements of *value* and the *value component* from the VC meta-model in the ArchiMate meta-model. Note: the *value component* from the VC meta-model may be a type of *value*.

Accordingly, the integration rule that we have exploited to integrate the *value components* with the ArchiMate meta-model is the addition of attribute, and as a result, we have considered that the *value component* is an attribute of the *value*.

Another integration rule that we have used for one element is the merge, i.e., the concept of value from the VC meta-model has been merged with the concept of value from the ArchiMate meta-model because both concepts are defined more or less equivalently, respectively: as *the degree of worth that concerns something [which] improves the well-being of the beneficiary after it is delivered* (VC meta-model) and as *the relative worth, utility, or importance of a core element or an*

Table 7.5 Mapping between the VC elements and the ArchiMate elements

VC elements	**ArchiMate element**	**Mapping**	**Integration rule**	**Integrated element**
Value	Value	1-1	Merge	Value
Nature of the value	Meaning	1-1	Specialisation	<<Nature of the value>>
Value component	-	-	Addition of attribute	<<Value>>, Value component: description
Object	Business, Application and Technology layers	1-n	Generalisation	Business, Application and Technology layers
Measure	Assessment	1-1	Specialisation	<<Measure>>
Activity	Business function	1-1	Specialisation	<<Activity>>
Method	Business Process	1-1	Specialisation	<<Method>>
Stakeholder	Business actor	1-1	Specialisation	<<Stakeholder>>
Resource	Resource and Capability	1-2	Generalisation	Resource
Information system	Business, Application and Technology layers	1-n	Generalisation	Information system
Context	Driver	1-n	Generalisation	Context
Goal	Goal	1-n	Merge	Goal

outcome (ArchiMate meta-model). Four concepts from the VC meta-model also consist in specialization of concepts from ArchiMate: *nature of the value*, *measure*, *method*, and *stakeholder* are, respectively, specialization of *meaning*, *assessment*, *business function*, and *business actor*. For instance, the method is defined by *a property on which calculations can be made for determining the amount of value expected from a value creation method* at the VC meta-model level and by *the result of an analysis of the state of affairs of the enterprise with respect to some driver* at the ArchiMate meta-model level. The second definition is hence more general than the first.

Finally, four concepts from the VC meta-model consist in generalization of concepts from the ArchiMate meta-model: *object*, *resource*, *information system*, and *context* are generalizations of different elements from the *business*, *application*, and *technology* layers, as well as the *strategy* and *motivation* extension. According to ArchiMate semantics, the value creation concepts may be expressed using the ArchiMate symbols, as explained in Table 7.5, and modeled as represented in Fig. 7.9. In that figure, concepts between double angle quotes (e.g., ≪Context≫) are specializations of ArchiMate concepts, and concepts without guillemets are original concepts from ArchiMate (e.g., resource).

7.4 Value Co-creation Process Model and Language

In this figure, we investigate how VCC process may be considered as an instance of value creation (see Fig. 7.1). After this, in Sect. 7.4.2, we propose an extension of the ArchiMate language for value creation to express SPCC process.

7.4.1 From VC to VCC in KIBS

In our previous work, we have explained to what extent the value creation meta-model (Fig. 7.6) is suitable to model the processes of value co-creation in KIBS proposed in Lessard (2015) (see Fig. 7.10). To that end, we have considered one specificity of value co-creation which is that value is cocreated on the basis of a collaboration between many stakeholders who have different responsibilities during the co-creation, including the generation of the appropriate resources needed for co-creation activities. Consequently, a prerequisite before modeling the value co-creation process was to enrich the value creation model with the concepts of the stakeholder and the resources. This improvement was achieved by integrating the value creation model presented in Feltus and Proper (2017a) with the value model proposed in Li (2017) and Li et al. (2017). This chapter does not explain the integration but further details are available in Feltus et al. (2018b). The processes and generative mechanisms of value co-creation in KIBS engagements are illustrated in Fig. 7.10. These processes are, respectively (Lessard, 2015), *developing*

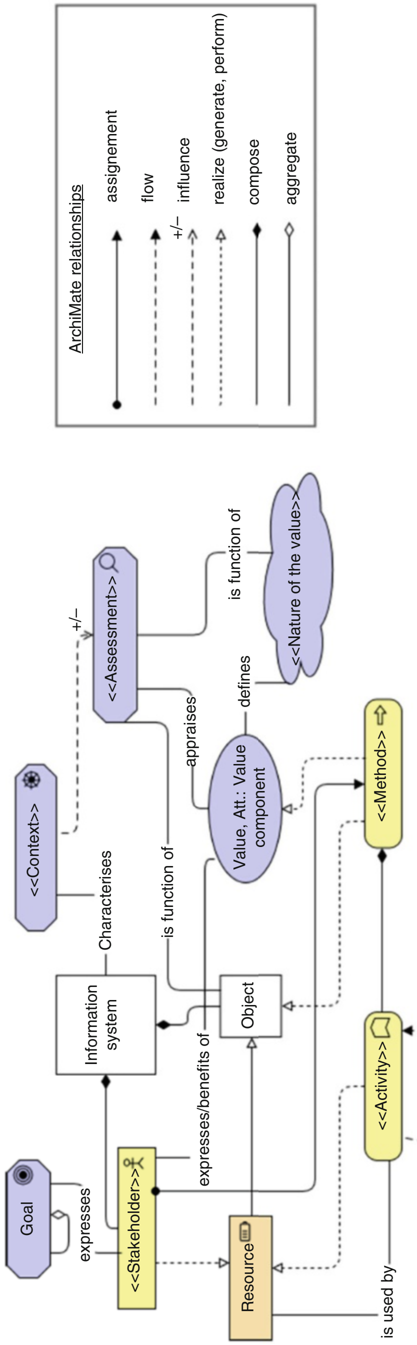


Fig. 7.9 ArchiMate extension to value creation

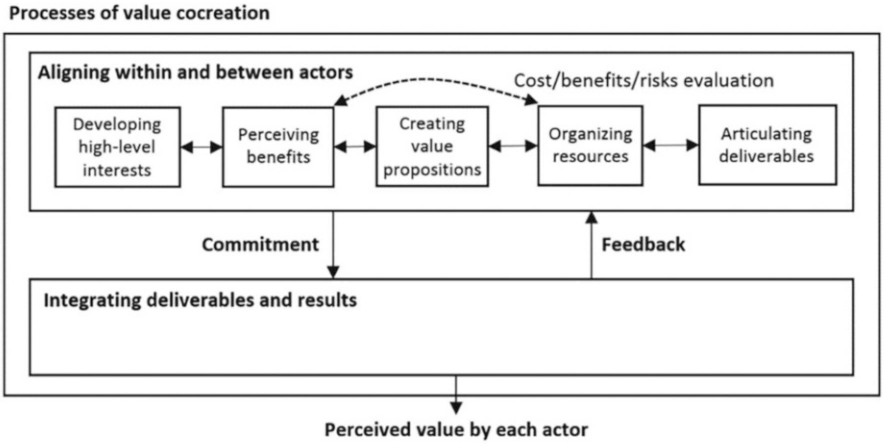


Fig. 7.10 Processes of value co-creation in KIBS engagements, adopted from Feltus et al. (2018b) ©2018 Springer-Verlag Berlin Heidelberg; reprinted with permission

high-level interest (concerns the generation of actor's motivation in co-creation), *perceiving benefits* (concerns the perception of the value proposition [in the context of service-dominant logic]), *creating value proposition* (concerns the process of adapting the proposition to the customer's need), *articulating deliverables* (concerns the definition of the output of the service engagement), and *organizing resources* (concerns the organization of the resources to create deliverables and to integrate them to the business activity). Only the processes dedicated to the alignment within and between actors are considered in the following.

The result of the specialization of the concepts of value creation into concepts of value co-creation and then into concepts of security and privacy co-creation process is the following:

- *Stakeholders*—They are the entities that perform the method that cocreates value, who benefit from this value, and who generate the resources used by the method activities. These stakeholders are of three types in the field of KIBS: companies, their customers, and partner organizations.
- *Information system*—This concept is not addressed in the processes of value co-creation (Lessard, 2015). However, it is relevant to consider it and to introduce it in the VCC to keep the specialization of the VC coherent.
- *Object*—In the value creation meta-model, like in the value co-creation meta-model, the object concerned by the value is the element from the information system that aims to be better off after that value is proposed and accepted.
- *Context*—The context of the creation of value is equivalent to the context of value co-creation.
- *Nature of the value*—The nature of the value defines the value generated by the creation or the co-creation.

- *Value component*—This concept expresses the different elements that constitute the value in the case of VC and VCC.
- *Measure*—In VC and VCC, the measure appraises the level of value created or cocreated.
- *Method*—The method corresponds approach followed to create or cocreate value (e.g., in Lessard, 2015, value is cocreated following a process-based approach. The first process related to the need for alignment among KIBS actors and the second concern the integration of the deliverables and results). The specialization of the VC to VCC only focuses on the first part and considers that the integration of the deliverables and results may be achieved similarly.
- *Activity*—To be achieved, the method is composed of activities that are articulated with each other. These activities are equivalent in each specialization of the VC meta-model, to know developing high-level interests, perceiving benefits, creating value propositions, organizing resources, and articulating deliverables.
- *Resource*—According to the definition, a resource is a type of object used by an activity. These resources are needed for the realization of the activities of the value creation process but also of the value co-creation processes.

7.4.2 ArchiMate Extension

Based on the specialization of the VC meta-model for KIBS, it is possible in turn to propose an ArchiMate extension to support value co-creation processes. Table 7.6 provides a summary of the VC concepts and of their specializations to VCC process concepts. Practically, the following concepts remain unchanged when they are specialized from VC to VCC, to know the following: *object*, *measure*, *method*, *stakeholder*, *resource*, *information system*, *context*, *goal*, *value*, *nature of the value*, and *value component*. The concept of method and activity is specialized, respectively, in *process of value co-creation* and *developing high-level interests*, *perceiving benefits*, *creating value propositions*, and *organizing resources articulating deliverables*.

Based on the mapping and the ArchiMate extension proposed in Table 7.6, Fig. 7.11 summarizes the value co-creation process language extension.

7.5 Case Study in the Smart Airport

This section illustrates, with a case study from the air transport sector, the ArchiMate extensions to express the value co-creation process.

Table 7.6 ArchiMate extension for the VCC process

VC elements	VCC elements	ArchiMate element	Integrated element
Value	Value	Value	Value, with the attribute value description
Nature of the value	Nature of the value	Meaning	<<Nature of the value>>
Value component	Value component	-	-
Object	Object	Business, Application and Technology layers	Business, Application and Technology layers
Measure	Measure	Assessment	<<Measure>>
Activity	Developing high-level interests, Perceiving benefits, Creating value propositions, Organizing resources articulating deliverables	Business function	<<Developing high-level interests>>, <<Perceiving benefits>>, <<Creating value propositions>>, <<Organizing resources articulating deliverables>>
Method	Process of Value cocreation	Business Process	<< Process of value cocreation>>
Stakeholder	Stakeholder	Business actor	<<Stakeholder>>
Resource	Resource	Resource and Capability	Resource
Information system	Information system	Business, Application and Technology layers	Information system
Context	Context	Driver	Context
Goal	Goal	Goal	Goal

7.5.1 Case Study Description

The case study for validating the VCC meta-model and language concerns the optimization of the passenger flow at a large European airport with around 25 million passengers per year. The information and background concerning this case study are collected from a public description of the Smart Airport Turnaround pilot that was developed as part of the European lighthouse initiative TransformingTransport[2] (Metzger et al., 2019).

[2] Smart Airport Turnaround Pilot Design, TransformingTransport Deliverable D8.1, March 2017; https://transformingtransport.eu/sites/default/files/2017-08/D8.1%20-%20Smart_Airport_Turnaround_Pilot_Design_v1.0.pdf.

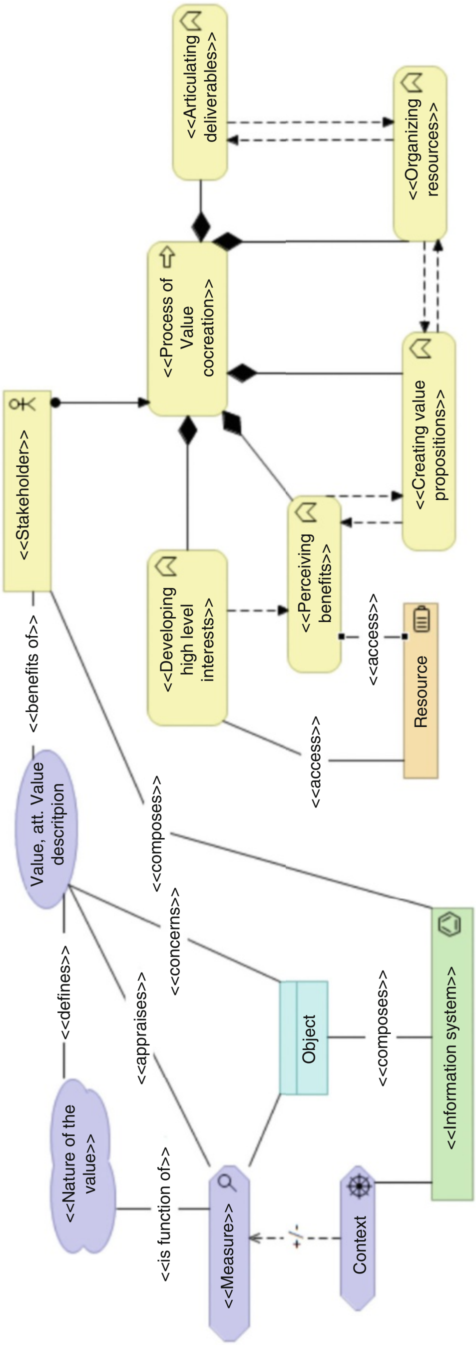


Fig. 7.11 Value co-creation process language (process-language) extension

The pilot involves a large European airline with around 14 Mio passengers per year, whose main hub is at the aforementioned airport, thereby facilitating the airport and the airline to share their data about passengers to jointly turn this available data into integrated intelligent information. At the airport and airline side, this allows significant savings in operational efficiency. Concretely, three main results were achieved: (1) decreased number of passenger losing the connecting flight, (2) facilitating a better scheduling of daily operation and resources required, and (3) enabling a better understanding of the impact of each process on the airport performance. These results potentially impact both the airport business model and the passenger's experience.

On the passenger side, the pilot aimed to improve the passenger travelling experience, which results mainly in less missed connections and decreased passenger waiting times.

The following functional requirements were addressed by the pilot:

- Req. 1: Predict time of passenger arrival to the terminal.
- Req. 2: Predict time of passenger arrival to the processing stations and their demand.
- Req. 3: Elaborate passenger movement heat maps based on demographics.
- Req. 4: Identify passenger movement models and patterns.
- Req. 5: Assess and predict time to reach the gate.
- Req. 6: Identify transfer passenger late arrivals.

Concretely, two main outcomes were delivered from the pilot to achieve these objective: an operation management predictive optimization module and a descriptive passenger system (Fig. 7.12), each of both being structured in sub-objectives. We focus in this chapter on one concrete sub-objective, which consists in reducing the delays in departure flights caused by late passengers. Due to the many possibilities, which can affect a passenger's transit in the airport, passengers may arrive at the boarding gate later than the scheduled boarding time. This delay implies a reduction of revenue for the parties involved, a reduction of the SLA expectations, and a negative perception of the airport and airline. To anticipate these flight delays, the airport together with the airline try to identify the passengers and their movement in the airport and to carry out preventive actions to facilitate passengers in reaching their gate on time. Practically, this sub-objective is achieved by realizing the above functional requirements 1, 2, 4, 5, and 6.

Figure 7.12, modeled with the e3value language proposed by Gordijn and Akkermans (2001), portrays the exchange of value between involved stakeholders (depicted by the links between the actors). The airport systems support the complete operations of the airport, including in particular arrival and departure control system (such as assignment of planes to gates), on-site check-in, baggage handling, and security control. The airline management systems support the activities of the airline companies in offering transport services to its customers and in particular support ticketing, online check-in, and passenger management. Both airport and airline systems are essential for supporting the execution of the air transport. As part of the aforementioned pilot, these systems continuously provide data

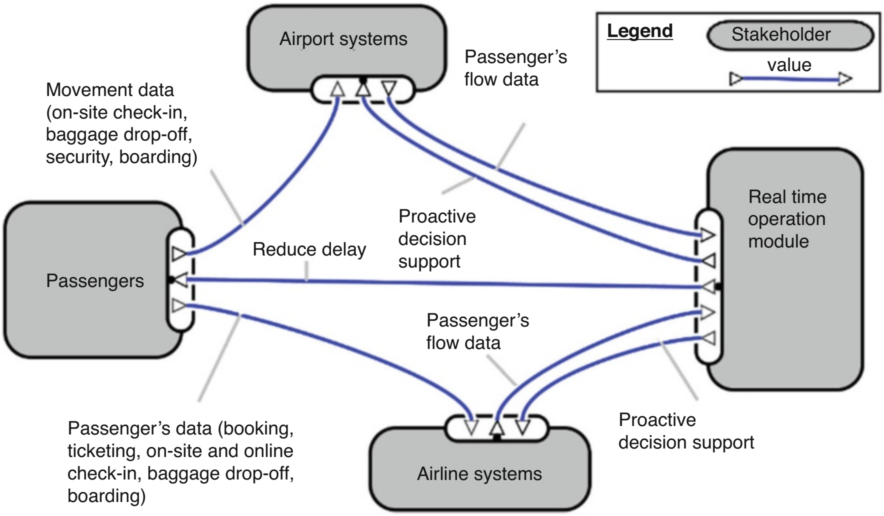


Fig. 7.12 Value co-creation case study summary—e3value model (Published before under nonexclusive copyright (https://fedcsis.org/for_authors/publication) in Feltus et al., 2018c)

to facilitate proactive decision-making based on the real context. The real-time operation module developed by the pilot provides this proactive support based on real-time information about passenger flow from the airport and airline systems. The operation module only uses anonymous passenger data to trace passenger flow, and does not use any personal-related information. Passenger movement is traced by means of when a passenger passed each of the different checkpoints (booking, ticketing, check-in, baggage drop-off, security, and boarding). Note, however, that this passenger movement is never matched with the personal data of passengers in the airline system.

7.5.2 *Value Co-creation Language*

As explained in the introduction of this chapter, the air transport case study illustrates the process of co-creation of value between the passengers and the airport/airline companies. During that co-creation, anonymized data about passenger position in the airport is shared with the airport/airline companies in order to improve their travelling experience. In turn, the airport and airline companies also exchange data among them in order (1) to improve the passengers' satisfaction and (2) to save operational efficiency and by the way improve financial return.

Figure 7.13 clearly highlights the advantages of using a language to express the value creation. The suggested ArchiMate extension for that matter offers the following facilities:

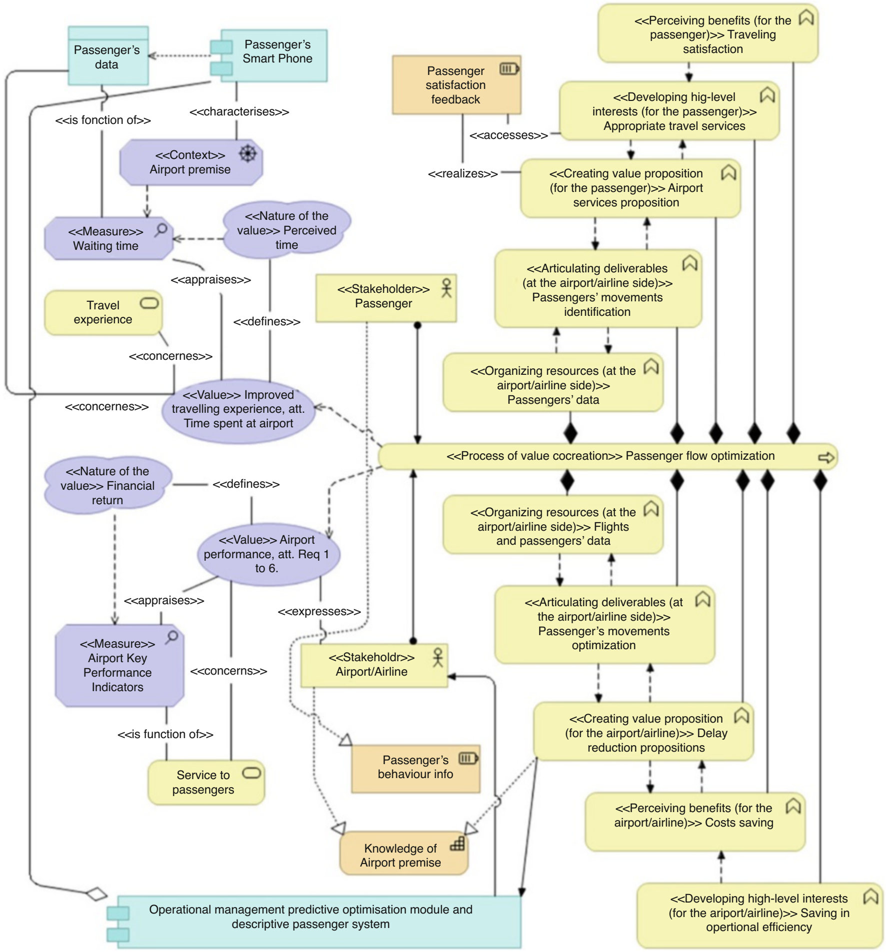


Fig. 7.13 Illustration of the VCC process language

- The elements expressed in the model are classified using a code of colors. That is, business concepts are in yellow, IS technical assets are in light blue, and value-related concepts are in purple. The latter are mainly specializations from the motivation extension of ArchiMate, which means that the co-creation of value is something that may be perceived in parallel to the information system but that motivates the design of the latter.
- Concept reading is facilitated using the shapes of the symbols. For instance, value elements are rapidly detectable on the model because they are oval. The nature of this value is also easily isolated because they are clouds, etc.

Using this ArchiMate extension to value creation thus may facilitate communication about the creation among all stakeholders (IT, business, operation, etc.) and to understand the relationships between the VC elements, e.g., the ≪*activities*≫ compose the ≪*method*≫, the ≪*stakeholder*≫ is assigned to the execution of the ≪*method*≫, etc. Moreover, ArchiMate allows us to benefit from the fact that concepts can have an explicitly attached meaning. For example, a task that realizes a capability is illustrated using a dot line with a white arrow, the flow between the activities is illustrated in dashed line with a black arrow, and the generic association is illustrated using a plain line. To improve the semantics of the latter, this association has also been specialized. For example, the ≪*measure*≫ ≪*appraises*≫ the ≪*value*≫. Finally, the diagram of Fig. 7.13 also allows clearly distinguishing the activities of value co-creation assigned to the passengers (upper part of the diagram) and assigned to the airport/airline companies (lower part). It is also worth noting that this case study has the particularity of combining performance (of the services offered to the passengers) and quality (of the travel experience offered by the airport/airline companies). Such a co-creation of value of different natures (performance and quality) is coherent with the semantic of the value concept (see Sect. 7.3.2).

7.6 Conclusion

The increase of competitions and the arising of new forms of communication between companies generated new types of collaboration during which providers and customers need to join their forces to cocreate value. Value co-creation is a concept developed by the marketing theory (e.g., Vargo and Lusch, 2004; Grönroos, 2008; Vargo and Lusch, 2008; Wille et al., 2013; Frow et al., 2015), which has progressively been integrated in the field of computer and service science (e.g., Weigand, 2009; Chew, 2016; Razo-Zapata et al., 2016; Gordijn et al., 2000; Feltus and Proper, 2017a). Although a plethora of research exists aiming at depicting the fundamental of VCC, few contributions have been poured so far in the area of language to support a method for VCC design and deployment. Nevertheless, such a language is necessary to describe and to visualize different components of the information system, as well as their underlying relationships and dependencies.

Acknowledging that, this chapter has proposed firstly to extend ArchiMate enterprise architecture language to express the creation of value following the value creation model proposed in Feltus and Proper (2017a). This extension was realized by exploiting the two ArchiMate extension mechanisms, to know the specialization and the addition of attributes. Secondly, the chapter has proposed an ArchiMate extension to express the process of value co-creation. Therefore, we have considered value co-creation as an instance of value creation and based on the observation that the process of value co-creation is a type of KIBS (Lessard, 2015). Finally, we have illustrated the designed ArchiMate extension with a case study from a smart airport.

This chapter also illustrated the value co-creation regarding two types of value: the *performance* of the airport and the quality of the travel experience. More recently, value co-creation has also been considered of relevance for the field of

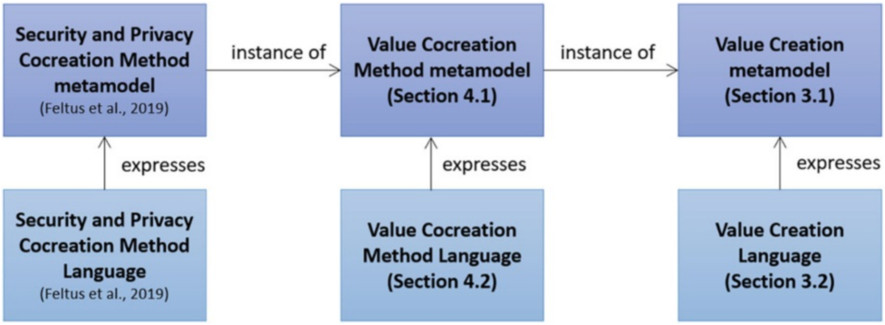


Fig. 7.14 Modeling approach and language extension design of SPCC

information security (e.g., Hawley et al., 2013; Vicini et al., 2016; Bennaceur et al., 2018). In that field *security* and *privacy*, when rightly deployed on the IS, also bring value for the enterprise. The reason is that security and privacy are two characteristics of elements of the information system that improve the stability and reliability of the latter. Furthermore, both security and privacy, according to Feltus et al. (2018b), are themselves defined by the following characteristics: availability, confidentiality, integrity, non-repudiation, etc. (for security) and anonymity, pseudonymity, access to resources, etc. (for privacy). Finally, just like all types of value, security and privacy are also created by dedicated methods (like risk assessment, cryptography, packet filtering, etc.). Based on that statement, we have also investigated how security and privacy co-creation process may be considered as an instance of value co-creation, itself being an instance of value creation (Fig. 7.14). Furthermore, considering security and privacy activities as similar to knowledge-intensive business services (KIBS), we will further specialize the process of value co-creation proposed in Lessard (2015) in a process of SPCC.

In that field of SPCC, some preliminary works have been achieved so far. Vicini et al. (2016) have highlighted that the challenge of security co-creation is twofold: first, to extract the value of the enormous amount of data available in distributed environment and, second, to improve the perception that these data are handled by a trusted system to store privacy-protected content. This challenge is especially important when end-users are directly engaged in the co-creation process (Prahalad & Ramaswamy, 2004). Hawley et al. (2013) show how it is possible to integrate practical co-creation processes into security and privacy by design methodologies and propose a methodology and guidelines to translate high-level requirements into verifiable low-level and technological ones. Bennaceur et al. (2018) address the support of collaborative security in the field of Internet of Things and explain how the collaborative security tends to exploit and to compose the capability of the connected device to protect assets from potential harm. The authors propose an approach supported by a dedicated tool to support the above composition using a combination of feature modeling and mediator synthesis. Hawley et al. (2013) stress the importance of the collaborative approach to security management in the area of air traffic management, due to the fact that operations and systems become increasingly integrated. Accordingly, they claim that for a successful collaborative approach, security managers need to adopt collaborative leadership skills and approaches. More recently, Garrido-Pelaz et al. (2016) propose a collaborative security approach through the perspective of information sharing which can help to develop early prevention mechanisms. Therefore, they exploit a model for sharing cybersecurity information between dependent organizations that are impacted by different cyberattacks. Finally, an ArchiMate-based language to express SPCC was proposed in Feltus (2019).

Acknowledgments The research was supported by the National Research Fund, Luxembourg (http://www.fnr.lu), and financed by the ValCoLa (VCC Language) project, as well as by the European Union's Horizon 2020 research and innovation program under grant agreement no. 731932 (TransformingTransport).

Chapter 8
Conclusion

Kazem Haki, Bas van Gils, and Hend*erik* A. Proper

This concluding chapter will briefly reflect on the contributions of the chapters in this part. As a first observation, it should be noted that several contributions in this part suggest that digital transformation is not so much about *technology*, but rather about the *people* driving/realizing the transformation effort, and have to work in the transformed enterprise. For example, Chap. 3 emphasizes the need for organizational clarity (what is the distribution of responsibilities among stakeholders involved in the transformation project), and Chap. 7 clarifies this further by emphasizing the need to use a well-defined modeling language to support digital transformation in light of value co-creation.

This puts the following point to the fore: digital transformation of enterprises should be considered in light of the fact that organizations increasingly use a co-creation model to create value for stakeholders. The modus operandi seems to be shifting away from a "linear" and traditional value chain model with a chain of organizations providing each other services until the "final" product is delivered to the end customer. In the new model, parties co-create from the start to create value—meaning that the distinction between consumers and producers is blurred. The idea is that all participants benefit from this more "tight" collaboration.

In that sense, this perspective can be seen as *utilitarian* as the focus has shifted from *value in exchange* to value in useṾalue in exchange means—as shown in,

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_8

e.g., van Gils et al. (2006)—that a is exchanged for b and actors are expected to be rational in their exchanges, meaning that they will only engage in the transaction if the value of b is perceived to at least be equal to that of a. By engaging in value co-creation, this up-front decision is changed: the actor will engage in an activity which hopefully will give benefits (and value) that will exceed the (value of the) invested resources such as time, money, and effort. This is a major shift in perspective and has big consequences for business models, strategy, and architectures of organizations. These effects require more study in the near future.

The third conclusion is threefold. First, the need for transparency—especially in value co-creation networks—is key. Second, new(er) technologies such as blockchain/smart ledgers can provide the required technological capabilities in this area. This is illustrated by Chap. 5. Third, we do not know what is coming, and an approach with "fast" and "slow" elements is required to make headway. The term "fast" refers to the experimentation mindset (Chap. 4) and the need for agility, whereas "slow" refers to the more traditional (engineering) approaches based on the idea of analyzing a domain, modeling its solution which is subsequently implemented.

Part II
The Need for a New Design Logic

Chapter 9
Introduction

Kazem Haki, Bas van Gils, and Henderik A. Proper

This introduction will briefly reflect on the need of a new design logic and then position the chapters within this part.

Over the past decades, theoretical discourses in the literature have changed remarkably in accordance to the changing role of information technologies (IT) and to the waves of interests in novel topics (Baskerville and Myers, 2009; Gill and Bhattacherjee, 2009). The first wave was concerned with whether the rapidly growing information technologies (in the 1980s and early 1990s) truly matter. As such, the very early discussions treat IT as "automated plumbing" that is used to automate existing operations and to increase the speed of communications (Zammuto et al., 2007). Scholars later theorize the impact of IT investments on productivity of IT (Melville et al., 2004) as well as on gaining competitive advantage in organizations (Mata et al., 1995). The second wave was raised along the emergence of enterprise systems that yields various theoretical discourses on enterprise integration. A manifestation of enterprise systems is the introduction of enterprise resource planning (ERP), supply chain management (SCM), customer relationship management (CRM), or business analytics (BA) systems (Shang and Seddon, 2002), which allows integration across historically distinct inter- and intraorganizational boundaries. The recent wave is concerned with digitalization and digital transformation, which discusses how digital technologies have changed the

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_9

fabric and organizing logics of organizations (Besson and Rowe, 2012; Yoo et al., 2012; Zammuto et al., 2007). This recent wave goes beyond merely discussing how the business and IT sides of organizations are inextricably intertwined; instead, it gives rise to the notion of digital technologies (vs. information technologies) (Yoo et al., 2010) and theorizes on digital artifacts (vs. IT artifacts). In this part of the book, our focus lies on new organizing logics due to particularities of the new wave, i.e., digital transformation. This part prudently promotes value co-creation, service orientation, and platform economy as the foundational logics in digital transformation.

This part is organized as follows:

- Chapter 10 discusses digital capabilities for digital transformation and maps the outlined digital capabilities to various archetypes of organizational logic.
- Chapter 11 gives rise to the socio-technical and multidisciplinary character of digital transformation and discusses how the social and technical aspects can be integrated to provide a more comprehensive view on digital transformation.

Chapter 10
Digital Transformation Requires a New Organizational Logic

Janne J. Korhonen

Abstract In this chapter, digital transformation is conceived as a successive disruption in a series of digital revolutions that exalt digitalization to a new institutional level and call for a respective new organizational logic that is in line with the increasing complexity of the environment. Putting forward a vertical typology of capabilities and an ordinal typology of organizational logics, it is argued that organizations that are adaptive in the face of today's turbulent digital environment exhibit co-adaptation logic along four aspects—embeddedness, association, awareness, and potentiality—at all levels of capability ranging from zero capabilities at the level of concrete work to adaptive capabilities at the level of interorganizational ecosystems. It is concluded that requisitely "digital" organizations are highly interdependent at all levels, the human intentions and values of all stakeholders are genuinely appreciated, and the future potential of the organization is both recognized and acted upon with the digital affordances. The conjectures shed light on why some organizations seem more capable in their digital transformation efforts than others and why most organizations fall short in their digital transformation.

10.1 Introduction

In the face of today's pervasive digital platforms, ubiquitous cloud computing, and advanced analytics technologies, traditional business models are faltering. As the fundamental shift brought about by liquefied digital data sweeps across all industries in every part of the world, most organizations are, to an extent, digital enterprises. However, there are vast differences in the way in which organizations are able to harness the affordances of digital data and technologies: while the leading digital platforms—the Amazons, Googles, and Apples—are "digital to the core," intertwining information technology in all aspects of their business and broader

J. J. Korhonen (✉)
Department of Industrial Engineering and Management, Aalto University School of Science, Helsinki, Finland
e-mail: janne.korhonen@aalto.fi

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_10

ecosystem, the majority of organizations are still learning the ropes of digital possibilities, and yet a long tail of laggards is just taking first steps in digitizing and automating physical records and processes.

Digital transformation is not only about decoupling of information from its physical container. It also goes beyond advanced technological tools and new ways of leveraging digital data. The digital transformation is, first and foremost, about the effect that digitalization has on organizing and operating business. The environment has become unpredictable, fast, even perilous. Consequently, the operating and distribution models, structures, and culture that have worked well before are suddenly rendering obsolete.

Over the past few years, notions of *digitalization*, *digital disruption*, and *digital transformation* have pervaded mainstream business language. Why now? After all, binary arithmetic dates all the way to Leibniz, and information and communication technologies have been the mainstay of business for decades. What has changed? What is going on?

In a recent article (Korhonen, 2016), we went for a lookout for an explanation: "is digitalization a mere hype or a moniker for a more fundamental systemic change that underlies the concurrent developments?." We concluded that the digital transformation currently being referred to is not just a new name for an old phenomenon. However, it seems to follow a common pattern that we have experienced time and time again: new technological innovations emerge, proliferate, and enable new ways of working and trading that were not possible before. What we now call digital transformation is but one disruption in a series of digital revolutions that every 15 years or so shake the business as usual.

Each disruption seems to have transpired, when technological solutions have developed, diffused, and enabled new ways of working, prompted by and contributing to the progressively more complex environment. Each time, the revolution seems to have exalted digitalization to a new organizational/institutional level, at which it has brought about an order of magnitude larger impact than before. Each such revolution also seems to have called for a new organizational logic that is in line with the complexity of the environment.

In this chapter, we will propose that the turbulent digital environment calls for an essentially new organizational logic. Under this logic, customer orientation and service thinking are not just noble goals. They are the starting point for the very organizing. "Digital" is not a separate concern. It pertains to the whole organization. Everyone must understand not only their own role in the digital capability but the importance of the work and knowledge of others in the whole. Everyone must appreciate, on a deeper level, the motivation and impact of their own work, the values guiding the work, and the values that the work promotes.

In this chapter, we present a tentative theory of the levels of organizational capability and apply it to the "digital capability" to provide an account of (1) what seems to make some organizations more capable in their digital transformation efforts than others and (2) why most organizations fall short in their digital transformation capability.

We would like to thank the Foundation for Economic Education in Finland for a travel grant that partly enabled writing this contribution on the other side of the world.

10.2 Levels of Capability

While it has long been suggested that capabilities form a hierarchy (Nelson & Winter, 1982; Collis, 1994; Winter, 2003), there is considerable “ambiguity around capability types and their competitive focus, propensity to change, and fundamental mechanisms of action” (Hine et al., 2013). In earlier work, we have earlier identified and described six levels of capabilities: (1) zero capabilities, (2) routine capabilities, (3) systemic capabilities, (4) creative capabilities, (5) strategic capabilities, and (6) adaptive capabilities (Korhonen & Halén, 2017). Leveraging the underlying metatheory of levels of work (Jaques, 1998; Rowbottom & Billis, 1987; Hoebeke, 1994), we further extrapolate this typology by the seventh level of capability: (7) generative capabilities. The levels of capabilities are defined in Table 10.1 and described in more detail below.

10.2.1 Zero Capabilities

The lowest level in the capability hierarchy consists of *zero capabilities*. These are elementary, atomic activities in the organization that are so pedestrian that they do not provide even a short-term competitive advantage (Hine et al., 2013). They are

Table 10.1 Levels of capability

Level of Capability	Definition
7. Generative Capability	The ability to deliberately and decisively amplify and attenuate field forces to create desirable patterns in the context.
6. Adaptive Capability	The organization’s ability to enable quick responses to and proactive enactment of disruptive changes in its environment to sustain its viability and effectiveness in the shifting context.
5. Strategic Capability	The organization’s ability to use and dynamically adjust its creative capabilities vis-à-vis the changing environment.
4. Creative Capability	Organization-specific, dynamic bundles of systemic capabilities that underlie the organization’s strategic capability.
3. Systemic Capability	Dynamic sets of interlinked routines that constitute responsive and relatively independent systemic functional wholes, or work systems.
2. Routine Capability	Relatively static routines that allow some degree of situational latitude.
1. Zero Capability	Concrete and pre-specified elementary, atomic activities that underlie routine capabilities.

the minimum requirement for an acceptable level of business operations. In terms of complexity, zero capabilities are commensurate with concrete and pre-specified work at the lowest level of organization (cf. Rowbottom & Billis, 1987; Hoebeke, 1994).

10.2.2 Routine Capabilities

Next level up, *routine capabilities* are ordinary, first-level capabilities (Hine et al., 2013) that "reflect an ability to perform the basic functional activities" (Collis, 1994). These are static routines that the organization does at any given time given its stock of factors of production (Nelson & Winter, 1982). They are focused on everyday subsistence tasks of the organization using current resources (Hine et al., 2013). These capabilities have some impact on the competitiveness of the firm (Hine et al., 2013). The corresponding work pertains to routines that allow some degree of situational latitude (cf. Rowbottom & Billis, 1987) within minimal critical specifications (Hoebeke, 1994).

10.2.3 Systemic Capabilities

Systemic capabilities are dynamic functional capabilities (Hine et al., 2013) that pertain to developmental change (cf. Anderson & Ackerman Anderson, 2001) in the organization's activities: "repeated process or product innovations, manufacturing flexibility, responsiveness to market trends and short development cycles" (Collis, 1994). They determine the period-by-period augmentation or diminution of the organization's factors of production, which are fixed in short term (Nelson & Winter, 1982). These capabilities are focused on change and use less-patterned routines and more specialized resources than routine capabilities (cf. Hine et al., 2013). Systemic capabilities pertain to work systems that adjust to the varying needs (Rowbottom & Billis, 1987; Hoebeke, 1994).

10.2.4 Creative Capabilities

Creative capabilities are about organization-specific creative ability—"the more metaphysical strategic insights that enable firms to recognize the intrinsic value of other resources or to develop novel strategies before competitors" (Collis, 1994). These dynamic learning capabilities (Hine et al., 2013) operate to extend, modify, or create ordinary capabilities (cf. Winter, 2003). The concept is commensurate with notions such as distinctive competence (Hitt & Ireland, 1985; Snow & Hrebiniak, 1980) or core competence (Prahalad & Hamel, 1990). Creative capabilities require intuitive judgment to detect gaps in services and to compare known systems with one another (Gould, 1986).

10.2.5 Adaptive Capabilities

While strategic capabilities provide competitive advantage in moderately turbulent environments, highly turbulent environments call for adaptive capabilities that enable the organization to quickly respond to and effectuate change in its environment to ensure its effectiveness in the shifting context. This resilience does not only pertain to recovery, flexibility, or crisis preparedness but also to a capacity for continuous innovation (Hamel & Välikangas, 2003). In some cases, the inherent limits of the organization's resilience are reached, and it must completely transform to maintain its existence of function in a new stability domain (Gunderson & Holling, 2002).

Adaptive capabilities enable quick creation of new knowledge (cf. Eisenhardt & Martin, 2000) and improvised response to rapid, unpredictable, and novel events (cf. Pavlou & El Sawy, 2010). As high uncertainty, high-risk environments do not allow time to respond effectively, these capabilities often rely on the network (Villar & Miralles, 2014). At the organizational level, adaptive capabilities enable strategies that are more personal, less linear, more ad hoc, and improvisational (Weick, 1998).

10.2.6 Generative Capabilities

Extrapolating from the previous levels of capabilities, generative capability, in this typology, is defined as the ability to deliberately and decisively amplify and attenuate field forces to create desirable patterns in the context. The way to go about rapid change that does not allow time for thought-out response is to act quickly and decisively, to sense the impact of the intervention, and to respond accordingly (Kurtz & Snowden, 2003). Generative capabilities open up new possibilities and create the conditions for innovation (cf. Kurtz & Snowden, 2003).

10.3 Aspects of Capability

We would further conjecture that capabilities are underpinned by a number of aspects that are manifested across levels. A plausible basis for such aspects might be found in the fundamental parameters of purposeful choice (Ackoff & Emery, 1972) and the respective adaptive ideals (Emery, 1977):

- *The probability of choices*—Other things being equal, the probability of choosing one course of action rather than some other, because it seems more fitting to oneself or one's idea of himself. This parameter is increased through the pursuit of homonomy: a sense of relatedness and belongingness between self and others.

- *The probable effectiveness of choices*—Other things being equal, the probability of choosing one course of action rather than some other, because it seems more effective according to the knowledge possessed or valued by oneself. This parameter is increased through the pursuit of nurturance: cultivation and growth of one's own competencies and those of others.
- *The probable outcome of choices*—This dimension is a function of the two first dimensions. It is increased through the pursuit of humanity: fostering the well-being of people, society, and ecology.
- *The relative value of intention leading to choices*—Other things being equal, the probability of choosing one course of action rather than some other, because of the underlying intentions between the actions. An ideal seeking system pursues beauty, continually expanding horizons of desire, yet the desire for outcomes is relative.

According to Emery and Trist (1973), maladaptations (improper adaptations) stem from artificial attempts to reduce the anxiety of making choice, when making choices becomes too difficult and too anxiety-laden. By contrast, adaptive strategies in the face of uncertainty rely on values and ideals (Emery, 1977).

Building on this foundation of (mal)adaptive dimensions, the respective four aspects of capability can be discerned: (1) embeddedness, (2) association, (3) awareness, and (4) potentiality. These aspects are summarized in Table 10.2 and discussed in the subsections below.

10.3.1 Embeddedness

Embeddedness would refer to the quality of attention that is paid to contextual organizational concerns: are they seen as separate considerations or intrinsically intertwined with the focal concern? The aspect is about meaningful relating to others, increasing scope of boundaries, and progressive elimination of separations such as functional silos, division of labor, or separation between planning and execution.

Table 10.2 Aspects of organizational capability

Aspect	Description
Embeddedness	The quality of attention that is paid to contextual organizational concerns.
Association	The quality of engaging and interrelating with stakeholders.
Awareness	The extent to which underlying intentions and values are recognized and interconnected.
Potentiality	The extent to which the future potential is recognized and enacted.

10.3.2 Association

Association would refer to the quality of engaging and interrelating with stakeholders. It is about increasing understanding of one's role and those of others in the larger context as well as appreciation of knowledge, skills, and work of others. It is about capacity for understanding and taking on other perspectives.

10.3.3 Awareness

Awareness would refer to the extent to which underlying intentions and values are recognized and interconnected. It is about taking into account the intentionality of humans and consequences of actions to individuals.

10.3.4 Potentiality

Potentiality would refer to the extent to which the future potential is recognized and enacted. It is about developmental motivation as well as understanding of strengths and limitations.

10.4 Organizational Logics

Organizational logics are "sense-making frames that provide understandings of what is legitimate, reasonable, and effective in a given context" (Guillén, 2001). They are historically constituted, rooted in shared understandings and cultural practice, and resilient in changing contexts (Biggart & Guillén, 1999). Organizational logics tend to be similar—"institutionally isomorphic"—in the shared embedding context (DiMaggio & Powell, 1983).

In the following, drawing on the literature, we put forward an ordinal typology of seven organizational logics, the "seven Cs"—(1) conformance logic, (2) competence logic, (3) cooperation logic, (4) coordination logic, (5) collaboration logic, (6) co-adaptation logic, and (7) concord logic. We also propose how the prominent organizational logic would have evolved through these archetypal logics historically over time.

10.4.1 Conformance Logic

The conformance logic is characterized by absolute belief in one right way, immutable laws, obedience to authority, order, stability, and predictability.[1] Problems are assumed to be well-structured (Joiner & Josephs, 2007) and beliefs are relatively unexamined and unjustified (King & Kitchener, 2004). Conflicts are avoided (Cook-Greuter, 2005; Torbert, 2004).

Under the conformance logic, the focus is typically on the quality of output (cf. van Vrekhem, 2015), excellence of task (cf. McMorland, 2005), financial stability, short-term profitability, and employee health (cf. Barrett, 2006).

It seems that this logic was manifested in the industrial betterment movement of the late nineteenth century. The focus of industrial betterment was on the physical health and comfort of individual workers rather than on the working conditions: "the path to profit, control, and industrial peace lay in bringing the workers' interests, values, and beliefs in line with those of the owner" (Barley & Kunda, 1992).

10.4.2 Competence Logic

The competence logic values efficiency (McMorland, 2005; Torbert, 2004) and expertise in one's craft (Torbert, 2004). As opposed to the conformance logic, this logic encourages independent thinking and does not eschew confrontation (Joiner & Josephs, 2007). Different perceptions are recognized (van Vrekhem, 2015), and simple abstractions and generalizations are used to think beyond the present moment and to imagine possibilities (Fowler et al., 2004).

The competence logic tends to be applied to define rules and procedures that standardize and formalize work, to monitor and diagnose this work (Macdonald et al., 2006), and to respond to each situation based on this diagnosis (Rowbottom & Billis, 1987).

This logic was arguably reflected in scientific management of the early twentieth century. Scientific management (e.g., Taylor, 1911) favored efficiency, division of work, and rationally optimized work processes.

10.4.3 Cooperation Logic

The cooperation logic emphasizes biologically motivated adaptation and change over the machine model of scientific management (cf. Barley and Kunda, 1992. Organizations are recognized as cooperative systems rather than products of

[1] Cf. the blue values meme (sensu Beck and Cowan, 2005) and "Conformist-Amber" paradigm (sensu Laloux, 2014).

mechanical engineering. The logic is characterized by effectiveness, mutuality, and proactiveness (cf. Torbert, 2004.

The cooperation logic is applied to systematically position products and services in a recognizable space (van Vrekhem, 2015) and to design and optimize the organization's work systems (Macdonald et al., 2006).

The cooperation logic seems to have underpinned the human relations movement of the 1920s through the 1950s that characterized organizations in terms of change, humanity, creativity, delegation of authority, employee autonomy, trust and openness, and interpersonal dynamics (cf. Barley and Kunda, 1992).

10.4.4 Coordination Logic

In the coordination logic, the perspective shifts from internal to external. It is characterized by adaptability, continuous learning, employee empowerment, and accountability (Barrett, 2006). It is attracted by difference and change more than by similarity and stability (cf. Torbert, 2004). The perspective is relativistic (Torbert, 2004) and has a contextual, subjective, and interpretive epistemology (cf. King and Kitchener, 2004).

In organizations, the coordination logic is typically employed in integrating and controlling the interactions between a number of systems (cf. Macdonald et al., 2006).

The coordination logic would have been the logic of systems rationalism of the 1950s through the 1970s that underlay approaches such as management science, operations research, process theory, and contingency theory (cf. Barley and Kunda, 1992). Systems rationalism made use of computer metaphors and "systems thinking."

10.4.5 Collaboration Logic

The collaboration logic is about mutuality and autonomy. It is characterized by trust, commitment, honesty, integrity, and enthusiasm that beget an increased capacity for collective action (Barrett, 2006). Strong interpersonal collaboration is emphasized and formal control is replaced by social control and self-discipline (Greiner, 1998).

Under this logic, the organization's identity is interdependently and coherently defined and manifested as a shared vision and shared values. This identity informs decision-making and is reflected in the organization's system and processes across the levels. As per this logic, the focus is no longer on just the focal organization and its well-being, but the whole "biotope" of the organization is consciously taken into consideration (cf. Glasl, 1997). Responsibility is shared in "shared destiny relationships" (Ghoshal et al., 1999), and there is awareness of "internal and environmental 2*nd and 3rd order effects*" (Jaques, 1998).

The collaboration logic appears to have underlain the organizational culture and quality discourse of the 1980s and 1990s. This rhetoric criticized systems rationalism for rewarding specialization, parochialism, and exclusive reliance on rational controls; instead, it advocated autonomy within shared beliefs and values (Barley & Kunda, 1992).

10.4.6 Co-adaptation Logic

As per the co-adaptation logic, external connectedness is expanded through building strategic alliances with like-minded partners, engaging with external stakeholders, and embracing environmental stewardship (Barrett, 2006).

The logic is prompted by hyper-competition (D'Aveni, 1994) and manifested through co-specialization (Teece et al., 1997). It is about real-time response to and proactive effectuation of change to ensure the organization's effectiveness in the shifting context—recovery, renewal, and continuous innovation. The logic stems from adaptation to the system of systems of other adaptive systems—akin to "co-jumping on a trampoline" (van Eijnatten, 2004).

Co-adaptation logic would underlie the sense-and-respond organization (sensu Haeckel, 1999), based on customer pull rather than production/marketing push. As per this logic, capabilities and resources are organized in a modular fashion and dispatched dynamically on demand, work is parallel and coordinated across the business network, and decision-making is decentralized (Haeckel, 1999).

10.4.7 Concord Logic

The concord logic is characterized by all-embracing scope in terms of time and space: consciousness about fundamental worldwide forces that drive changes in the environment (Macdonald et al., 2006), social responsibility, and consideration of future generations (Barrett, 2006). It is "unitive" and characterized by mindful, nonevaluative attention to the present moment, where the past and the future interpenetrate (cf. Cook-Greuter, 2005).

10.5 Concomitance of Levels and Logics

Spicer (2006) positions organizational logics on a spatial scale. Following his line of thinking, we suggest that each new level of capability would call for a progressively more advanced organizational logic, as exhibited in Table 10.3. The two typologies seem to share notably common characteristics between the levels of capability and the respective types of organizational logic.

Table 10.3 Requisite organizational logics

Level of Capability	Requisite Organizational Logic
Generative	Concord
Adaptive	Co-adaptation
Strategic	Collaboration
Creative	Coordination
Systemic	Co-operation
Routine	Competence
Zero	Conformance

Table 10.4 Recursive nature of capabilities

Level of Capability	Capabilities Coordinated	Resources	End Results
Generative Capability	Adaptive capabilities	Global economic sphere	New industries and institutions
Adaptive Capability	Strategic capabilities	Business ecosystem	Value co-creation
Strategic Capability	Creative capabilities	Full-scale business	Integrated value proposition
Creative Capability	Systemic capabilities	Strategic systems	Portfolio response
Systemic Capability	Routine capabilities	Practices, teams of teams	Systematic service provision
Routine Capability	Zero capabilities	Skills, processes, teams	Situational outcomes
Zero Capability	-	Menials, automata	Concrete outputs

Helfat and Peteraf (2003) define an organizational capability as "the ability of an organization to perform a coordinated set of tasks, utilizing organizational resources, for the purpose of achieving a particular end result." Table 10.4 shows how the definition could be recursively applied at all levels of capability: each level of capability would call for coordination of capabilities one level down.

Table 10.5 exhibits the posited relationship between organizational logics and levels of capability. Increasing environmental complexity requires response at a commensurable level of capability, which, in turn, requires corresponding logic across the levels.

As per the competence logic, routine capabilities are managed relatively independently of each other. These are functional activities that coordinate elementary activities (i.e., zero capabilities) one level down utilizing workflows, teamwork, and people's skills to achieve situational outcomes.

The cooperation logic enables systematic response, manifested by more dynamic systemic capabilities. To constitute these systemic capabilities, routine capabilities must be coordinated through the means of defined practices at the team of teams level. The end result is systematic service provision that can be dynamically adjusted to conceivable contingencies.

One level up, creative capabilities require coordination logic—creative bundling of systemic capabilities to unique distinctive, or core, competencies. This integration calls for formal strategic systems such as enterprise architecture, strategic

Table 10.5 The juxtaposition of organizational logics and levels of capability

Level of Capability					*Adaptive capabilities* are managed as inter-organizational capacity
				Strategic capabilities are managed as idiosyncratic, organization-specific capabilities	Strategic capabilities are coordinated (often across organizations) to constitute adaptive capabilities
			Creative capabilities are managed as discrete core competencies	Creative capabilities are coordinated to constitute strategic capabilities	--->
		Systemic capabilities are managed as relatively separate subsystems	Systemic capabilities are coordinated to constitute creative capabilities	--->	--->
	Routine capabilities are managed independently as functional activities	Routine capabilities are coordinated to constitute systemic capabilities	--->	--->	--->
	Competence	*Co-operation*	*Coordination*	*Collaboration*	*Co-adaptation*
	Organizational Logic				

portfolio management, governance frameworks, etc. to enable comprehensive provision (Rowbottom & Billis, 1987), i.e., a portfolio response.

Strategic capability denotes an organizational closure: an organization-specific ability to coordinate its creative capabilities in line with its unique value proposition. This integration follows the collaboration logic based on vision, values, and other means of social cohesion, rather than formal systems.

Adaptive capabilities often rely on a network of organizations, whose strategic capabilities are choreographed to constitute an interorganizational adaptive capability. The integration of inter-independent organizations calls for the co-adaptation logic.

It should be noted that generative capabilities are excluded from this analysis.

10.6 Implications of Co-adaptation Logic on Digital Capability

If "digital transformation" is defined as the latest revolution in the series of digital disruption, it denotes—in the light of the above discussion—transformation of organizational logic from collaboration to co-adaptation logic. This logic will be manifested at different levels of work and along different aspects of capability, as shown in Table 10.6.

As per the co-adaptation logic, digital data and technologies are embedded in the organization's modus operandi across all levels of capability. The entire digital data and content life cycle from data intake to analysis and reporting are integrally embedded in the organization's zero capabilities (i.e., elementary activities). They are integrated to routine capabilities (manifested as business functions and processes) that are thereby data-infused and analytics-informed. They constitute systemic capabilities (e.g., business services) that are continuously improved, based on the analytics-driven feedback loop. The systemic capabilities

Table 10.6 Digital capability across levels under co-adaptation organizational logic

	Embeddedness	***Association***	***Awareness***	***Potential***
Adaptive	Digital platform business model	Creating Shared Value	Shared destiny relationships	Business portfolio management
Strategic	Digital business model	Collaborative business ecosystem design	Collaborative conversation to enhance shared understanding	Business model transformation
Creative	Integrated enterprise model and digital business service portfolio management	Open Innovation	Experiencification	Integration of foresight to strategic and innovation management
Systemic	Continuous closed-loop service improvement	Co-specialization	Service co-design	Future insights are integrated into work practices
Routine	Data-infused and analytics-informed business processes	Business Process Interdependence	Accumulation and synthesis of stakeholder needs and impact	Foresight integration from multiple sources
Zero	Embedded, integrated data flow and processing	Technical interaction infrastructure	Constant contact and liaison with customers and other stakeholders	Proactive scanning of weak signals

are managed as a coherent portfolio through an integrated model that gives rise to creative capabilities (e.g., innovation). These creative capabilities are applied within the bounds of the business model that builds on organization-specific strategic capabilities. Finally, the organization's digital business model is intertwined with those of other organizations at the level of digital platforms. The resilience, renewal, and co-evolution of these platforms are governed by adaptive capabilities.

The co-adaptation logic calls for deep collaboration, engagement, and interrelation between constituents at all levels. At the most concrete level, a shared infrastructure must be in place to allow technical interconnectivity and interoperability. Next level up, business processes must be integrated across the business network. On one hand, end-to-end business processes between long-term strategic partners may be tightly and idiosyncratically coupled, optimizing for efficiency. On the other hand, standardized B2B processes enable loosely coupled plug-and-play type of ad hoc connectivity to a larger pool of collaboration partners (e.g., bulk component vendors). Co-specialization would exemplify association at the systemic level: the organization produces a product or provides a service as part of a larger total solution provided by the network as a whole. This requires a good mutual understanding of respective roles in the ecosystem. Similarly, innovation is open, i.e., distributed across organizational boundaries, which requires mutual trust, clear and frequent communication, high degree of commitment, shared decision-making, and clarity into roles and responsibilities. On the business model level, association is evinced in collaborative business ecosystem design, wherein the value propositions of partnering organizations are aligned with each other and the market. This design sets an overarching context for open innovation and co-specialized offerings. At the adaptive level, association deals with establishing the shared vision and values. From the focal organization's perspective, this requires extensive proactive and preemptive global networking and interaction with the external environment.

Along the awareness aspect, appreciation of customers' and other stakeholders' needs, intentions and values, as well as the impact of the organization's products and services on these stakeholders is formed through constant, personalized contact (zero level) and accumulation and synthesis of the profusion of perspectives (routine level). At the systemic level, the experiences, ideas, and expectations of service beneficiaries are brought into service design through their active engagement in the design process. At the creative level, awareness is reflected, for instance, in the phenomenon of experiencification (Normann, 2001): the starting point for innovation is the way how customers experience and give meaning and purpose to these experiences. At the strategic level, the awareness aspect plays a role in collaborative conversations and strategizing (cf. Jarzabkowski, 2005) that enhance shared understanding across the ecosystem. At the adaptive level, awareness is exhibited in shared destiny relationships between stakeholders (Ghoshal et al., 1999).

From the digital capability perspective, the potential aspect would largely pertain to how digital data and technologies are harnessed to create foresight and to turn this foresight into new value. At the lowest level, this entails proactive scanning of weak signals, e.g., through surveillance. At the next level up, the data collected from multiple sources are integrated into actionable foresight information. This information is then integrated into work practices at the systemic level and to strategic and innovation management at the creative level. The strategic level is about business model transformation based on the future insights: where is the external environment evolving to and what capabilities will need to be developed or obtained to address the likeliest contingencies in the coming years. These transformations, in turn, are typically informed by business portfolio management.

10.7 Conclusion

In this chapter, we have proposed that the much-hyped "digital transformation" refers to the latest disruption, i.e., *collaborative revolution* (Korhonen, 2016), in a series of digital revolutions, each of which seems to have exalted digitalization to a new organizational/institutional level and required a new organizational logic, which is better in line with the increasingly turbulent and complex environment.

Putting forward a vertical typology of capabilities and an ordinal typology of organizational logics, we argue that organizations that are adaptive in the face of today's turbulent digital environment exhibit digital capability that embraces the *co-adaptation* logic at all levels of capability ranging from *zero capabilities* at the level of concrete work to *adaptive capabilities* at the level of interorganizational ecosystems. By contrast, organizations that fall short of the co-adaptation logic would resort to maladaptive responses that are not requisite in the environment.

We also argued that any capability can be conceptualized in terms of four fundamental aspects: *embeddedness*, *association*, *awareness*, and *potential*. Applied to digital capability, "digital" is deeply intertwined in the fabric of the organization, organizations are highly interdependent at all levels, the human intentions and values of all stakeholders are genuinely appreciated, and the future potential of the organization is both recognized and acted upon with the digital affordances.

The conceptual contribution of this chapter is intended to lay a foundation for practicable diagnostic instruments and theoretically informed organizational interventions.

Chapter 11
Organizational Identity and Self-Awareness: Creating Convergence Between Enterprise Engineering and Organizational Design

Rodrigo Magalhães, José Tribolet, and Marielba Zacarias

Abstract In this chapter, a proposal is made for framework aimed at bringing about a convergence between the concepts of organizational self-awareness (OSA) and organizational identity, as part of a much needed integration effort between enterprise engineering and a situated interpretation of organizational design. Such a proposal is presented with human-centered design as the intellectual background. As aforementioned, one aim of the proposal is to identify potential research topics. The integrated framework fosters multidisciplinary studies that take into account both identity and organizational awareness issues. In particular, the framework is intended at promoting studies aimed at assessing the effect that OSA tools, methodologies, or models have on organizational identity. Conversely, it also aims at informing studies that develop forms of organizational self-awareness, based on theories of organizational identity.

11.1 Introduction

The theme of "convergence" or "integration" between artifacts that most people would call organizational (e.g., mission statements, organizational charts, or operating procedures) and artifacts which derive from information technology (e.g., information architectures, process models, or management control software) is

R. Magalhães (✉)
INESC, Lisbon, Portugal
e-mail: rodrigo.magalhaes@inov.pt; rmagellan@netcabo.pt

J. Tribolet
Instituto Superior Tecnico, University of Lisbon, Lisbon, Portugal
e-mail: jose.tribolet@inesc.pt

M. Zacarias
Algarve University, Faro, Portugal
e-mail: mzacaria@ualg.pt

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_11

central to the discipline of information systems and has been widely explored in our own writing (Magalhães & Tribolet, 2009; Magalhães & Silva, 2009; Zacarias et al., 2007, 2010; Magalhães, 2014). One of the key achievements of this work has been a progressive narrowing of the gap between the disciplines of organization studies and computer science, both understood in the broadest possible sense.

The recent editorial for the journal of *Organizational Design and Enterprise Engineering* (Magalhães & Proper, 2017) is evidence of such accomplishment. However, despite the achieved progress, most of the papers purporting to contribute toward integration are still very general and difficult to pin down, in terms of practical implementation.

One of the most difficult challenges to overcome has been the lack of a common body of knowledge that both disciplines can use in addressing the problems, which in the main are also common. The impact of digital transformation of enterprises entailing the integration of information technologies in most if not all business areas goes beyond technical and procedural changes in business operations. They also require a cultural change enabling people to embrace their implications, risks, and continuous exploration of opportunities. Digital transformation and organizational change are concerns shared by both disciplines. However, they have addressed it separately and with different approaches and focuses. We posit that the only way to achieve the benefits of truly interdisciplinary works is through the integration of such disjoint knowledge bases.

The literature on information systems, on the one hand, has been dominated by the social sciences, and many of the papers found in top-ranking journals are indigestible by the computer science community. On the other hand, the papers produced by engineering-oriented researchers mostly lack an epistemological basis and are guided by "proof-of-concept" designs. In the search for a platform which might be palatable to the engineering community, Dietz et al. (2013) put forward a vision for the discipline of enterprise engineering (EE) with great success, judging by the growth, in terms of papers and participants, of its key annual event—the *Enterprise Engineering Working Conference*. From a practical standpoint, EE success has been reflected through the emergence and development of subfields such as enterprise architecture (EA) and business process management (BPM).

In the traditional social science-oriented approaches to organizational design, the word "design" is often used as a metaphor, usually with a meaning equivalent to "structure" or "configuration." However, a number of writers taking a more design-oriented perspective have emphasized that organizational design (a) is a holistic phenomenon or a "gestalt" (Yoo et al., 2006) and (b) is not a static configuration but a never-ending process of designing (Boland et al., 2008; Ciborra, 1996; Dunbar et al., 2008; Garud et al., 2008) and (c) is driven by not only technical-structural rules but also generative ones (Garud et al., 2006; Romme & Endenburg, 2006) and (d) has the ability to shape and even create new environments, rather than being determined by the environment (Sarasvathy et al., 2008). Adding to such situated perspectives, Magalhães (2011) has suggested that organizational design (as a noun) can be understood as a cognitive interface between an organization and its environment, an interface made up of perceptions, including perceptions of the organization's identity as well as perceptions about the state of the organization's architecture as collectively observed by the organization's stakeholders. Architecture is defined as the "fundamental organization of a system embodied in its components, their relationships to each other and to the environment, and the principles guiding its design and evolution" (Maier et al., 2001, p. 108).

Thus, one of the pillars in this conceptualization of organizational design is the notion of organizational identity, a concept with very close links to the concept of organizational self-awareness (OSA) put forward by Tribolet (2005) and further developed by Zacarias et al. (2007); Magalhães et al. (2007, 2008); Aveiro et al. (2010) and Zacarias et al. (2010). As part of the concerns of enterprise engineering, OSA emerges as one of the consequences of applications of information technology to organizations and especially the applications enhancing the sensing and responding capabilities of organizational members. OSA is part of an actor-centered approach to organizations, dealing with what individual actors know about their own work, as well as the work of others, and it also deals with what resources are required or available, and, importantly, it deals with the contribution of each individual actor to the organization as a whole. Similarly, OSA evolves from being an individual capability to becoming an organizational capability and concerns the decision-making that all individuals in organizations have to undertake, in order to steer their daily activity. At any time, individuals model reality around them, search for the best possible course of action in their own contexts, take action, observe the outcome, and adjust to their renewed contexts.

According to the OSA approach, organizations are the result of the mental models of organizational members and their state of awareness regarding organizational experiences and events. With technology, such awareness tends to be acquired more and more quickly, in relation to actual events, leading to near-real-time (or even real-time) awareness. This has clearly important consequences for the design and the performance of organizations. At the collective level, OSA is said to be the key to the systemic governance of organizations, i.e., the more knowledge about the whole, the better will be the steering. However, at this level, the OSA concept is more problematic, given that individual actors do not always agree on objectives nor on the methods to reach such objectives. Organizational politics plays a crucial role in this, and whether we like it or not, consensus is very difficult to achieve, when we talk of "steering" the organization. Thus, in order to achieve the level of synchronization of mental models required for a true shared awareness at the organizational level (*active sync*), we need to factor in a number of intangible elements, such as values, emotional states, or motivation, which contribute decisively toward the achievement of human-centered *active sync*. Such elements might be summed up under the well-known concepts of organizational identity and identification.

Elsewhere, we have proposed that the human-centered approach to design (Krippendorff, 2006) can be usefully applied as the epistemological underpinning of organizational design and that in such application, organizational identity plays a central role (Magalhães, 2020). In this chapter, human-centered design (HCD) is proposed as an important complement to the engineering-oriented aspects of the design discipline, which underpin enterprise engineering. HCD originates from Krippendorff (1989, 's) definition of design as "*making something, distinguishing it by a sign, giving it significance, designating its relation to other things*" (Krippendorff, 1989, p. 9, emphasis added). This conception of design is distinguished from Simon (1969, 's) conception in the following way:

> Simon shifted our attention from the ontology of the natural to the logic of artificial, but failed to see that his very "project" dissolved material products into our dynamic relationships with them. Meanings too are made. They are not inherent in artifacts, cannot be attached to their surfaces, nor inscribed in static symbolisms. Nor are they derivable from ergonomics or the kind of cognitive science that goes for computational-logic accounts of operations on stimuli. Meanings, conceptions, and practices are invented and brought into play by people needing to cope with particular artifacts or achieve something with them (Krippendorff, 2011, p. 414)

Thus, it is submitted that HCD provides an essential tool for a better understanding of how OSA and organizational identity can be integrated and how the shared awareness resulting from a synchronization of the mental models of all actors involved is essentially dialogue- and identity-based. It is further submitted that the OSA concept provides an important enhancement to the notion of organizational identity and that both are powerful shapers of the design of organizations. From this, it follows that the two concepts—OSA and organizational identity—form a vital bridge, and a potential source of research themes, between EE and a situated interpretation of organizational design. Figure 11.1 shows the complementarity of the two concepts.

In the next section, we provide an overview of the key concepts of EE, including the notion of "enacted organization," a notion that leads to organizational self-awareness. Then, we move to a description of the human-centered approach and highlight how it serves as an epistemological underpinning of organizational design, with organizational identity as the backbone. Next, we discuss organizational self-

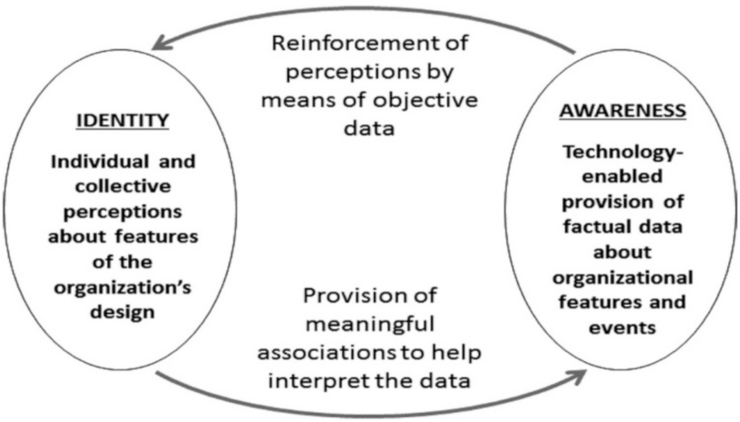


Fig. 11.1 Complementarity of identity and awareness

awareness, explain the similarities and differences between OSA and organizational identity, and propose an integrated framework as a future bridging tool between enterprise engineering and organizational design.

11.2 Enterprise Engineering

Engineering is the result of mankind's constant pursuit of adapting and even controlling the surrounding environment through the design, construction, operation, maintenance, and, eventually, disassembly of artifacts or systems. The range in nature and complexity of the artifacts dealt with engineering is very wide, from single artifacts to highly complex systems (Rouse, 2005b). Conceptualizing organizations as systems led to the idea that they could be properly analyzed, designed, and managed using engineering principles and methodologies and to the emergence of the enterprise engineering.

11.2.1 The Early Days (Liles et al., 1995; Liles & Presley, 1996)

A proposal for an *enterprise engineering* (EE) discipline emerged a couple of decades ago from an acknowledgement of the need of expanding the industrial engineering field to encompass more holistic views of the enterprise (Liles & Presley, 1996). The *Society of Enterprise Engineering* defined EE as "a body of knowledge, principles, and practices having to do with the analysis, design, implementation, and operation of the enterprise" (Liles et al., 1995). The EE worldview stemming from this definition is that the enterprise is a complex and adaptive system of processes that can be *engineered* to accomplish specific organizational objectives. But what does the phrase "to be engineered" mean? Liles et al. (1995) answer this question by proposing to apply to organizations the same engineering principles and practices used in dealing with other kinds of artifacts or systems. The main elements of engineering applied by EE are (1) *theory*, to set the stage and give focus to the development of principles and practices; (2) *abstraction or modeling*, as a way to represent the focus of study in a way that can be tested; (3) *design*, in order to iteratively generate alternative solutions to meet identified needs; and (4) *implementation*, where both users and designers evaluate design results and identify possible improvements. Another relevant aspect of enterprise engineering is the recognition in the ever-changing organic nature of the enterprise (Liles et al., 1995).

11.2.2 The Proposal from Dietz et al. (2013)

A more recent proposal for an enterprise engineering discipline has been put forward by Dietz et al. (2013) as (1) a means to address organizational design and (2) enabling a paradigm shift from Taylorist management approaches, which, according to the authors, continue to influence management functions today. Dietz and colleagues are strongly influenced by design science as conceived by Simon (1969) and adopted by the information systems discipline (Hevner et al., 2004; Hevner, 2007). Those authors interpret organizational design as the activity of devising corporate strategies with the aim of changing current enterprise situations into preferred ones. The authors consider design to be based on organizational learning as a means to cope with unexpected situations and posit that the responsibility for enterprise (re)design should be seen as distributed among all enterprise members.

Dietz and colleagues argue that EE has three main goals: (1) intellectual manageability, (2) organizational concinnity, and (3) social devotion. *Intellectual manageability* refers to the need for understanding enterprise phenomena, so that they can adequately address such phenomena. This means developing proper theories about the construction and operation of enterprises that provide insights concerning enterprises and enterprise changes and to master their complexities. *Organizational concinnity* refers to the need to operate as a unified and integrated whole, taking into account all aspects that are deemed relevant in order to perform optimally and to implement change successfully. The authors acknowledge that it is not sufficient to consider enterprise design domains like processes, resources, and software applications and their underlying infrastructure. They consider that all relevant aspects must be included, even those not presently foreseen, in a properly integrated way, so that the operational enterprise is always a coherent and consistent whole. *Social devotion* means that employees are fully empowered and competent for the tasks they have to perform, are endorsed with transparent authority, and have access to all information they need in order to perform their tasks in a responsible way.

As a means to achieve enterprise engineering main goals, the authors formulate seven EE fundamentals, of which the most relevant feature shifts the focus of organizational design from processes and resources to interaction patterns and commitments among actors called transactions. Transactions are considered the essential building blocks of enterprises. There is also a clear distinction between design and implementation, defined as the activities for putting a design into effect. Whereas implementation is considered mostly deterministic, design is regarded as a highly non-deterministic process because it entails unrestricted exploration of possibilities rather than following a pre-defined plan. These positions are problematic in terms of current research in organization studies, which shows that there can be no rigid distinction between design and implementation, given that these are mere subjective boundaries placed for the benefit of project management. In the reality of every organization, the stages known as design and implementation intermingle, and a situated conception of organizational design seems a truer picture of reality.

11.2.3 Enterprises as Dynamic Control Systems

The foundational concepts from dynamic systems and control (DSC) theories have been defined and used successfully in the engineering fields for decades (Kuo, 1995; Guerreiro, 2015). DSC is founded on three essential concepts: *dynamic system*, *control*, and *feedback*. A dynamic system is a system whose behavior changes over time, often in response to external stimulation or forcing. Broadly speaking, control refers to the mechanisms that regulate the behavior of dynamical systems, most commonly through feedback loops. The term feedback refers to a situation in which two (or more) dynamic systems are connected, such that each system influences the other and their dynamics are thus strongly coupled (Åström & Murray, 2008). Abraham et al. (2013b) conceptualize enterprises as systems and use the theory of hierarchical, multi-level systems to introduce three orthogonal dimensions of hierarchy (layers, strata, and echelons). They then position enterprise architecture management (EAM) as a cross-dimensional transformation support function in this three-dimensional hierarchy space. Control theory is used to derive a model of control and feedback loops that enables a designed EAM support of system-wide transformations.

11.2.4 The Enacted Organization as a Complex, Real-Time, Network of Actors, Acting, Controlling, and Designing

Zacarias et al. (2010) put forward the notion of enacted organization. Whereas according to the conventional conceptualization, the organization's design is represented from the perspective of its goals and strategies, structure, processes, or resources, the enacted organization is manifested through actions and interactions among its actors (human or automated, individual or collective). Moreover, a useful distinction is made between "enacted" and "prescribed" organization. In terms of its enactment, the organization is essentially the actions that its actors perform as part of their daily activity, and while the enacted organization is made visible through actors' actions and interactions, the prescribed organization is a picture of what the organization should be, in the minds of participants. In other words, while the enacted organization is made up of action, the prescribed (or "designed") organization is made up of conceptual and mental scenarios. Since there is always a misalignment between the organizational design and its realization or implementation, the enacted and prescribed organizations are never the same.

These authors put forward an actor-centric framework and propose an ontology aimed at facilitating the alignment between individual and collective views of the organization. Two key ideas are:

- Understanding and assessing the alignment the organization's design and the enacted organization requires developing means to inter-relate the actor-centric perspective and the models encompassed by the organization's design.
- Due to the complex behavior of actors and their multitasking capabilities, such interrelationships can only be properly understood within the contexts that emerge from the associated sequences of actions and interactions (conversations).

The constant misalignment between the intended and the enacted organization raises the importance of the notion of steering. Steering is the effort on the part of an observer to (1) evaluate how things are going by observing the result of their actions and (2) compare those results with actors' intentions. Based on deviations or innovations during the trajectory, actors review and refresh their purposes or goals, which are not stationary, but dynamic.

Drawing on Strube (2001) cognitive architectures, Zacarias et al. (2010) address the issue of actors' feedback loops and propose three behavioral layers to model organizational actors: (1) action, (2) deliberation, and (3) change/learn. The action layer represents intention (i.e., pre-defined behavior); the deliberation layer represents planning, decision-making, or scheduling behavior; and the change layer represents reflective behavior, i.e., the behavior that determines the evolution of the behaviors of the former two layers, based on performance measures attributed to by the involved actors. Whereas change takes place any time new behaviors arise, learning only occurs if performance is improved. Organizational actors continually switch among the three behavioral layers. Upon expected events, actors plan, decide, or schedule what to do, and once actors decide what to do, they act according to pre-defined behavioral patterns. Actors will modify their behavior in case of an unexpected event that cannot be dealt with reactive or deliberative behaviors.

This work feeds directly into the notion of organizational self-awareness (OSA), which will be discussed further along in this chapter.

11.2.5 A Comment on the Current State of Enterprise Engineering

The EE proposal put forward by Dietz et al. (2013) represents an important evolution, since it provides further developments regarding all four main engineering activities (theory, modeling, design, and implementation). Nonetheless, the most important distinction is the shift from a process-centered to an actor-centered approach. The actor perspective is highlighted not only as a main enterprise engineering goal (social devotion) but also through several of the EE fundamentals (focusing on transaction among actors, distributed operational and governance responsibility, and human-centered and knowledgeable management).

However, despite pointing research in the right direction, the current enterprise engineering body of knowledge does not provide sufficient tools for dealing with a complex, fast-changing, and uncertain environment where the boundaries between human and artificial actors and actions are becoming increasingly blurred. First and foremost, no current views, perspectives, or ontologies within the field of EE provide proper concepts to capture and look into the enacted organization. From our point of view, the most adequate approach to understand the enacted organization is a systemic network approach, where actors can be human or artificial. In the enacted organization, human actors represent actual individuals and groups rather than roles; nonetheless, the term "actor" continues to apply since the aim is to capture their behavior as members of the organization.

Control theory and dynamic systems have long been used as an analogy for organizations; however, what is new in our proposal is the usage of control theory and dynamic systems in combination with an actor-centered approach. Thus, we argue that not only the organization as a whole but also each actor in the network has to be regarded as a complex, dynamic system. To this end, we enrich the previous analogy by drawing upon control theory and dynamic systems. Hence, we regard both individual and organizational actors as dynamic systems with feedback loops. All actors have a model of their behavior. At each point in time, each actor is both acting according to their model and observing the results of its actions. In each cycle, expected events triggered as a result of their actions will cause actors to make corrections to the model parameters. Unexpected events will cause changes to the model, and disruptive events will result in the definition of a new model.

11.3 Human-Centered Design

In the search for a useful tool that might simultaneously stay within the bounds of a situated approach to organizational design and conform to the requirements of design as a discipline, we were struck by the conceptual proximity of the writings by Weick (1979, 1995) on enaction and the so-called human-centered approach (Krippendorff, 2006). Both authors are concerned with meaning, but from different perspectives.

While Weick is concerned with enaction as the main explanation for the way people behave in organizational settings, Krippendorff is driven by the meaning and significance behind the manufacturing of products or the rendering of services. Weick's concept of enaction suggests that perception is not just a property of the perceiver but is the result of an ongoing interaction between the perceiver and the perceived, which means that the perception of design in organizational contexts is a result of internally generated meanings (i.e., interpretations). The HCA, on the other hand, places meaning at the core of the design process and claims that as designers, "we need to be concerned with what the artifacts of design could possibly mean to users and interested parties, with the multiple rationalities that people can bring to bear on them" (Krippendorff, 2011, p. 413).

Giacomin (2014, p. 607) defines human-centered design as a "language which is absorbed and exchanged between people, providing the basic units of meaning" and suggests a useful model based on a hierarchy of human-centered design factors (see Sect. 11.3). At the bottom of the pyramid, the factors highlight the physical nature of the interaction of people with products, systems, or services, while the factor at the top—*meaning*—highlights the ambitions, desires, or wishes of the people using the products, systems, or services. The suggestion behind the model is that meaning "whether pre-existing or still to be created through contact, is considered to be the key to social acceptance, commercial success, brand identity and business identity" (Giacomin, 2014, p. 612). Considering that organizational design is an overall property that sums up the characteristics of the organization, it is apparent how meaning must be at the apex of organizational design, while activities, tasks, and human factors rest at the bottom. The model contains a series of questions which focus the attention on the different levels of the involvement of people with artifacts. As one moves further up the hierarchy, the answers to the questions not only provide a deeper understanding of the artifact but also offer a wider range of affordances that people may expect to get from the artifact.

In one of the best synopses of the HCA, Krippendorff (2011) provides a useful summary of the trajectory of design in the last 40 years, from industrial design to the design of discourses. From such a trajectory of artificiality, the evolution of the design discipline can be traced, from the design of products to the design of communicative and social artifacts, the type of artifacts that organizations are mostly made of. In the same article, Krippendorff proposes a set of nine design principles, put forward as the basis of the governance of contemporary design. The present paper approaches the first and fundamental principle—*Meaning is the Only Reality that Matters*—and extends it to organizational design, through the proposition that, in a fashion similar to meaning in the context of design in general, identity plays a central role in the shaping and maintenance of organizational design, acting as a *superordinate conveyer of meaning and purpose*.

In Sect. 11.3, it is shown how meaning (i.e., identity) is also at the top of a hierarchical view of organizational design and how the entirety of the organization's design shapes and is shaped by the messages communicated by the organization's identity. In the second layer, the model features the range of communicative artifacts that the organization requires to exist and to prosper, such as its mission and vision statements, its brand identity, its policies and procedures, and crucially its managerial discourse. The third layer answers the question of "when" and represents the organization in action, with its stakeholders interacting through its horizontal processes. The next layer stands for the traditional understanding of organizational design, i.e., the organizational structure, detailing the roles, functions, activities, and reporting lines for the execution of individual tasks. The bottom layer, arguably the most important, contains the stakeholders or the agents who turn the organization's design from intentions to performance. In such a transition, the meanings contained within the organization's identity are instilled into the practices of stakeholders, whether the interactions take place within the organization or between internal and external stakeholders.

The arrow going from the top to the bottom layer signifies that stakeholders are affected directly by the collective perception of identity and that when performing their tasks or interacting with other stakeholders, identity is being enacted, and work practices are being shaped and re-shaped. The small arrows running up and down signify the effect of identity from one layer to the next. For example, the meaning contained within the organization's brand is likely to affect interactions with customers through the call center. Likewise, the meaning contained within an individual's job description will most likely have an impact on the way the customer complaints process is operated.

11.4 Organizational Identity

Identity is defined as the shared perceptions of stakeholders about what is "central, distinctive, and continuous" in their organization (Albert & Whetten, 1985), a definition that does not stray far from Krippendorff's (Krippendorff, 1989) etymological explanation of the word design as "de + signare" and his definition of the verb design as "*making something, distinguishing it by a sign, giving it significance, designating its relation to other things*" (Krippendorff, 1989, p. 9, emphasis added). Looking closely at the two definitions, one finds that they are both underpinned by an intention to designate or create meaning.

Ashforth et al. (2008) explain organizational identity and identification as processes of formation of meaning working at both levels—collective and individual. It starts with identity being formed as an iterative process between the collective processes of sensebreaking and sensegiving and individual identity enactment. While sensegiving is about guiding the "meaning construction of others toward a preferred redefinition of organizational reality" (Gioia & Chittipeddi, 1991, p. 445), sensebreaking is about questioning of who one is when faced with some new organizational reality. In this manner, newcomers begin to learn the features of the organization's identity, in a recursive process that encompasses both the individual and the collective levels. Moreover, the term enactment embeds the notion of action as an integral part of perception and is taken to mean an act of "bringing forth meaning from a background of understanding" (Varela et al., 1991, p. 130). This, in turn, means that an organizational member's perceptions and understandings about the organization's identity are not limited to abstract verbal representations of feelings or beliefs, but they are also embodied perceptions.

From an embodied perspective, the construal of organizational identity is predicated on processing, examining, interpreting, and expressing emotional, visual, aural, bodily, or temporal information that can be formal and informal, official and unofficial, and symbolic and material (Harquail & King, 2010). On the other hand, given that embodied cognition is closely associated with emotions and the ethical sense of individuals (Colombetti & Torrance, 2009), it can be asserted that identity also has an emotional or motivational component. Indeed, such a component constitutes the individual's identification with the organization. Hence, identification can be defined as the motivational component of identity in the process of creating and being created by the emotional bonds that each individual participant forms with the organization, in the flow of organizational life. Such processes establish identity and identification as *superordinate conveyers of meaning and purpose*, shaping all organization artifacts and strongly influencing the formation and maintenance of communitarian principles, values, and codes of conduct.

The term "superordinate" has been utilized by Argote and Kane (2009) to explain that organizational identity acts as a governance and coordinating mechanism influencing knowledge creation and transfer in organizations. This is indeed supported by a number of distinguished economists, namely, Arrow (1974), Simon (1997), Akerlof and Kranton (2012), in flagging the issues of identity and identification as major sources of motivation, commitment, and organizational effectiveness. Brickson (2007) suggests that identity processes rest at the heart of the role of the firm in determining how organizations relate to their stakeholders, and Simon (1969) goes as far as saying that identification is the "principal reason for carrying out economic activities in organizations rather than markets" (Simon, 1969, p. 44). Kogut and Zander (1996) add a behavioral component, in stating that *the knowledge of the firm has an economic value over market transactions when identity leads to social knowledge that supports coordination and communication* (Kogut & Zander, 1996, p. 502). Thus, the superordinate status of organizational identity is linked to both its cognitive and motivational elements.

11.5 Cartography, Governance, and Organizational Self-Awareness (OSA)

Aveiro et al. (2010) define OSA as the continuous effort of minimizing the gap that exists between the understandings shared among all organizational members about the organization, the formal representations of those understandings, and the real and concrete organization. Considering that the "real," i.e., enacted, organization is made up of networks of actors, such synchronization requires integrating individual, partial, and frequently incoherent views of the organization self into a unique and shared view. Moreover, Aveiro et al. (2010) argue that since organizational reality is constantly changing, it is not only necessary to have a shared view of the organization as up-to-date and coherent as possible but also a shared record of the history of changes of the organizational self.

This idea has been little explored in EE since it is believed that the time delay between the perceived shared reality and the individual capability to access that shared reality is so long that it makes it no longer useful. However, it is worth noting that feedback speeds have been increasing non-stop in the last couple of decades, as can be exemplified by our GPS systems. These are the result of dynamic traffic systems that connect satellites, reports coming from helicopters, sensors in the road, telephone companies that monitor cell phones in the road, and emergency services reporting accidents. All this enables information to be relayed in real time, enabling us to make timely decisions regarding the best route to take. System-wide self-awareness and the capability to use it as part of the artifact are something that is routinely achieved in engineering. Likewise, in the rapidly changing organizational environments of today, we are moving toward a stage where it will be possible to achieve and maintain up-to-date and accurate organizational self-awareness, through a combination of structural arrangements and action, supported by properly designed conceptual, physical, manual, and automated artifacts (many of which to practically become part of human beings).

In a previous paper entitled "The Role of Enterprise Governance and Cartography in Enterprise Engineering," (Tribolet et al., 2014b) a general view of enterprise cartography (EC) was presented. EC deals with the dynamic design and production of architectural views that depict the components of an organization and their dependencies. It shares its constructs with enterprise architecture. However, its goal is purely descriptive. EC takes into consideration a view of enterprises as organized complexities, whose existence at any point in time is the result of the dynamic synergies between purposeful, intentional, goal-oriented designs, such as those generated by EA, with the emergent, opportunistic, individualistic, non-orchestrated, and often incoherent and chaotic actions that naturally occur as the result of the free will of the human actors that are, after all, the true core of any organization. We concluded this latter paper by connecting the dots between purposeful EA designs leading to intended pre-determined enterprise transformations and the realities of real-time events, with the emergent phenomena that in fact end up being the reality of the enacted enterprise as de facto is.

The key to connecting, integrating, and making sense of these two powerful reality drivers, one more top-down and the other clearly bottom-up, is precisely the capability to sustain, at the systemic organizational level, commonly shared representations of the AS-IS of perceived reality as accurate as possible. We have coined the concept of "organizational self-awareness (OSA)" to denote this systemic capability, whose fundamental technical basis is the notion of enterprise cartography (EC).

Providing to all involved—from the lowest level of operational workers to the top level of enterprise executives and strategists—a formal, updated, common view of reality is key to sustain the most essential feature of any organization: meaningful, informed, objective, explicit, non-ambiguous means of support to human-to-human communication. Achieving such capability entails (1) collecting the sensory data of the enacted reality as it happens and is perceived by those that are involved and immersed in the respective action contexts; (2) feeding such data into semantic organizational models capable of being read, understood, and interpreted by the organizational actors, according to their multiple and specific points of view; (3) providing the dialectic contradictory mechanisms to challenge, verify, and consolidate the sensory data being reported as true; and (4) systematically building a verifiable and verified explicit knowledge base of the enterprise reality and its dynamics on the fly.

This capability of an organization to be aware of itself at any point in time, in space and in context, is the equivalent to what we, single humans, do possess as most valuable to steer ourselves in life: our own self-awareness, on top of which we build higher-order levels of self, such as conscience, values, and ethics. EC is, at its very root, the basic mechanism for us to equip modern-day organizations with the artifacts we need to deal with the tremendous challenges we face today: exploding complexity, increasing speeds of interactions, gigantic volumes of information flows, shorter and shorter decision times, and increased requirement to build up on-the-fly agile and realizable answers to previously unknown challenges.

11.6 Proposing Convergence Between Organizational Identity and Self-Awareness

In order to move forward with the building of bridges between organizational design and enterprise engineering, it is essential to have a common understanding of what organizational design (OD) is. OD has a "hard" and a "soft" interpretation. According to the hard interpretation, OD is made up of the structures, policies, and procedures that form the skeleton of the organization. According to the soft interpretation, OD is the ensemble of the perceptions that stakeholders have about the visible as well as the invisible components of the structures, policies, and procedures. Therefore, it might be said that the soft interpretation of OD is the brain that rules over the entire body of the organization's design. While the soft interpretation relies mainly on meaning and identity as its analytical tools, the hard interpretation relies on raft of automated and non-automated tools, collectively known as organizational self-awareness, which provide feedback on the functioning of the skeleton. Hence, both identity and OSA are clearly complementary tools of OD, as shown in Fig. 11.1.

Developing OSA means having the means to capture and represent identity in a common language, as well as distributing such representations among relevant actors. Hence, OSA goes beyond a simply "sending" or "processing" of raw images, and it rather entails communication, negotiation, and agreements among actors on meanings of their images of reality. In other words, OSA also plays an important role in the constitution of OD.

The proposals made in this section are based on the premise that organizations are designed at different levels, some in the traditional realms of strategy and organization (higher levels of abstraction) and some within the sphere of influence of computer science/engineering (lower levels of abstraction). However, ultimately, *the success of the organization in terms of the fulfillment of its aims is determined by the degrees of connectedness and cooperation achieved by its interacting actors, both internally and externally, regardless of the level of analysis*. Hence, the approach is clearly actor-oriented. Another important premise is that organizational design is both an outcome and a process of designing, unique to human organizations. As a process, organizational design can (and should) be inspired on the design processes of other types of artifacts, such as buildings, bridges, surgical operations, or computer software, but cannot be treated as being *the same* as such processes. In adopting an actor-oriented approach, it must be conceded that organizational design thinking cannot be reduced to the static plotting of entities and relationships in an architectural plan or model. The process is live and dynamic and the original design keeps changing, making it difficult to create a neat division between design and implementation.

A review of definitions of organization identity shows that this concept can easily accommodate the notion of OSA and that both share a number of elements, although to varying degrees. The most important are:

- *Meaning*—In line with human-centered design, Scott and Lane (2000, p. 49) state *people's perceptions of organizations are not simply descriptive statements of organizational features, attributes, and characteristics, but also evaluations relating to cultural values, definitions, and meanings*.
- *Emotions and feelings of belonging*—Cornelissen et al. (2007, p. 3) define organizational identity as the "shared meaning that an organizational entity is understood to have arising from its members' (and others') awareness that they belong to it, including emotional and value significance of that group membership."
- *Skills and habit*—Dutton and Dukerich (1991, p. 546) claim that an organization's identity is "closely tied to its culture because identity provides a set of skills and a way of using and evaluating those skills that produce characteristic ways of doing things. Cognitive maps, like identity, are closely aligned with organizational traditions."

Seidl (2005) has put forward a model of identity divided into three interacting parts: (1) self-description (internal description of the organization), (2) image (internal description of the external description of the organization), and (3) reputation (external description of the organization). The model, which brings together many of the topics and components of organization identity featured in the literature, has been modified to include also individual identity and identification. The process model can be seen in Fig. 11.2.

11.6.1 Internal Self-Description

Self-description refers to the collective perception on the part of the organization's stakeholders about the organization's identity. By *identity*, it is not meant one monolithic whole, but multiple images of identity, formed by tacit understandings sitting alongside overt forms of identity, for example, the physical premises or the company's products (Schultz et al., 2000). The organization's self-description is

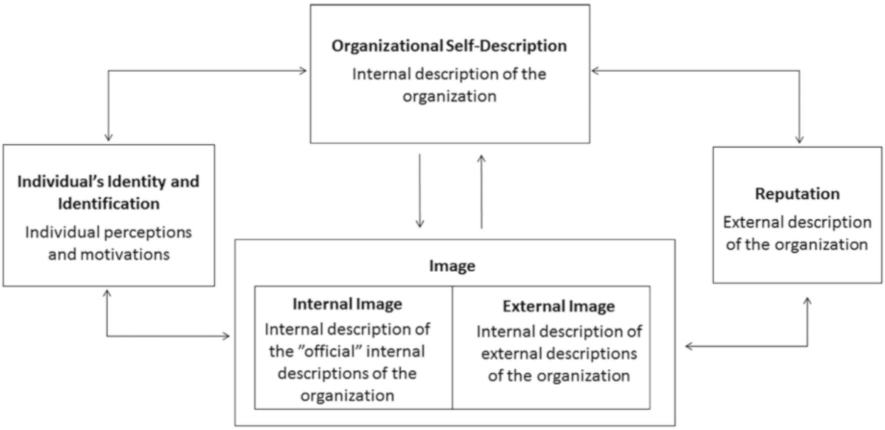


Fig. 11.2 The processes of organizational identity; modified from Seidl (2005)

influenced by the activity of organizational members, as well as the organization's image, internal and external. For example, when companies advertise, campaigns target not only consumers but also employees, community members, suppliers, the media, as well as stockholders and investors. In a similar fashion, campaigns to build brand identity or brand equity are often designed to develop the company's corporate identity, either by building up the company's image or by advocating support for the company's policies and programs. All these communication practices are powerful contributors to the organization's internal self-description.

11.6.2 Reputation

Reputation refers to descriptions of the organization made by external stakeholders. It can be based on observation of explicit identity manifestations from the organization or inferred from attitudes or stances taken by the organization. Reputation has an indirect influence on the organization's self-observation as reputation-driven actions in the environment trigger responses from within the organization (Dukerich et al., 2002; Dutton & Dukerich, 1991).

11.6.3 Image

Image refers to descriptions of the organization made by internal stakeholders, and it may denote an internal orientation or an external orientation. Internal image refers to internal description of the "official" description of the organization, while external image refers to internal descriptions of the external descriptions of the organization (Brown et al., 2006). The distinction between the two types of image is important because they elicit different kinds of responses from organizational members. Internal image is strongly influenced by managerial pronouncements inside the organization, not generally known in the outside, while external image is shaped by everything the organization does with an external impact. Both have important repercussions in organizational identity; however, internal image has a more direct and dramatic impact on the identification and motivation of individual members.

11.6.4 Individual's Identity and Identification

The framework hinges on the crucial point of identification with the organization, defined as an individual's willingness to commit to the organization and to contribute to its goals in a positive and truthful manner. Identification grows in tandem with identity and with the individual's self-observations, which confirm a certain type of phenomenon or expectation. Situations of no confirmation of self-descriptions lead to *diminished identification*, which may hamper change or cause undesirable changes to occur in behaviors and relationships in the organization. Identification is the outcome of processes of identity formation, which may or may not be consistent with the expectations created in the individual organizational member, through the organization's communicative artifacts, including its managerial pronouncements.

11.7 The Organizational Identity: Self-Awareness Framework

Combining the above dimensions of identity with Alvesson and Empson (2008) empirically derived framework of themes involved in the construction of organizational identity, we have developed our own framework integrating OSA with identity. The result is shown in Table 11.1, featuring the four dimensions, the associated components of organizational awareness/identity, and two columns containing a subjective assessment of the degree of relevance of each component to awareness and identity. The aim is to create a reflection opportunity about (1) the suitability of the identity rationale for OSA and (2) the identification of potential research topics, using the integrated framework.

Table 11.1 OSA framework

Identity/OSA Dimensions	Associated Themes or Components of Organizational Awareness/ Identity	Relevance to Awareness	Relevance to Identity
1. ORGANIZATIONAL SELF-DESCRIPTION (KNOWLEDGE & WORK) What do we know and how do we work? How do we provide service to clients?	▪ Knowledge content	2	1
	▪ Work processes	2	1
	▪ Enterprise Cartography	2	0
2. INTERNAL IMAGE (MANAGEMENT & MEMBERSHIP) How is the organization managed and how do members relate to each other and to management?	▪ Formal structure and systems	2	1
	▪ Informal structure and systems	1	2
	▪ Managerial decision-making	0	2
3. INDIVIDUAL'S IDENTITY & IDENTIFICATION Where I am in relation to the organization's aims, objectives, values and principles?	▪ Aims/Objectives	2	2
	▪ Values/Ethics/Morality	0	2
4. EXTERNAL IMAGE How are we seen and how do we see others?	▪ Reputation	1	2
	▪ Clients and competitors	2	1
	▪ Other external stakeholders	2	1

Key: 0 = Not relevant. 1 = Of little relevance. 2 = Very relevant

11.7.1 Organization's Self-Descriptions (Knowledge and Work)

In terms of identity, this section captures the internal perceptions of the organization about "Who we are," and it does so in terms of the knowledge and processes that shape the nature of our work. The question "What do we know?" goes beyond basic descriptions of the content of the technical knowledge of the firm, but encompasses how knowledge is conceptualized within the organization and the extent to which knowledge is formally managed and codified. The question "How do we work?" addresses the way in which service is delivered to clients, i.e., how knowledge, skills, and attitudes are put to work in servicing clients. The answers to these questions are the self-description of the organization.

In terms of awareness, identity can be reinforced if members are aware not only of the current status of business processes but also of their standing in the relevant processes. This is part of the domain of enterprise cartography (EC). EC principles are found within the discipline of business process management, especially in process mining. These techniques make use of event logs to discover process activities and to assess the conformance of existing processes against constraints (van der Aalst, 2011; van der Aalst et al., 2012; Agrawal et al., 1998). Mined processes correspond to the actual instances of processes, not to the designed or modeled processes. Another example of EC is the inference of inter-organizational processes based on EDI event logs (Engle et al., 2012). Mining techniques have been applied to other organizational structures and social networks from event logs.

Business intelligence (BI) techniques that collect data from organization systems to produce reports and dashboards are another example of EC. Business intelligence actually supports the feedback control loop by providing managers with a model of the organization that allows them to ground their actions and decisions. However, business intelligence is mostly concerned with analytic aspects of the business and thus addresses a focused aspect of cartography. Essentially, BI techniques also play observer and controller roles within organizational feedback loops. Tribolet et al. (2014b) aim at a more generic and systemic approach, where several OSA artifacts, such as the enterprise current and future state, are displayed through dashboards, and define organization variables as specific information of a given organization artifact.

11.7.2 Internal Image (Management and Membership)

This dimension focuses on the informal and formal systems and structures that support the delivery of that service to clients, as well as the links between the organization and its members. The core questions, therefore, are "How is the organization managed?" and "How do organizational members relate to management and to the organization?" Does the organization emphasize a performance orientation in the form of explicit measurement and rewards, or does it place normative controls, based on values, beliefs, and forms of symbolism? Answers to these questions, along with features such as creativity, performance, status, interpersonal relations, pay, and career prospects, help to define image and identity. The dimension also has a strong emphasis on identification and intends to capture how the informal and formal structures and systems are utilized to mold ideals, as well as to motivate or to control organizational members.

In terms of awareness, a technology-enabled assessment of the effectiveness of management structures would greatly reinforce the perception of internal image.

11.7.3 Individual's Identity and Identification

This component is concerned with the way in which organizational identity has an influence on personal values and vice versa. It intends to tap the way members are carriers of the morals or values contained within the organizations' managerial pronouncements and behavior.

In terms of awareness, this dimension concerns work-related individual indicators, which provide answers to the question "where I am in the context of the achievements of the organization?" The indicators might be linked to the achievement of personal objectives, such as in the case of balanced scorecard systems.

11.7.4 External Image

Organizational identities are mainly constructed within organizations; however, organizational members are also strongly influenced by their interactions with the external world. The question "How are we seen?" is a reflection of how members believe themselves to be perceived by others, especially clients, suppliers, and competitors. In terms of identity, the external image is a "highly personalized view" (Alvesson & Empson, 2008, p. 10) that works through the perceptions of individual members. Therefore, the organization's external image has implications for individual's personal identity and also has an influence on the question of "how do we see others?"

In terms of awareness, there are multiple methods for objectively assessing external image, such as surveys aimed at informing about the awareness about the company's products or services (e.g., brand awareness).

11.8 Conclusion

In this chapter, we make a proposal toward a convergence framework between concepts—OSA and organizational identity—as part of a much needed integration effort between enterprise engineering and a situated interpretation of organizational design. Such a proposal is presented with human-centered design as the intellectual background. As aforementioned, one aim of the proposal is to identify potential research topics. The integrated framework fosters multidisciplinary studies that take into account both identity and organizational awareness issues. In particular, the framework is intended at promoting studies aimed at assessing the effect that OSA tools, methodologies, or models have on organizational identity. Conversely, it also aims at informing studies that develop forms of organizational self-awareness, based on theories of organizational identity.

Design research (Hevner et al., 2004) may prove a valuable approach for this type of research since it puts together behavioral approaches to develop and justify theories about business phenomena and design science approaches to build artifacts and evaluate their utility in satisfying business needs. In brief, the challenges ahead are many. Properly defining the most adequate research lines and teams will not be a straightforward issue. However, from our point of view, design theory is the way forward in achieving true convergence between OD and EE.

Chapter 12
Conclusion

Kazem Haki, Bas van Gils, and Henderik A. Proper

This conclusion briefly reflects on the contributions of the chapters in this part.

As outlined in the introduction of this part, digitalization, specifically digital transformation, is the recent wave in studying the role of information technologies. This wave discusses how digital technologies have changed the fabric and organizing logics of organizations (Besson & Rowe, 2012; Yoo et al., 2010; Zammuto et al., 2007). The constituent chapters of this book contribute to the ongoing discussions in this wave.

Research on strategic management argues organization's need to three types of capabilities, namely, *operational*, *dynamic*, and *improvisational*, in dealing with turbulent environments. Operational capabilities are organizational routines and processes that are developed over time through learning and provide organizations with the capacity to undertake activities in a reliable manner (Winter, 2003). Dynamic capabilities are forward-looking capabilities by which organizations extend, modify, or reconfigure existing operational capabilities into new ones in response to disruptive technological shifts and innovations (Winter, 2003; Teece, 2007). Improvisational capabilities are second-order dynamic capabilities by which organizations spontaneously reconfigure existing resources into new ones to address urgent and unpredictable environmental situations (Pavlou & El Sawy, 2010). In specifically dealing with digital transformation, Chap. 10 proposes a typology of

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_12

capabilities and an ordinal typology of organizational logics. It argues that, in facing today's turbulent digital environment, organizations need a *co-adaptive* organizing logic and should equip themselves with all types of capabilities ranging from operational (*routine* capabilities, in the author's terminology) to dynamic (*systematic* and *creative* capabilities, in the author's terminology) and improvisational (*adaptive* and *generative* capabilities, in the author's terminology) capabilities.

Digital transformation is multidisciplinary and socio-technical in nature that needs an integration between organizational and technical knowledge to account for its inherent aspects. To this end, Chap. 11 proposes an integration between *enterprise engineering* and *organization design* disciplines. This chapter further instantiates such an integration between the disciplines by discussing the integration between the notions of *organizational self-awareness* and *organizational identity*, with the aim to identify potential research topics in fostering multidisciplinary studies.

To conclude, the constituent chapters of this part make us aware of the need for a new organizing logic and its associated digital capabilities to encounter digital transformation. They further give rise to socio-technical nature of digital transformation requiring a multidisciplinary standpoint in its investigations.

Part III
An Architectural Coordination Perspective

Chapter 13
Introduction

Kazem Haki, Bas van Gils, and Henderik A. Proper

As of its inception in the late 1980s, enterprise architecture (EA) and its associated approaches and principles draw a considerable attention in both research and practice. The early approaches to EA essentially promoted an enforcement-centric doctrine to systematically guide local technology initiatives toward long-term objectives. More recently, however, a mix of enforcement-centric with influence-centric doctrine is promoted to account for organizations' need for agility to effectively and promptly adapt with changes in the contemporary hyper-turbulent environments (Winter, 2014). Notwithstanding such advances in EA, there is an ongoing discourse on whether and how EA can be employed to tame the complexity of the environment and to deal with challenges associated with digital transformation. This discourse is indeed provoked by referring to the raison d'être of EA in systematically deriving long-term and consensus-based directions to an organization's technology investments rather than swiftly responding to changes in the environment.

To contribute and give direction to the abovementioned discourse, we first need to have a thorough understanding of EA's theoretical foundations. In effect, EA comprises both a descriptive and a prescriptive aspect, as two sides of the same coin (Greefhorst & Proper, 2011a; Haki & Legner, 2021). The descriptive aspect is associated with the artifacts representing an organization in its as-is

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_13

and to-be states, with a focus on creating architecture representations to depict and explain an organization's design in terms of its constituents, properties, and relationships (Haki & Legner, 2021). The prescriptive aspect, in turn, emphasizes the principles governing architecture's design and evolution, drawing attention from architecture representations toward the architectural shape and the question of "how" organizations should be designed and built (Haki & Legner, 2021). Although the twin descriptive-prescriptive aspects have been inherent in the EA concept, the extant literature mostly emphasizes the descriptive side, such that the prescriptive side (more specifically *architecture principles*) remains the crux of the EA concept.

The constituent chapters of this part seek to make advances in prescriptive EA. While Chap. 14 discusses how architecture principles are defined and implemented in EA initiatives, Chap. 15 raises the need for brand-new architecture principles in the realm of digital transformation, and Chap. 16 proposes a principle-based framework for digital transformation. After seeing all these chapters, in the conclusion of this part (Chap. 17), we will discuss how EA can be employed to adapt with changes in the contemporary hyper-turbulent environments and to address digital transformation's associated challenges.

Chapter 14
IT Architecture Principles: Foundation for Digital Transformation?

Michiel Borgers and Frank Harmsen

Abstract Information systems are a vital part of the digital enterprise, and implementation of information systems plays an important role in digital transformation. The core hypothesis of our research is that one of the success factors of information systems implementation is the use of IT architecture principles. In this chapter, we describe the results of our empirical research to validate a measurement instrument measuring IT architecture principles, starting from a hypothesis that IT architecture principles contribute to the implementation of the IS requirements. We describe the results of four case studies at the Dutch Tax and Customs Administration: how can we define, describe, and measure IT architecture principles, with the focus on practical application and quantitative comparability?

14.1 Introduction

Digital transformation is a complicated endeavor for most enterprises. It usually is a series of transformations, typically "wrapped" in several programs and projects. In larger enterprises, the digital transformation requires not just building new systems but also taking into account the proverbial legacy that are in many cases still the core information systems of an enterprise. The complexity is increased by external influences, such as laws and regulations, and a dynamic technological context. The portfolio that encompasses digital transformation therefore needs mechanisms to handle this complexity.

Judging whether that portfolio is well aligned requires insight into the desired overall result as well as the planned and achieved effects of the individual transformations. It also puts a major challenge on the management of an enterprise to make

M. Borgers (✉)
Dutch Ministry of Defense, The Hague, the Netherlands
e-mail: michiel@deklimboom.eu; mac.borgers@maastrichtuniversity.nl

F. Harmsen
Maastricht University, Maastricht, the Netherlands
e-mail: f.harmsen@maastrichtuniversity.nl

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_14

the right decisions at the right time and ensure that these decisions are translated into the right actions. As discussed in Greefhorst and Proper (2011a), enterprise architecture offers a means for management to obtain insight, as well as to make decisions, about the direction of enterprise transformations. This is what Harmsen et al. (2009) refer to as *informed governance*.

Informed governance is a way to handle complexity—not to reduce it! Enterprises need to accept that size and a long history imply a certain degree of complexity. Once accepted, the question would be as follows: "How can enterprise architecture contribute to handling and managing complexity, in order to achieve informed governance and, in the end, a digital enterprise that achieves its objectives?"

In our research, we focus on a specific aspect of enterprise architecture: architecture *principles* that oversee the implementation of an *information system* (IS), being part of a digital transformation. Research indicates that principles are positioned at the foundation of a digital transformation (Greefhorst & Proper, 2011a; Stelzer, 2009; Greefhorst et al., 2013)—without principles, a successful digital transformation is hard to achieve.

However, not much has been published about the practical application of principles—do they deliver what they promise (Borgers & Harmsen, 2016)? Given the assumed importance of principles for digital transformation and the digital transformation, this is a relevant question.

In this chapter, we describe the results of our empirical research into enterprise architecture principles, as a follow-up to the work we described in Borgers and Harmsen (2016). We choose a subset of enterprise architecture, IT architecture, and we try to identify and measure IT architecture principles. Our assumption is that in the near future, information systems will be fully integrated into the digital enterprise and will affect the way digital enterprises operate. Because our hypothesis is that IT architecture principles contribute to the implementation of the IS requirements, it is important to have more insight in the use of IT architecture principles. To get more insight in the use of principles, we need a measurement instrument to measure those principles. Our investigation is based on the question "How are IT architecture principles defined, described, and measured?" The answer to this question can be used to investigate the contribution of IT architecture principles on the successful implementation of information systems, which is, in itself, a vital part of digital transformation.

This chapter is structured as follows. We start in Sect. 14.2 with an introduction to our research on IT architecture principles, including a perspective on their definition and description, and our research approach. In Sect. 14.3, we present our instrument to measure the contribution of IT architecture principles to the successful implementation of information systems. We validated the measurement instrument in four case studies in a large Dutch enterprise that is currently in a digital transformation: the Dutch Tax and Customs Administration (DTCA). The results of this validation are presented in Sect. 14.4. We conclude this chapter with some reflections on the case studies in the context of digital transformation.

14.2 Research on Enterprise Architecture Principles

Principles are "central elements of enterprise architecture frameworks" (Borgers, 2021; Borgers & Harmsen, 2016). They play the regulative role of enterprise architecture. It turns out that architects have a positive attitude toward architecture principles (El Emam & Koru, 2008). But so far, there is no empirical evidence of any contribution of principles to the success of information systems implementation (Greefhorst & Proper, 2011a; IEEE, 1990). And we all know that in current practice, the implementation of information systems in itself is not obvious (IIBA, 2015; The Open Group, 2011; IEEE, 2000). So, do IT architecture principles work in practice? To be able to do empirical research on the contribution of IT architecture principles, we first have to identify them. What is the difference between an architecture principle and a guideline, technology standard, or design principle? A good definition is necessary for the right identification. Subsequently, we need a way to describe the architecture principles distinctively, so we can compare them. For example, what are the differences between the principles "good is good enough" and "business data have to be used throughout the application"? Which of those architecture principles is formulated well, or are they both adequate? A description of the characteristics of the architecture principles will help in answering this question.

14.2.1 Definition of a Principle

In general, there are two types of enterprise architecture principles: design principles and representation principles (Borgers & Harmsen, 2016; Stelzer, 2009; Haki & Legner, 2012; Winter & Aier, 2011; Fischer et al., 2010; Aier, 2013). The latter type refers to the way architectures should be represented, while the first directs the design of a system. Some authors define enterprise architecture principles with both types in scope. We take the view that those two types have a different kind of nature. Moreover, we want to investigate the contribution of IT architecture principles to the successful implementation of information system requirements, which justifies defining enterprise architecture principles as design principles.

Based on (Greefhorst & Proper, 2011a; Stelzer, 2009; Haki & Legner, 2012, 2013), we define architecture principle as follows:

> An architecture principle is a declarative statement based on business and/or IT strategy that normatively describes a property of the design of an artifact, which is necessary to ensure the artifact meeting its essential requirements. The architecture principle is described using a group of characteristics in such a way that people involved can understand and use it when elaborating the design of the artifact.

In this general definition, we use the term "artifact," instead of a specific scope of the architecture (such as business, information system, or technology). In our

Table 14.1 Characteristics and corresponding attributes of individual IT architecture principles

Characteristic	Attribute	Definition
Specification		Description of the principle according to The Open Group (2011):
	Statement	Succinctly and unambiguously communicates the fundamental rule
	Rationale	Highlights the business benefits of adhering to the principle
	Implications	Highlights the requirements for carrying out the principle
Quality		Five attributes defining the quality of the principle The Open Group (2011):
	Specific	Specific enough for the user to understand its intention and its effects to use it in his work
	Measurable	Possible to determine whether or not a given behavior is in line with architecture principle
	Achievable	The implications of it can all be performed by or adhered to by all those affected
	Relevant	The principle should lead to a significant improvement of the information system meeting the requirement
	Time framed	Principe should be stable in context and time
Usage		The degree the directive of the IT architecture principle can be detected in the implemented information system (in the usage stage)

research, the research object is the information system, and the artifact is (a part of) an information system.

14.2.2 Description of a Principle

Although we now have defined the essence of enterprise architecture principles, we still have to describe them. Therefore, we answer the question "With what characteristics are enterprise architecture principles described and what are their interdependencies?" In answering this question, we distinguish between the characteristics of individual enterprise architecture principles and the set of architecture principles. Characteristics which are summarized in Table 14.1.

An enterprise architecture principle is usually part of a set of principles. Although in most literature the focus is on individual architecture principles, a principle is only effective if it is an element of a set (Greefhorst & Proper, 2011a; Stelzer, 2009; Lindström, 2006; Marosin & Ghanavati, 2015; Marosin et al., 2016). Therefore, to be able to measure the contribution of enterprise architecture principles, we have to describe "the enterprise architecture principle set" as well. To do so, we introduce the three different characteristics as shown in Table 14.2.

Table 14.2 Characteristics and corresponding attributes of the set of architecture principles

Characteristic	Attribute	Definition
Number		The amount of IT architecture principles used
Classification		Type of IT architecture principles used focusing on part of the information system: IS generic, application, or technical infrastructure (Greefhorst & Proper, 2011a)
Quality of the set		Three attributes defining the quality of the set of IT architecture principles (Greefhorst & Proper, 2011a):
	Representative	The set is representative for the problem domain and covers all relevant requirements
	Accessible	Users can found and retrieve the set of principles and they are easy to comprehend
	Consistent	No obvious conflicts between the IT architecture principles in the set

14.3 Measurement Instrument and Approach

Now that we provided a definition and a description of enterprise architecture principle, we are able to develop a measurement instrument for measuring the characteristics of the IT architecture principles. In this section, we explain the measurement instrument we used in the case studies, by describing the characteristics measured. Next we describe the method of quantifying those characteristics. We finish with a description of the approach of case data collection and processing.

14.3.1 Measuring IT Architecture Principles

We would like to measure the characteristics of each IT architecture principle meeting our definition. Furthermore, we also would like to measure the characteristics of the set of IT architecture principles. Given the fact that there are many characteristics to specify IT architecture principles, we started with a basic set, because we did not know in advance which characteristics of IT architecture principles are important to the contribution of the implementation of IT system requirements. And using all possible characteristics in the measurement instrument would not be feasible. We used those characteristics, which were mentioned most in literature.

For measuring the individual IT architecture principles, we used the characteristics listed in Table 14.1.

We measured the characteristics listed in Table 14.2, attributes included, to specify the set of IT architecture principles.

We are aware that some of the characteristics are dependent on each other. For example, an IT architecture principle cannot be specific, if there is no statement defined. In the analysis of the case study results, we took those possible dependencies into account.

14.3.2 Measuring Method

The first step was to determine all IT architecture principles used for implementing the IT system. To do so, we checked whether or not statements, found in documents and interviews, did meet the definition of an IT architecture principle.

Step 2 was to measure the characteristics of the individual IT architecture principles and of the set of IT architecture principles. Our aim was to quantify the characteristics of the IT architecture principles as much as possible, because this would enable us to compare the results between the case studies (Earl, 2015; Muijs, 2004). For example, we counted the amount of IT architecture principles in a set. For some other characteristics, we introduced a three-point scale to determine the level of presence of those characteristics. A low score (0) implied the characteristic or related attribute was (almost) not present, a medium score (0.5) corresponded with partly present, and a high score (1) meant (almost) completely present. This three-point scale was used to achieve an adequate measurement of the characteristic, without suggesting a precise significance. In case of a characteristic definition with more attributes, the average score of all attributes became the overall quantification of the characteristic. Although we used this quantitative approach for measuring the quality criteria, the basis of the score was still an expert opinion of the researchers.

14.3.3 Data Collection and Processing

To collect the data needed to enter into the measurement instrument, we used both interviews and desk research. We interviewed senior users, architects, and designers about the IT system and the IT architecture principles. We always interviewed the person(s) concerned with two researchers to verify the data gathered in the interviews. The interview was documented, and documentation was sent back to the interviewee for validation.

During the desk research, we reviewed different kinds of documents, such as requirement specifications, architecture descriptions, and design documents of the IT system concerned. When data in the documentation were not self-explaining or conflicting, we used the interviews to clarify those data. Finally, we processed all data collected by entering the data into the measurement instrument. One researcher entered the score into a simple spreadsheet, based on the data collected. A fact basis was added to each score. The second researcher reviewed the scores, and in case of disagreement, the facts were discussed with each other. The last step was to discuss the scores and to reach consensus about the overall conclusions.

14.4 Case Studies

We conducted four case studies in the Dutch Tax and Customs Administration (DTCA) to test our measurement instrument. The DTA is responsible for the tax, allowance, and customs in the Netherlands. To support those business functions, DTA has a variety of information systems available. We selected four of them differing in function, size, amount of users, age, and development approach of the information system. We describe the cases based on the STARR[1] aspects. With the Situation aspect, we describe the context of the information system in the organization. Task exposes the essential requirements and how the information system meets them. The IT architecture principles used are described in the Approach and evaluated in the Results section. Each case study ends with a Reflection on the identification of the IT architecture principles and the measurement of the characteristics.

14.4.1 MON

14.4.1.1 Situation

The Dutch Tax and Customs Administration has a department responsible for monumental buildings (in Dutch, "Monumentenpanden," therefore, MON). This department has to determine the fiscal consequences of renovations of historical buildings and of specific rentals. Four employees of this department have to handle around 2,000 cases a year. To support this process, they use the information system called MON, which supports the registration and processing of all cases. MON is a non-critical information system with no connections to other business applications.

14.4.1.2 Task

Originally, MON is an information system developed in 1999 and rebuilt in 2004 with new technology using a reversed engineering methodology. In 2014, MON has been rebuilt again because of the availability of newer technologies.

The core functionality of the MON system is reading incoming letters, calculation of tax returns, and printing letters based on the decision of the tax employee. One important function is missing: the business intelligence export function, for the use of management information. MON functionality is implemented with front-end client software on desktop and the use of a database server positioned in the data center, accessed via a network. This front-end client software is loaded during runtime from a distribution server to the client desktop. To print letters, Word is used

[1] STARR: situation—task—approach—result—reflection.

Table 14.3 Measurement results of the IT architecture principles of MON

System	Specification	Quality	Usage	Number	Classification	Quality of the set
MON	0.5	0.9	1.0	2	1 generic 0 application 1 infrastructure	0.5

as a separate application. The implemented MON system has a size of 131 FPAg. The only explicit quality-of-service requirements defined were the amount of users and the maximum amount of cases per year. Although not documented, the MON system should be available during office hours, and the response times have to be acceptable for the end users.

14.4.1.3 Approach

We identified two IT architecture principles used to develop MON. One of those principles was documented explicitly; the second one was only passed on orally in the development team. One principle states “there are no relationships between the MON system and other information systems.” The second principle prescribes “the data of the application should be placed on a centralized managed server to ensure availability of the data after incidents.” Both principles were not quite well specified: the rationale and implications were described incompletely (see Table 14.3). We calculated the score for the specification of the IT architecture principles with a result of 0.5. The calculated score for the quality of the IT architecture principles was high (score 0.9). Only for one principle, the attributes “specific” and “achievable” were not concrete enough. Based on the interviews and documentation, we concluded the implemented MON system did meet the IT architecture principles as defined in the development stage; so the score for usage was 1.0. One of the IT architecture principles is focusing on the technical infrastructure; the other one is a generic principle. For the quality of the set scored MON only a 0.5, because the IT architecture principles are hard to retrieve and to comprehend. Besides, the set of principles did cover some but not all relevant requirements.

14.4.1.4 Result

As could be seen from the documentation, the original MON system was built in a short period of time with a small team. The explicit use of IT architecture principles was not so obvious. At the start, no explicit set of IT architecture principles was defined as we could see in the quality of the set of principles and the fact that only one principle was documented. This case was also interesting because one of the IT architecture principles used was well known by the development team, but not documented explicitly. In this case, however, we determined this had no negative effect on the quality or usage of the IT architecture principle. On the contrary: because the two principles prescribed the norm for the design of the IS system explicitly, the quality characteristic scored high.

14.4.1.5 Reflection

In this case, it was possible to identify the two IT architecture principles although the explicit use of IT architecture principles was limited. With the help of the MON architect and the few documents available, we could identify the principles satisfying our definition. We were also able to measure the characteristics of those principles, maybe because of the clearness of the normative description of the principles.

One limitation in this case research was the availability of information to the researchers. Because the MON system was rebuilt twice in 15 years, the knowledge of the development of the MON system was limited and therefore difficult to verify with other sources. However, the conclusions we drew were based on information available from different employees and documentation.

14.4.2 PBT

14.4.2.1 Situation

DTA is also responsible for the payment of allowances to Dutch citizens. To support this process, DTA uses an allowance system, which contains many parameters (approximately between 400 and 500). To manage all those parameters, they built a separate information system called Parameter Management Tool (in Dutch: “Parameter Beheer Tool,” PBT abbreviated). The objective of PBT is to manage the (functional) parameters, parameters and subtypes, and domains and reference tables of the allowance system.

14.4.2.2 Task

The core functionality of PBT is to show, change, and approve all kinds of parameters. Also, all parameters can be replicated to the allowance system. PBT has a consistency check with the allowance system components to ensure the right parameters are in place. Furthermore, a reporting function is available to get insights in the history of all values of the different parameters. Most quality-of-service requirements are equal to the allowance system, meaning the availability of the PBT system should be near 100% and response times has to be near real time. There are eight employees using PBT.

Table 14.4 Measurement results of the IT architecture principles of PBT

System	Specification	Quality	Usage	Number	Classification	Quality of the set
PBT	0.6	0.8	1.0	28	9 generic 17 application 2 infrastructure	0.8

Technically, PBT is a custom-built application in .Net using a separate SQL database. The PBT application has its own website with a web interface to a client browser. PBT is using Excel for the reporting function. There are connections with a zip code database and with 13 applications of the allowance system. The first version of PBT has been implemented on October 2013. Since then, different releases are introduced. In this research, we investigated the latest operational version of PBT, the N.1900 release.

14.4.2.3 Approach

During the development of PBT, there were 28 IT architecture principles defined and more than 35 design principles as well. Only six IT architecture principles were specified with a "statement," "rationale," and "implication," and even those descriptions were not always clarifying. In this case study in particular, the rationale of many IT architecture principles were not present. The quality level of the IT architecture principles was relatively high (score of 0.8; see Table 14.4). Only the relevance of 19 IT architecture principles could not be determined, and a few other IT architecture principles were less effective on some other quality attributes. The implemented PBT system seems to meet the directions of all IT architecture principles defined, based on both the available documentation and the architect's clarifications.

The 28 IT architecture principles can be divided into principles focusing on the application layer (2), on the technical infrastructure layer (17), and on the PBT system generically (9). The overall score for the quality of the set of principles was 0.8. The IT architecture principles were accessible, and there were no inconsistencies. The set of principles were less representative: the connection between the IT architecture principle and a (essential) requirement was often indefinable.

14.4.2.4 Result

With 28 IT architecture principles defined and traced back in the implemented PBT system, it is obvious that PBT was developed using IT architecture principles. But in this case study, it was hard to determine whether or not all IT architecture principles were needed to ensure the essential requirements. One important reason for this is the lack of rationale of many IT architecture principles. Secondly, many

IT architecture principles were focusing on specific details of the PBT system: the implementation of user authorization using a specific application component or using a specific technical infrastructure component for the audit function. That might be an indication that many of those were design principles instead of architecture principles. But because of the non-existence of the rationale, we could not determine whether or not they were necessary for implementing the essential requirements.

It is interesting to mention both users and developers were satisfied with the implemented PBT system. The PBT did meet all requirements, and during the use of the system, only a few incidents did occur, as mentioned by the architects.

14.4.2.5 Reflection

The identification of the IT architecture principles was difficult in this case study. There were many IT architecture principles, design choices, and assumptions defined. But when there is no direct or indirect link described between a potential IT architecture principle and an essential requirement, it is hard to select the right set of IT architecture principles. In this case study too, the measurement of the characteristics was uncomplicated (see the appendix for all IT architecture principles of PBT). The IT architecture principles were addressing specific properties of the design, and therefore, we could rate the characteristics easily.

Because PBT is a relatively new IT system, all kinds of documents and people involved were available during this research. So in this case, the information gathering and validation was easy.

14.4.3 PDO

14.4.3.1 Situation

DTA has built a portal environment (in Dutch: “Persoonlijk Domein Ondernemers,” abbreviated as PDO) for companies to do their declarations for tax returns. Small- and medium-sized companies (up to ten employees) can do their declarations for the VAT, income tax, excise tax, and cooperation tax as the most important ones. With 1.5 million users and 1.25 million companies, it is a well-used information system. PDO requires a high availability for users. PDO was introduced in 2005.

14.4.3.2 Task

In 2007, PDO has been extended with the functionality to report excise tax items. In 2011, the so-called Mini One Stop Shop (tax legislations for foreign entrepreneurs) also felt under the responsibility of PDO. Recently, it has been decided to merge

Table 14.5 Measurement results of the IT architecture principles of PDO

System	Specification	Quality	Usage	Number	Classification	Quality of the set
PDO	0.8	0.7	0.8	6	5 generic 1 application 0 infrastructure	0.8

the PDO system into another domain of the Dutch Tax and Customs Administration (Mijn Belastingdienst).

PDO consists of six application components, of which one is the core application with the key functionality. Basically, PDO runs on a large server connected via an internal network with desktops and connected with the Internet for the external users. PDO has three interfaces with other IT systems for data exchange. PDO is an important information system for receiving the declarations of entrepreneurs. Besides an availability of at least 98%, the response time of transactions should be 5 seconds at most. Because PDO processes confidential information of entrepreneurs, extra security regulation is mandatory.

14.4.3.3 Approach

PDO was developed based on an IS architecture, including three IT architecture principles compliant with our definition. In time, extra IT architecture principles have been added to the original list of principles. For example, extra principles were added recently, because PDO will be merged with Mijn Belastingdienst and PDO will become a "legacy" IS system. We rated the specification of the IT architecture principles with 0.8 (see Table 14.5). Three of the principles were well defined, while the other three were lacking some information on the rationale or implications. The three IT architecture principles, which were well defined, scored also well on quality (score 0.9). But the quality level of the other principles was much lower, so the overall quality score became 0.7. Most of those principles were not specific enough for the user to understand ("specific" attribute) and were also less effective on the other quality attributes. With two IT architecture principles, we could not detect in the implemented information system to what extent they were directive. The limited description of those principles was the main reason for this.

In our research scope, PDO had six IT architecture principles, of which there are five generic architecture principles and one application architecture principle. The set of IT architecture principles were consistent: no conflicts between the principles were found. The accessibly of the principles was not optimal. For example, the IT architecture principles were described in different documents, and IT architecture principles were not always explicitly addressed. The set seems to be representative for the requirements defined, although we could not determine which IT architecture principles were available in past releases. So the overall score for the quality of the set was 0.8.

14.4.3.4 Result

We did see in this case three IT architecture principles well defined for all characteristics, while for the others, there was an imperfection on almost all characteristics. Despite those imperfections, we did see the guidance of the IT architecture principles back in the implemented PDO system. We took note of the fact that all essential requirements were implemented in PDO, whether or not effected by the IT architecture principles. We did not find any technical infrastructure-related IT architecture principles. Because the development team had to use the standard technical infrastructure of DTA, no detailed architecture and/or IT architecture principles were defined.

14.4.3.5 Reflection

In this case, we could identify IT architecture principles, although it was sometimes difficult to differentiate IT architecture principles from standards and guidelines. It might be that we addressed IT architecture principles as guidelines because they were mentioned so in some documents and we could not find out whether or not it gave direction to the property of the design. The characteristics of the IT architecture principles could be measured as well. Of course, it was easier to measure those principles, which were described extensively.

14.4.4 DOBRA

14.4.4.1 Situation

DOBRA is a message system used by the Dutch Customs Agency, an agency operating in a 24/7 economy and with highly automated processes. The DOBRA system takes care of the system-to-system communication with external organizations, like transport companies and foreign customs services. So DOBRA is an internal message broker between the internal customs systems and systems of external organizations. It receives different types of messages and translates them into a standard format. DOBRA processes approximately 60 million messages annually.

14.4.4.2 Task

The functionality of DOBRA is built in two parts: DOBRA Basic and DOBRA Extension. Validation, transformation, and exchange of messages are the core functions, implemented in DOBRA Basic. DOBRA Extension is responsible for addressing outgoing messages, error handling, and supporting system administrators.

In the current situation, most communication goes directly through DOBRA Basic. In the future, the messages should also go through the extension. The DOBRA system has a size of 1.152 FPAg and 17 interfaces with other systems. Because DOBRA is critical to the business operation of the Dutch Customs Agency, DOBRA is 99.9% available and has response times of less than 0.5 sec. To make the DOBRA system secure, an authorization and logging function has been implemented.

14.4.4.3 Approach

In the development approach of DOBRA, the use of IT architecture principles was an explicit choice. The development team defined eight IT architecture principles in accordance with our definition.

The extent in which the IT architecture principles are specified with a "statement," "rationale," and "implication" was different between the principles. Four of them were specified clearly, but the other four IT architecture principles did lack explicitness on one or more attributes (score 0.7; see Table 14.6). The quality of the individual IT architecture principles was rated with 0.7 as well. Three of the IT architecture principles had a real high quality level (score near 1.0). Other principles did score less because two or more quality attributes were less effective. The score of the usage of the IT architecture principles is relatively low (score .7). The main reason for this result is the fact that the IT architecture principles were defined so generic that we could not find enough evidence of the impact of the IT architecture principle in the implemented IS system. For example, one principle was defined as "good is good enough." Because both descriptions were not that specific and there was no measurement indicator defined, no evidence was found in the implemented IS system.

The eight IT architecture principles could be classified as three generic IT architecture principles, four related to the application, and one related to the technical infrastructure. The quality of the set of IT architecture principles was 1.0. As a set, the IT architecture principles were representative for implementing the requirements. All principles did have a (in)direct link with the most important requirements. The IT architecture principles were documented clearly in an official architecture document, which was used by the development team. Also, no inconsistencies were found between the eight principles.

Table 14.6 Measurement results of the IT architecture principles of DOBRA

System	Specification	Quality	Usage	Number	Classification	Quality of the set
PDO	0.7	0.7	0.7	8	3 generic 4 application 1 infrastructure	1.0

14.4.5 Results

In this case, we did see three IT architecture principles, which were specified clearly, fulfilled the quality criteria, and were detected in the design and implementation of the information system. Four IT architecture principles were defined poorly on several characteristics, and one principle was not defined clearly at all. Although there was some coherence between the characteristics, as mentioned in Sect. 14.3, it was interesting to see that in this case, an IT architecture principle was either defined well for all characteristics or failed (partly) for all characteristics. This effect cannot be explained by the coherence between the characteristics only. DOBRA's requirements were realized almost completely. This was proved by the test results and the small amounts of incidents in the usage stage and confirmed by the users and developers of DOBRA.

14.4.5.1 Reflection

In this case, we were able to identify and measure DOBRA's IT architecture principles using our measurement instrument. We could identify and define the IT architecture principles easily because of the clear documentation and the presence of many people involved with DOBRA. Only some background information was lacking regarding the technical infrastructure, so for one specific principle, we only good use the information of the interviewees.

14.5 Conclusion

We argued that information systems implementation is a vital part of digital transformation. IT architecture principles play an important role in this information systems implementation. Because we stated IT architecture principles contribute to the implementation of the IS requirements, it is important to get more insights in IT architecture principles. Therefore, we tried to define, describe, and measure those IT architecture principles. In the case studies, focused on information systems implementation, we tried to identify and measure the IT architecture principles used. In evaluating our measurement instrument to define, describe, and measure IT architecture principles, we start with reflecting on the four cases. Based on the reflection of both the IT architecture principle set and the individual architecture principles, we give some conclusions on measuring IT architecture principles including suggestions for further research.

Table 14.7 Overview results of the set of IT architecture principles

System	Number	Classification	Quality of the set
MON	2	1 generic 0 application 1 infrastructure	0.5
PBT	28	9 generic 17 application 2 infrastructure	0.8
PDO	6	5 generic 1 application 0 infrastructure	0.8
DOBRA	8	3 generic 4 application 1 infrastructure	1.0

14.5.1 The Set of IT Architecture Principles

Looking at the results of the different case studies, we did see some interesting findings (see also Table 14.7). For example, we identified a large deviation in the number of IT architecture principles used, independent of the functional size of the IS system (between 2 and 28). In three of the four case studies, IT architecture principles were, in general, specified quite accurately. However, we also encountered inadequately specified principles. Of the four cases investigated, there was no case study with on average much better specified IT architecture principles than other case studies.

In these case studies, the technical infrastructure principles, of all types of IT architecture principles, were used less. An explanation for this might be the fact that in most case studies, the common technical infrastructure of DTA has been used. When technical infrastructure is already available, less direction is necessary to the design of the technical infrastructure.

Viewing the results of the quality of the set, all the sets did score positive on the consistency: there were no IT architecture principles found that were conflicting with others. In both comprehensiveness and availability of the IT architecture principles, there were some mismatches in some case studies. In two cases, the set was not representative for the requirements they had to fulfil. In one case, there were more essential requirements than they could tackle with the IT architecture principles. In the second case, there were many IT architecture principles, and there was not always a direct link with the essential requirements. And although we did not examine the implementation of the requirements itself, we did find some indications that if the quality of the set is high, the implementation of the requirements is done adequately.

Table 14.8 Average results of the individual IT architecture principles

	Specification	Quality	Usage
Average over 44 principles	0.7	0.8	0.9

14.5.2 The Individual IT Architecture Principles

By analyzing the 44 IT architecture principles in the 4 case studies, we did discover some interesting findings as well. In Table 14.8, we summarized the average score of the IT architecture principles, while in Table 14.9, we listed all 44 IT architecture principles with their scores. The 44 IT architecture principles were specified with an average score of 0.7. Nine of those principles were accurately described with a "statement," "rationale," and "implication." When the specification of the principle is inadequate, in many cases, the rationale is missing or partly described.

The average quality of the IT architecture principles was scored with a 0.8. For 14 IT architecture principles, the quality level was even higher (score of 0.9 or more). The relevance of the IT architecture principle was the attribute deemed unclear most of the times. Because of the dependence between the attributes "rationale" and "relevance," this is explainable. It is interesting to see that when the IT architecture principle is specified well, then almost always the quality of the principle was high too.

14.5.3 Overall Conclusions and Further Research

Based on the four case studies, we conclude that we can measure the IT architecture principles in real-life cases. The measurement instrument is useful in quantifying the characteristics of both the individual IT architecture principles and the set they are into. Based on these quantifications, we can compare the IT architecture principles and analyze the differences.

Whether or not the IT architecture principles are specified well and have a high or low quality level, the declarative statements of most principles are identifiable in the implemented IT system. However, for some principles, the description was too poor, which is the reason that we could not identify the directives of the principles in the usage stage.

Looking at the four case studies, we might conclude that, whether or not good described, using IT architecture principles in itself is useful in implementing IS requirements. This is because of the fact that the IS implementation of those four case studies was successful. However, we didn't investigate the successful implementation of the IS requirements in detail. So the contribution of IT architecture principles to the successful implementation is still a question. This hypothesis is the topic of our further research: the results of the four case studies provide us with some indications and directions, but are only the beginning of a more extensive study into the practical use of architecture principles in the successful implementation of IS requirements and therefore useful in the digital transformation.

Table 14.9 Overview of the individual scores of the IT architecture principles

Nr.	IT architecture principle	IT system	Specification	Quality	Usage	Classification
1	Data on separate server	MON	0.5	1.0	1.0	Infrastructure
2	No relationships between MON and other systems	MON	0.5	0.8	1.0	Generic
3	The application landscape has to be simplified	DOBRA	0.7	0.8	1.0	Application
4	Data have to be used through the Custom organization	DOBRA	1.0	0.9	1.0	Generic
5	Use the application as meant to be	DOBRA	0.5	0.8	1.0	Application
6	Good is good enough	DOBRA	0.0	0.4	0.0	Generic
7	Application is flexible	DOBRA	1.0	0.5	0.5	Application
8	Interfaces between applications are flexible	DOBRA	0.5	0.5	0.5	Application
9	Integrate what have to be, separate what can	DOBRA	1.0	0.9	0.5	Generic
10	Use the standard technical infrastructure of B/CIE	DOBRA	1.0	0.9	1.0	Infrastructure
11	Reuse, before buy, before build	PDO	0.5	0.8	0.5	Generic
12	Forms should be added easily to the application	PDO	0.8	0.6	1.0	Application
13	PDO is legacy	PDO	1.0	0.9	1.0	Generic
14	Only connection with the current architecture and processes	PDO	1.0	0.9	1.0	Generic
15	PDO will be transformed to MBD, but meanwhile PDO will be available for companies to use	PDO	1.0	0.9	1.0	Generic
16	Figurative use of PDO is not allowed	PDO	0.7	0.3	0.5	Generic
17	No manual input of BAG files through the office portal	PBT	0.3	0.8	1.0	Generic
18	Input of BAG files by using SQL server	PBT	0.7	0.8	1.0	Infrastructure
19	During the reading of BAG files into the application no data processing	PBT	0.7	0.8	1.0	Generic
20	Data contracts are recorded to be able to query specific data elements	PBT	0.7	0.8	1.0	Generic
21	Connection strings to SQL server are configurable outside the source code	PBT	1.0	1.0	1.0	Application
22	Location of logging is configurable outside the source code	PBT	1.0	1.0	1.0	Application
23	Correctness and integrity of the data is guaranteed	PBT	0.5	0.8	1.0	Generic
24	Reuse of software components	PBT	0.7	0.6	1.0	Application
25	Maintenance, release and distribution of singular parameters and domain tables are rebuilt into two generic solutions	PBT	0.8	0.9	1.0	Generic
26	The PBT entity "*ParameterWaarde*" is loaded with actual production parameter values once only, using an installation script.	PBT	0.5	0.7	1.0	Application
27	The PBT meta-model is loaded using an installation script	PBT	0.5	0.7	1.0	Application
28	The PBT meta-model doesn't support plural norms	PBT	0.5	0.7	1.0	Generic
29	The component can't use the parameter meta-model	PBT	0.5	0.7	1.0	Application
30	A PBT user has the role of 'parameterbeheerder' or the role of 'parametervrijgever'.	PBT	0.3	0.5	1.0	Generic
31	MD5 is used for the cryptographic hash function	PBT	0.8	1.0	1.0	Generic
32	One basis function for calculating hash, ownership PBT, but the hash function have to be reusable for the total application landscape	PBT	0.7	0.8	1.0	Application
33	One basis function for verifying hash, ownership PBT, but the hash function have to be reusable for the total application landscape	PBT	0.7	0.8	1.0	Application
34	PBT user interface should be conform the "*Style guide Kantoorportaal*"	PBT	0.8	1.0	1.0	Application
35	User authorization and authentication in PBT using KPTL UC010 "*Verkrijgen toegang tot Kantoor-portaal*"	PBT	0.8	1.0	1.0	Generic
36	The content of the PBT entity "*ParameterWaarde*" is replicated automatically to the component entity "*ParamterWaarde*" and FiBi using the snapshot-replication	PBT	0.8	1.0	1.0	Application
37	The PBT is responsible for the creation of the component entity "*Parameter-Waarde*"	PBT	0.3	0.8	1.0	Application
38	The component entity "*ParameterWaarde*" can be loaded exclusively by the PBT	PBT	0.3	0.8	1.0	Application
39	Within the scope of this release the distribution of the content of the PBT entity "*ParameterWaarde*" exclusively per component to the 8 TSL-services and FiBi (the components)	PBT	0.5	0.8	1.0	Application
40	Distribution of parameter values to FiBi is always part of the release process of parameter values	PBT	0.7	0.8	1.0	Application
41	The PBT component doesn't support tailor made distribution of a parameter to different components	PBT	0.7	0.8	1.0	Application
42	Distribution of the PBT entity ParameterWaarde content to a specific TSL service using a config-file.	PBT	0.7	0.8	1.0	Application
43	Audit trail PBT and the component entity "*ParameterWaarde*" (and FiBi) using Microsoft SQL Server 2008 R2	PBT	0.7	0.8	1.0	Infrastructure
44	When a parameter is not available, an exception occurs at the component which signals the unavailability of the parameter.	PBT	0.7	0.8	1.0	Application

Chapter 15
The Need for New Architectural Truths

Marlies van Steenbergen

Abstract The valorization of technological innovation depends on the way enterprises use it to offer value propositions to their clients. To keep offering value, enterprises have to continuously adopt new emerging technological opportunities. As the enterprise architecture discipline promises to enable the transition from strategy to execution, it should support enterprises by enabling them to do just that: keep absorbing new technologies to offer value to clients. However, traditionally, many enterprise architecture disciplines have been geared toward internal efficiency instead of toward continuous value delivery to customers. This chapter argues that some long-standing architectural truths are no longer universally valid and that rethinking of these truths leads to fundamentally different enterprise architectures.

15.1 Introduction

Digital transformation is a prerequisite for enterprises to be successful in today's digital world. Enterprises can only keep up when they are able to make full use of digital possibilities. They need it in order to understand their customers as well as to serve their customers (Catlin et al., 2014). Success is created by being part of a (digital) ecosystem that is constantly renewing itself to keep delivering value (Keen & Williams, 2013; Pagani, 2013). The ability of an enterprise to actively contribute value within this ecosystem is transparent to anyone participating in the ecosystem. If no value is delivered, the enterprise is marginalized.

To actively contribute value, enterprises need to be able to make sensible use of the newest technological possibilities (Ross et al., 2015). Otherwise, it is like delivering goods to clients by horse and wagon. No enterprise can sustain the use of obsolete technologies for any amount of time without making itself irrelevant.

M. van Steenbergen (✉)
Utrecht University of Applied Science, Utrecht, the Netherlands
e-mail: marlies.vansteenbergen@hu.nl

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_15

In this chapter, we argue that the technological and societal changes that are happening today represent a fundamental change in the playing field of enterprises and require fundamental changes in the enterprise architecture discipline as well. Because of the combination of technological innovation and social adoption, drivers, constraints, and roles are changing, affecting the foundation of enterprise architecture. The combination of speed and diversity generates complexity. Complexity implies unpredictability (Kurtz & Snowden, 2003). Enterprise architects have to reconsider their hard-earned truths and operational modes for suitability in the new playing field (Lapalme et al., 2016).

In Sect. 15.2, we explore the changes in context that are relevant to the enterprise architect and argue that these changes lead to new drivers and constraints for enterprise architecture, impacting its foundation. Section 15.3 discusses how this new context leads to a different focus for the enterprise architect. A change in focus implies a change in architectural content as well, especially the architecture principles, as is described in Sect. 15.4. In addition, the enterprise architect must change his behavior to stay relevant. This is the topic of Sect. 15.5. In Sect. 15.6, we draw conclusions and make some recommendations.

15.2 Changing Circumstances

What makes today's change so different from changes in the past years that it requires a fundamental rethink of the enterprise architecture discipline? The answer to this question lies in the fact that today's changes concern the drivers and constraints for enterprise architecture. The enterprise architecture is the result of choices made about the fundamental organization of the enterprise and the principles guiding its design and evolution. Enterprise architecture is successful if it enables the enterprise to realize its ambitions. The choices made in the enterprise architecture are based on the strategic and tactical enterprise drivers on the one hand and technological and societal constraints and possibilities on the other hand (Greefhorst & Proper, 2011b). If possibilities change, this may lead to adaptations of the architectural principles. Usually, these adaptations are limited to one or two principles, and they are often concerned with the implementation part of the principles. For instance, the availability of the cloud on its own led enterprises to reformulate their principles regarding sourcing. This led to a change in the implementation details of the architecture. But in many cases, it did not lead to a fundamental change in the enterprise architecture, as many organizations adopted cloud without having a clear digital strategy (Goutas et al., 2015). The drivers that give the enterprise architecture its direction, such as cost reduction and efficiency, did not change. The motivation and rationalization of the architectural principles remained the same.

However, if drivers or constraints change, usually because of a combination of innovative developments, it impacts the enterprise architecture as a whole. And that is what is happening today (Bonchek & France, 2015; Ross et al., 2015, 2016). A change in drivers and constraints represents a change, not merely in the implementation of the architecture but in the very foundation of the architecture. If such a change goes unnoticed by the architects, the architecture will be based on the wrong foundation. It is therefore essential that architects realize this change in their environment and seriously reflect on the underlying assumptions of their current architecture.

We notice the following changes in drivers:

- *Customer centricity*—Enterprises are more and more putting the customer in the center (Keen & Williams, 2013). Large organizations are introducing the role of customer journey manager or customer journey expert. Processes become increasingly centered around the customer, instead of around a product or service. Offerings and interactions are increasingly tuned to the specific needs of the individual customer (Bonchek & France, 2015; Ross et al., 2016).
- *Data-driven service innovation*—By making use of all kinds of data, both internal and external, enterprises can develop new services (Ross et al., 2016; van der Aalst, 2014). As with all new possibilities, customers will soon start to expect that all relevant information is incorporated in services, leading to services that are always spot-on. Anything less will not be accepted for long. This leads to a new marketplace of data, enterprises offering their own data on the one hand and using external data from various sources on the other hand. Data is getting a new meaning, with accompanying rules of the game. It certainly is no longer only about structured data in the enterprises' own systems of record (Ross et al., 2015; Erevelles et al., 2015).
- *Collaborative value generation*—Value is not generated by enterprises on their own, but in cooperation with other enterprises (Keen & Williams, 2013; Pagani, 2013; Guédria et al., 2013). This means that the boundaries of the enterprise are becoming increasingly fluent. Building an architecture on these boundaries is not wise. Instead, the architecture should become organization-agnostic, allowing for flexible sourcing. In addition, an architecture that concerns the ecosystem instead of an individual organization may be in order.

Apart from new drivers, architects must also be aware of changing constraints:

- *Infinite bandwidth*—Not a reality yet, but sometime in the future, bandwidth will no longer be a constraint (Gommans et al., 2006). This has great consequences for the way we think of distribution of data and processing power. An important issue at the moment is the difference in regulation between countries which makes it hard for enterprises to put their data in the cloud, without knowing where it is physically stored. With infinite bandwidth, enterprises can store their data within their own control, at their own location, which becomes increasingly easy because it takes increasingly less space, and have the processing done in the cloud. This is possible, because storage space, processing power, and connection are becoming less scarce. This allows for completely new business models and the discarding of many IT principles motivated by scarcity of resources.

- *Blockchain*—Blockchain technology is rapidly gaining attention. Its promise is to diminish the need for a trusted third party, thus enabling new business models. The blockchain has moved from being the underlying technology for cryptocurrency to being a new model to securely and transparently record transactions between agents, either human or robotic. It also provides a means for consumers to stay in control of their own personal data and whom they want to share it with.

But not only drivers and constraints have changed; the players are changing as well. With the event of agile, roles have become redefined. Responsibilities shift from individuals to teams. All employees are expected to contribute directly to the value creation. This is not done by merely producing knowledge, but by applying that knowledge in a meaningful way, within the context of a team effort.

All of these changes in context together have a profound impact on the enterprise architecture discipline: (1) It requires a shift in focus from the architect, (2) it can only be supported with new architecture principles, and (3) it asks for different behaviors.

15.3 From Internal Efficiency to Customer Centricity

The changes in drivers and constraints require a shift in focus from the architect. Traditionally, architects were expected to ensure internal efficiency, which led to a focus on standardization, uniformity, cost reduction, etc. It led to an inside-out approach of architecture. This internal focus is evident from an extensive literature study by Boucharas et al. (2010) and is supported by a survey by Plessius et al. (2014). The literature study by Boucharas et al. (2010) showed that of 100 benefits found in the scientific literature, only 2 were related to the customer perspective. The survey executed by Plessius et al. (2014), asking for the benefits employees experienced from having an enterprise architecture function within their organization, showed a remarkable low score on experienced benefits from the customer perspective.

In the survey, respondents were asked to what extent they perceived that, in their organizations, enterprise architecture contributes to the achievement of the organizational goals. The goals were categorized into the well-known perspectives from the balanced scorecard: financial, customer, internal, and learning and growth (Kaplan & Norton, 1992). From these perspectives, the customer perspective received the lowest scores. Less than 50% of the respondents held the opinion that EA has a positive effect on the organization's interaction with customers and the market. The shift in drivers requires a shift in focus to this customer perspective. Architects need an outside-in perspective, focusing on customer value, rather than internal efficiency.

15.4 New Architecture Principles

The new foundation from which enterprise architects must work not only leads to a new focus but also to new types of architecture principles. Based on personal experience of the author with the architecture principles of various enterprises in the Netherlands, a number of principles keep reappearing that are motivated by drivers such as complexity reduction, uniformity, reuse, and business/IT alignment. They seem to be part of the common knowledge around architecture principles. They include principles such as:

- *Reuse before buy before build*—this principle is motivated by efficiency and cost drivers. It is thought to be more efficient and less costly to use something that already exists, than to build it from scratch. Its implementation, however, frequently leads to forced use of IT systems that do not fit the bill.
- *No redundancy*—the fear for redundancy, especially as data are concerned, originates from the primacy of structured transactional data. Redundancy in this kind of data often leads to low efficiency because data can become contradictory and it is hard to determine the truth. There is no single truth in the applications of the enterprise. However, with the rise of unstructured data, and the use of non-proprietary data, the issue of redundancy is unavoidable. And therefore, we should think of other ways of ensuring data quality.
- *Standardization*—standardization occurs in many variations: from restriction to a specific technology stack to limitation to specific prescribed vendors. The driver, again, is efficiency and cost reduction. However, too strict adherence to standardization of technology or vendor may prevent an enterprise from innovation. Also, it may hinder free movement of the enterprise within the ecosystem. It may therefore be better for the architect to think about how to deal with heterogeneity.

Variations of these principles are also found in the principle catalogue provided in Greefhorst and Proper (2011b). In this catalogue, 59 principles are listed that are harvested from real-world architectures and are intended as a source of inspiration for practitioners in the field. For each of the 59 principles, the quality attributes are given that are positively influenced by the principle. Six quality attributes are mentioned: maintainability, portability, usability, efficiency, reliability, and functionality. From these, maintainability and efficiency occur most frequently (being positively influenced by around 50% of the principles), while usability and functionality occur least frequently (being positively influenced by around 20% of the principles).

With the change in drivers, however, the need arises for other principles, addressing themes such as:

- *Design without enterprise boundaries*—enterprises deliver value in the context of an ecosystem of partners, competitors, and clients. They cannot longer do it on their own. Enterprise boundaries are losing their fixedness and instead are becoming fluent (Keen & Williams, 2013). Also, enterprises come and go. Analysis over the past shows that the lifespan of enterprises is reducing rapidly (Anthony et al., 2016).

- *Design for use*—the driver for customer value leads to principles focused on delivering products and services that are fit for use. From the value perspective, fit for use, even in a single circumstance, is more important than having a service that is reusable, but does not serve anyone well. The latter will not be used at all, let alone reused. Services that are fit for use, on the other hand, will be used by many, simply because they are handy.
- *Standardization on interfaces*—though standardization on technology may hamper innovation, standardization on interfaces stimulates it. Standardization on interfaces enables enterprises to use each other's services. It stimulates collaboration and interaction. It enables rapid innovation by combining existing building blocks in new ways. While standardization on technology is driven by efficiency, standardization on interfaces is driven by value delivery.
- *Automation*—the driver of delivering customer value requires fast delivery of new features and services. Based on evidence gathered from available data, a digital enterprise develops an extreme automation capability to be able to do this.

Table 15.1 presents a first step toward formulating principles that address these themes. They need further tuning, but they give an indication of what kind of principles organizations may want to adopt.

These are just a few examples of new principles that might better suit a digital enterprise. The main message here is that organizations that undergo a digital transformation are in need of new architecture principles in alignment with the new digital ambitions of the organization.

15.5 Just-Enough, Just-in-Time Architecture

Not only the content of architecture principles is due to change but also the manner in which they are applied by the organization.

Kurtz and Snowden (2003) distinguish between various problem-solving domains in their Cynefin framework:

- *Known*—Cause and effect relations are repeatable, perceivable, and predictable. The decision model in this domain is based on sense-categorize-respond.
- *Knowable*—Cause and effect are separated over time and space. The relation between cause and effect is not as clear-cut as in the known domain, but given time and resources, they can be discovered. The decision model in this domain is based on sense-analyze-respond.
- *Complex*—Cause and effect are only coherent in retrospect and do not repeat. Emergent patterns can be perceived but not predicted. The methods, tools, and techniques of the known and knowable domains do not work here. The decision model is based on probe-sense-respond.

Table 15.1 A new breed of principles

Design without enterprise boundaries	Design choices are aimed at participation in an ecosystem with products and services from various partners maximising the customer value	
	Organisational boundaries are becoming increasingly less fixed and important. Designs should cater for the possibility that any business capability could be executed by any party.	
		Motivation - This is the essence of a platform organisation. - Enables agility
		Implications - Design in terms of independent capabilities - Focus is on interoperability between capabilities, not on internal operation of the capabilities - Continuous weighing of inhouse execution or not
Design for use	The organisation is designed in terms of well-defined business capabilities with a bounded, singular function that supports an actual need	
	Explicitly designing systems for (re)use by many, in practice often leads to complex systems that contain all kinds of compromises, do not make anyone happy and are not maintainable (the second-system effect or feature creep effect). *A well-defined, focussed business capability with a clear purpose, however, can and will be reused because of its inherent usability. Designing for use, therefore, automatically leads to reuse.*	
		Motivation - Fit for purpose. - Speed of development
		Implications - Do not include too much functionality in a single capability. - Cater for orchestration of capabilities.
Standardisation on interfaces	The focus of standardisation is on the interaction between capabilities, not on the technology used within capabilities.	
	Technologies come and go, but interfaces are here to stay. It is all about interoperability. Adoption of new technology is part of being able to innovate fast. Standardisation on technology may lead to inertia. However, inability to interoperate with other capabilities may hamper real innovation.	
		Motivation - To be able to innovate fast enough, new capabilities must be plugged into the existing ecosystem without unnecessary hassle. - Enables optimal use of technological innovation.
		Implications - Interoperability standards must be rigorously adhered to. - Use of APIs.
Automation	All processes from demand to delivery of products and services are radically automated, enabling rapid adaption to customer needs.	
	The ability to build temporary solutions with a clear focus on consistent master data and service interfaces, will allow the customer touch points (webs, apps, bots, things, etc) to move faster than what is possible in the back-end systems, and even external services. Automated builds, tests, and deployments/distribution will increase quality while accelerating.	
		Motivation - Increase in speed and quality of delivering customer value.
		Implications - The architecture is designed for automation.

- *Chaos*—No cause and effect relationships perceivable. The chaotic domain is in a very real sense uncanny, in that there is a potential for order but few can see it. The decision model is based on act-sense-respond.

Besides these ordered (known and knowable) and un-ordered (complex and chaos) domains, there is a fifth domain: the domain of disorder. This domain reflects the fact that individuals tend to pull issues to the domain they feel most empowered by. If this central area of disorder is large, there are a lot of conflicts between decision-makers about how to tackle important issues.

It can be argued that architecture must be able to deal with complex situations, as well as known and knowable situations. The rapid succession of technological innovations and the increased application of data analytics and machine learning make it hard to predict the course of technological development. But even in the knowable domain, it is hard to capture beforehand all possible issues in a predefined set of detailed rules, especially where interaction with customers and partners is concerned.

Refactoring should therefore be a fundamental part of the architecture. It is necessary to constantly refactor both the way we think, work, and interact and the architecture, technical solutions, and code. The only way to not get stuck in routines and processes that do not work and avoid building technical debt is to constantly question these things. This means, among other things, that architectural choices are evaluated for their retractability, that architectural choices are made at the latest possible moment instead of the first possible moment, and that by rule some architectural choices are provisional.

Complex problems in the sense of Kurtz and Snowden (2003) cannot be successfully addressed by a predefined set of detailed rules, simply because the correct set of rules cannot be known beforehand. The architect cannot produce detailed blueprints, but instead becomes part of the day-to-day decision-making process (Poort & van Vliet, 2011). To be able to make architecture decisions on the fly, the architect will want to have some essential models to use as a framework for his decisions. However, these may be different from the detailed enterprise-wide models that many architects are used to make. Whereas detailed enterprise-wide models may not be very effective in a complex world, solution patterns for particular types of problems may be very useful.

The more detailed the architecture principles, the lesser the room for interpretation. This works well in an ordered, predictable environment. However, in a complex environment, it is impossible to predict all possible situations. Hence, it is also impossible to define rules for every possible situation. In that case, it is better to limit the architecture to a small set of principles that express the why and what rather than the how. The distinction between rules and principles has been discussed extensively in the literature on regulatory compliance (Black et al., 2007; Black, 2008; Burgemeestre et al., 2009). The distinction between principles and rules is a continuum. Burgemeestre et al. (2009) use seven dimensions to determine the primary nature of a directive framework, i.e., is it primarily rule-based or primarily principle-based. These seem to be applicable to architecture as well. The dimensions (translated into an architecture context) are the following (Eusterbrock & van Steenbergen, 2016):

- *Conceptual dimension*—principles are characterized as general, universal, and abstract, whereas rules are specific, particular, and concrete.
- *Representation dimension*—principles are declarative representations, expressing desired outcomes. The amount of documents is relatively small. Rules are procedural descriptions, expressing by what actions an outcome should be achieved. This usually leads to a relatively large set of documents containing rules and patterns for different contexts.
- *Functional dimension*—this dimension deals with the discretionary powers of the participants in the architecture process. In a rule-based architecture, the rules are defined centrally by the architect and just to be followed by the designers. There is not much room for interpretation. The principles of a principle-based architecture leave room for interpretation to both designers and (controlling) architects: the exact way in which the principle is implemented is left to the discretion of the designers.
- *Knowledge dimension*—applying rules requires relatively little knowledge from the applicant because everything is spelled out in the rules. Knowledge of the rule itself and the concepts used to implement the rule suffices. Applying principles requires more knowledge, such as knowledge of the context in which the principle is applied and other relevant principles.
- *Temporal dimension*—rules define the boundaries of a design in detail; *before* their translation to a specific design and implementation, principles provide broader boundaries; and compliance of a specific solution is *audited after* the design is made. From another temporal perspective, principles are longer lived than rules. Rules need to be changed each time fundamental changes in the design approach are required.
- *Exception handling dimension*—rules do not allow for exceptions in the implementation, they are strict, and every implementation must adhere to the rule. Principles do not prescribe precisely how things are to be done, but point in a certain direction. Therefore, they lead to a form of reasoning and may lead to different implementations, i.e., exceptions may occur without the principle becoming invalid.
- *Conflict resolution dimension*—principles and rules need different processes to solve potential conflicts about their application. For principles, different conceptual explanations can occur of how they are to be applied in design and implementation. There must be a process in place to address conflicts that may occur between the authority that is responsible for the principle or rule and the team that uses it. This process needs to weigh the various perspectives based on some kind of priority, order, or weight. Sometimes, the principle itself can be under discussion. For rules, no conceptual conflicts are possible. If a team is breaking a rule, an escalation ensues, in which either the team is forced to adhere to the rule or deviation is allowed. The choice is often primarily based on cost and benefits.

Architectural rules and principles are a means to share knowledge in the organization. Rules are well-suited to express specific and detailed knowledge. Principles are better suited to express intentions and desired outcomes. Grant (1996) distinguishes four mechanisms for integrating knowledge: rules and directives, sequencing, routines, and group problem-solving and decision-making. Grant argues that rules and directives (i.e., written-down directions) are suitable for communicating explicit knowledge among specialists and between specialists and non-specialists. Rules and directives are useful for tasks that are well-defined and to a great extent predictable. Group problem-solving and decision-making requires active interaction between participants and is needed for non-standardized tasks that are complex and unpredictable. Translating this to principles and rules, the more detailed knowledge expressed by rules is comparable to rules and directives and thus is useful for providing direction to relatively standard, predictable tasks. Principles, which are less precise, indicating desired outcome rather than the means to achieve that outcome, require more interaction and discussion in their application. They are the first choice when the organization has to deal with complex and unpredictable tasks.

All of this does not mean that everything we did is wrong and should be abandoned. It does mean, however, that we must consider whether the things we did before are still applicable in all situations. And the answer will be no. Referring to the domains of the Cynefin framework of Kurtz and Snowden (2003), the known and knowable domains may need a different architectural regime than the complex domain. For the complex domain, a new regime is required: a regime with new principles, as discussed above, but also with new ways of working and new distributions of responsibility.

15.6 Conclusion

Digital transformation requires a new enterprise architecture approach because it fundamentally changes the strategy of enterprises and as a consequence the foundation of architecture. Both drivers and constraints are changing rapidly, causing the need to rethink the architectural principles. The focus of architects should shift from internal efficiency to delivering customer value. At least for the moment, there may be a need to differentiate between various architecture regimes within the enterprise, allowing for differences in nature between various parts of the enterprise.

Acknowledgments I would like to thank Piet Jan Baarda, Wijtze de Boer, and Ton Eusterbrock for their discussions about the new architectural principles.

Chapter 16
Presentation and Validation of an Integrative Approach for Digital Transformation Evaluation

Tamara Högler, Johan Versendaal, and Ronald Batenburg

Abstract Information and communication technologies (ICT) are the engine of competitiveness and innovation of the twenty-first century. The advances in ICT specifically enable digital transformation in (extended) enterprises. Yet, successfully leveraging ICT is a complex challenge, particularly when mobile technology with its many advantages and specific singularities is involved. Careful preparation before development and implementation is needed. We present an integrative framework that supports organizations in preparing their mobile technology-enabled digital transformation. It consists of seven activities that are categorized by three pillars of the framework: (1) *internal analysis*, (2) *economic analysis*, and (3) *integrative analysis*. We demonstrate the validity and applicability of the framework through a multimethod validation. First, by a retrospective case study about the introduction of a mobile app in a hospital, we show that most aspects of the framework are relevant and covered. Second, by qualitative semi-structured expert interviews, we conclude that the need for each of the activities of the framework is underwritten. Third, by successfully applying one key activity of the framework—definition of the target system—on the implementation of an electronic learning environment at a fire brigade.

T. Högler (✉)
Independent consultant, Karlsruhe, Baden-Württemberg, Germany
e-mail: Tamara@hoegler.de

J. Versendaal
Utrecht University of Applied Science, Utrecht, the Netherlands
e-mail: johan.versendaal@hu.nl

R. Batenburg
Radboud University Nijmegen, Nijmegen, the Netherlands
e-mail: r.batenburg@nivel.nl

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_16

16.1 Introduction

Mobile technologies are reshaping the global economic landscape, enhancing speed and comfort of communication and information exchange. They aim at integrating mobile processes and devices into the enterprise architecture to overcome spatial separation and accompanying information losses. If information becomes available any time at any place (cf. Schiller, 2003; Isaac & Leclercq, 2006), it can make digital transformation happen. We define mobile systems as "sets of mobile technology and human (system) components which are inherently related" (Högler, 2012). Mobile systems have a multiplicity of singularities,[1] which make them specific as compared to stationary information and communication systems (ICS). The challenge to take this into account in an evaluation makes mobile systems an interesting object of investigation.

While ICS in general and mobile systems in particular have the potential of being of great benefit for organizations, their implementation often fails in terms of scope, budget, and time. Addressing and reaching business objectives before implementation appears to be difficult or is failing (The Standish Group International, 2016). An effective a priori evaluation of ICS implementation (e.g., to decide whether or not including mobile components) will increase understanding the potential and the effects of its implementations, and knowing the success factors. Existing approaches for this a priori evaluation of ICS and mobile systems, however, are limited because:

- They often only consider monetary effects or, when including other qualitative effects, transform those into monetary effects as well (Horváth, 1988; Zahn et al., 1999; Mutschler, 2005)
- Evaluation methods lack a theoretical basis (Renkema & Berghout, 1997; Berghout et al., 2011)
- Existing evaluation approaches do not explicitly address singularities of mobile systems (Högler & Versendaal, 2014)
- Interdependencies seem insufficiently being taken care of in existing evaluation approaches, particularly interdependencies between objectives, risks, and success factors, which eventually affect implementation effects (Högler, 2012)
- They fail an overall integrative framework

According to many researchers and consulting companies, an improper requirement definition is the most-cited reason for implementation failures (Davis et al., 2006; Hughes et al., 2015). In fact, "the lack of clear understanding of what the company wants to achieve" (IMG, 2015) avoids that many of the requirements are not derived from the goals that should be achieved by implementing an ICS. This calls the importance of a profound "target system" definition in an a priori evaluation. Both the omission of a profound target system and a socio-technical

[1] Högler and Versendaal (2014) have provided in their work a detailed list of singularities of mobile systems.

view (Orlikowski, 2000) on mobile system implementation have motivated the development of an integrative framework that is presented in this chapter. In the next sections, we describe this framework and how it can be applied to conduct a valid and reliable a priori evaluation of mobile systems that enable digital transformation. The following sections present the framework, followed by three types of justification and validation of the framework. In the remainder of this chapter, we will reflect on the construction of the integrative evaluation framework and address the identified strengths and limitations.

16.2 The Integrative Evaluation Framework for Digital Transformation Evaluation

For the definition of an integrative framework, we follow design science research guidelines (Hevner et al., 2004; Peffers et al., 2007). The proposed artifact is our integrative framework, and its construction considers both the scientific body of knowledge and the business needs. In designing our framework, we draw principles from several theoretical foundations. Principles are drawn from system theory (Bertalanffy, 1979) and from strategic alignment, business process engineering, and digital strategy models. The latter have their roots in the work of Solow (1987) and Loveman (1994) who early on identified that ICT implementations do not necessarily lead to increased firm productivity. Labelled as the "productivity paradox" by Brynjolfsson (1993), notably Henderson and Venkatraman (1993) developed the strategic alignment model (SAM). The key claim is that strategic fit (degree of alignment between firm strategy and firm operations) and functional integration (degree of alignment between IT and the rest of the business) should be taken into account in order to gain the proposed benefit from ICT. At the same time, the famous article by Hammer (1990), "Reengineering Work: Don't Automate, Obliterate," set the stage for the business processes reengineering (BPR) through ICT. The literature has since been consolidated by, e.g., Hammer and Champy (1993) and Peppard and Ward (2016). Looking back, one could say that BPR is part of the operationalization of Henderson and Venkatraman (1993)'s SAM. Also, one can well argue that Peppard and Ward (2016, pp. 108–110) operationalized SAM by defining their digital strategy model (DSM), taking the external business environment, the internal business environment, the external IS/IT environment, and the internal IS/IT environment as inputs for the identification of the future ICS-application portfolio.

Based on these different theories and models, we define a first set of six basic principles for the design of our framework for digital transformation evaluation:

1. Start from a holistic approach, taking into account economic, technical, and social aspects in the a priori ICS evaluation.
2. Focus on interdependencies and mutual relationships; changes in one part effects other parts, taking into account (or not taking into account) certain success factors

may limit or increase likelihood of risks, striving for certain business objectives with system implementation may strengthen or weaken other business objectives.
3. Explicitly take into account the users of ICS.
4. Align ICS system implementation with processes, with business strategy, and with existing IT infrastructure.
5. Consider both the internal business environment (processes, [business] objectives) and the external business environment (the economic, industry, social, regulatory, and competitive climate).
6. Consider both the internal (existing) IS/IT environment, including the existing running applications, and the external IS/IT environment (trends in IT, competitor's IT infrastructure, etc.).

The second set of six principles is specific for the evaluation of mobile systems and based on earlier work (Högler, 2008, 2014):

1. Deal with the specific characteristics of mobile systems (singularities), like a need for Wi-Fi access, limited power supply, and the need for hands-free working (Högler, 2014).
2. Pay attention to business objectives, benefits, and costs.
3. Orientate on the life cycle of a system; e.g., a particular effect is not only taking place at a specific point of time they occur but according to the space of time they take effect.
4. Address multidimensionality; not only costs and benefits that are monetary measurable but also other beneficial aspects should be taken into account.
5. Address situationality; each project has its own context (e.g., sector, project size, implementation time), and therefore, the framework provides room for extending and scaling.
6. Consider critical success factors in respect to volatility effects.

The 12 principles can be categorized into 3 pillars for the proposed integrative framework for digital transformation evaluation: (1) a detailed enterprise internal evaluation, (2) a detailed economic evaluation, and (3) an integrative evaluation. These three pillars, key principles, are broken down into sets of activities, as depicted in Fig. 16.1, and elaborated in Tables 16.1, 16.2, and 16.3. In further detail:

- Definition of the target system (activity 1: A1) follows the methods for multi-attribute decision-making Hwang and Yoon (1981) and leverages the analytic hierarchy process (AHP) (Saaty, 1996). One of the main contributions of this work is that the AHP is extended and applied in the context of an integrative approach for ex ante evaluating the economic efficiency of mobile systems. The uniqueness of the extended AHP is that the determination of priorities is not based on subjective assessment (like the opinion of decision-makers), but on a preference-neutral approach that contains the following three steps:

    1. Interdependence analysis between individual goals (Kirchmer, 1999; Drews & Hillebrand, 2002; Rückle & Behn, 2007)

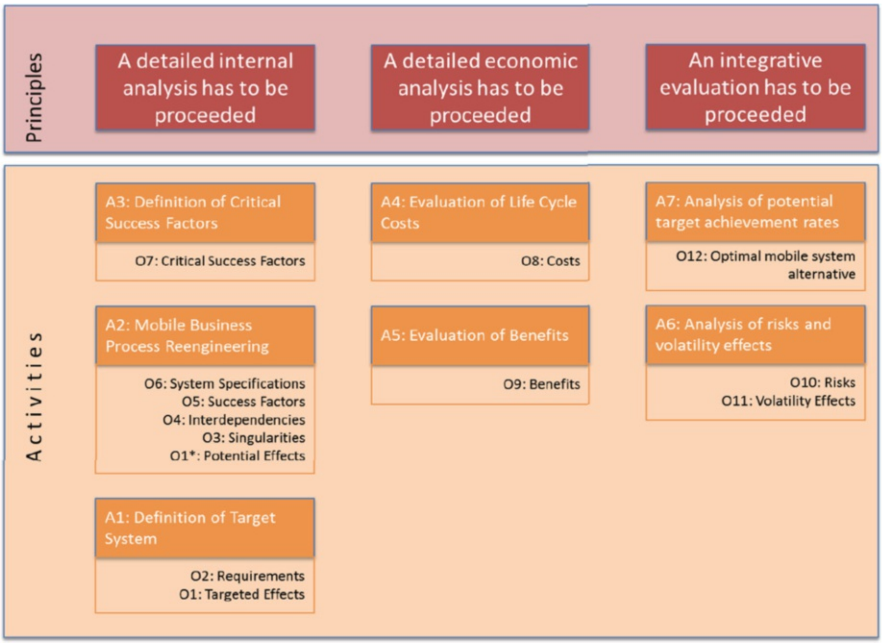


Fig. 16.1 Integrative framework

2. Consideration of the effective strength of objectives and the probability of occurrence of interdependencies (Klabon, 2007; Charette, 1991)
3. Preference-neutral weighting of objectives in the context of steps 1 and 2

The outputs of this activity include prioritized monetary and qualitative effects to be achieved by the implementation of a mobile system (output 1: O1) and general requirements (O2) of a mobile system.

- Mobile Business Process Reengineering (mBPR, activity 2) is aimed at analyzing and documenting existing processes with a specific focus on their mobile parts. The resulting process models can include—besides a detailed description of how and where operations are conducted—information regarding the employed data, IT resources, and other artifacts like KPIs and responsibilities (Scheer, 2000; Recker, 2009). We leverage the systematic approach as defined by Peppard & Ward (2016, p. 158), where BPR is not aiming—as called for in the early days of BPR—on a fundamental rethinking of the company and its business processes, but rather at optimizing existing (mobile) business processes using mobile technologies. In the framework, singularities (O3) and interdependencies (O4) of the mobile system are derived from this activity. During the mBPR, requirements (O2, from activity 1) are turned into system specifications (O6). Additional requirements can arise during activity 2, leading to further system specifications. Based on the requirements and specifications, general success factors (O5) for a special type of mobile system can be derived in this activity.

Table 16.1 First 3 activities of the integrative framework and their relevance for the 12 principles

	Definition of target system	Mobile business process reengineering	Definition of critical success factors
Principle 1 (holistic approach)	-	Economic, technical and social aspects are considered within the mBPR (e.g. who are the users / responsibilities, what are the current KPIs, what kind of technology is / will be used….)	This activity considers the specific environment and singularities of a project and thus takes a holistic perspective.
Principle 2 (interdependencies)	Considers interdependencies between objectives	Considers interdependencies between components of ICS (processes, users and technologies (existing infrastructure, planned infrastructure))	CSF are analyzed also regarding their interrelationships, i.e. how does CSF A affect CSF B?
Principle 3 (users)	Whenever possible, different kinds of users (white, blue collars…) should be involved in the definition of the target system	Identifies groups of users and analyses their activities, responsibilities, tasks, behavior, technical affinity, …)	Process of defining CSF considers also users and their affinity towards IT, knowledge, willingness to use IT, ….
Principle 4 (alignment)	-	ICT in alignment with (non-) mobile processes	CSF consider appropriateness of potential ICS to reach objectives
Principle 5 (business environment)	Considers objectives that have to be achieved with the project, also strategic and thus also business environment	Considers internal and external business environment (e.g. necessary changes in processes to achieve objectives; willingness of users to change kind of performing tasks / to adopt new technologies, ….)	Business environment as basis of defining CSF
Principle 6 (IS/IT environment)	Considers also external IS/IT environment	Considers internal and external IS/IT environment (analysis of existing technical infrastructure, current trends in similar / competing businesses, state of the art of technology, …)	IS/IT environment as basis of defining CSF
Principle 7 (singularities)	-	During mBPR singularities of the specific project environment are identified	Singularities can be taken care of by identifying them as critical success factors
Principle 8 (objectives, benefits, costs)	Defines objectives and serves as basis for evaluation of benefits & costs connected to achieving defined objectives	During mBPR it is already analysed if objectives can be achieved in general or if changes regarding the environment are necessary (e.g. replacement of users, additional / change of technical infrastructure, …)	CSF focus on achieving objectives (benefits) with a given budget (costs)
Principle 9 (life cycle)	-	mBPR identifies where and when a specific effect is occurring / will occur	Considers life cycle as regards to costs and timeframe in which benefits should be achieved
Principle 10 (multi-dimensionality)	All objectives, also strategic ones, are considered	mBPR, as suggested, considers different aspects (e.g. users, costs, benefits, singularities) and thus takes a multi-dimensional perspective.	CSF derived from different aspects (costs, benefits, environment, singularities) and follow thus multi-dimensional approach
Principle 11 (situationality)	-	mBPR considers singularities, overall environment and thus the situationality of a specific project	CSF are derived from situationality
Principle 12 (CSF)	Contributes to proper definition of CSF, based on requirements and objectives	CSFs can be derived from the findings gained during mBPR	Defines CSF

Table 16.2 Activities 4 and 5 of the integrative framework and their relevance for the 12 principles

	Costs	Benefits
Principle 1 (holistic approach)	Considers costs	Considers benefits, including strategic ones
Principle 2 (interdependencies)	Considers effects of exchanging system components on expected costs	Considers effects of exchanging system components on expected benefits
Principle 3 (users)	Considers necessary trainings etc. for users	Considers how users effect benefits
Principle 4 (alignment)	Considers how alignment between technology & users affects costs	Considers how alignment between technology & users affects benefits
Principle 5 (business environment)	Considers also costs of indirectly affected units	Considers also benefits of indirectly affected units
Principle 6 (IS/IT environment)	Takes into account necessary changes of indirectly affected IS/IT environment	Takes into account how indirectly affected IS/IT environment can contribute to benefits / objectives
Principle 7 (singularities)	Effects of singularities regarding costs are taken into account (e.g. specific trainings, specific hardware…)	Effects of singularities regarding benefits are taken into account (e.g. does my current team hinder planned benefits due to their affinity towards new technologies?)
Principle 8 (objectives, benefits, costs)	Pays attention to costs	Pays attention to benefits
Principle 9 (life cycle)	Considers whole life cycle of system	Considers benefits and their development during life cycle of system
Principle 10 (multi-dimensionality)	-	Also strategic benefits are taken into account
Principle 11 (situationality)	Considers situationality and provides different granularity of detail / detail level and thus options for scaling	Considers situationality and provides different granularity of detail / detail level and thus options for scaling
Principle 12 (CSF)	Allows consideration of cost changes if CSF are / are not taken into account	Allows consideration of changes in benefits if CSF are / are not taken into account

Potential effects are also an output of this activity (potential effects: O1*); they represent targeted effects in the given mobile business process context.[2]

- The outputs of A1 and A2 are used as inputs for A3, the definition of critical success factors (CSF) of mobile systems. The relevance of these success factors is analyzed subject to the singularities of a specific mobile system and the targeted effects determined in activities 1 and 2. General success factors from activity 2 are prioritized following the preference-neutral approach as described for activity 1, which is a key element of the integrative framework. Success factors with the highest priority are defined as critical.

In addressing economic analysis, two activities are considered necessary: evaluation of life cycle (a) costs and (b) benefits. In more detail:

[2] Example for a potential effect: During mBPR, it turned out that instead of the targeted effect "20% increase in efficiency," and "35% increase in efficiency" is possible (potential effect).

Table 16.3 Activity 6 of the integrative framework and its relevance for the 12 principles

	Risks
Principle 1 (holistic approach)	Economic, technical and social aspects are considered; also how they are affected by risks
Principle 2 (interdependencies)	It can be analyzed how risks influence each other (see procedure of analyzing interdependencies between objectives; same procedure can be applied here so that critical risks can be identified)
Principle 3 (users)	Risks caused by users are considered
Principle 4 (alignment)	Risks that go along with – particularly not happening – business/IT alignment are considered
Principle 5 (business environment)	Risks that are linked to the business environment are considered (e.g. what kind of risks is caused by different processes?)
Principle 6 (IS/IT environment)	Risks that are linked to the IS/IT environment are considered (e.g. what happens if network coverage is not good enough?)
Principle 7 (singularities)	Singularities of a specific system are analyzed in terms of risks that go along with them
Principle 8 (objectives, benefits, costs)	Risks influence benefits & costs
Principle 9 (life cycle)	-
Principle 10 (multi-dimensionality)	-
Principle 11 (situationality)	Risks analysis takes into account the situationality of a specific project
Principle 12 (CSF)	CSFs and risks are inherently related. If CSFs are not taken into account, risks are higher.

- Expected costs can be calculated (activity 4, O8) by applying the life cycle-oriented total cost of ownership approach (Ferrin & Plank, 2002; Gartner Group, 1997; Grob, 1993) as it takes all costs into account that occur during the lifetime of a mobile system. This includes costs that occur in other departments that are directly or indirectly affected by the implementation of a mobile system (Unhelkar, 2009). Targeted effects (O1) and potential effects (O1*) can be taken as a basis for the calculation of costs of different alternatives. An alternative is defined as a particular configuration of a mobile system.
- Taking the outputs of A2 and A3 into account, a first evaluation and estimation of the potential benefits (O9) for each identified alternative is possible. To do so, the mBPR model is to be examined, potential benefits are to be identified, and the best possible alternatives are to be taken as basis for further consideration. The evaluation of benefits, based on the total benefit of ownership model (Gadatsch & Mayer, 2004), involves the capture of cost savings and non-monetary benefits or qualitative and strategic variables, which are not considered in the traditional approaches of economic evaluation.

The last principle evaluates the mobile system in an integrative way, combining the results of the previous activities. In further detail, including activity 7:

- This pillar starts with the identification and analysis of risks (Kronsteiner & Thurnher, 2009) for the different alternatives. Based on the results, we perform a sensitivity analysis by taking CSFs into account for each risk for the alternatives. Subsequently, volatility effects are identified.

- The final assembly of hitherto existing outputs leads to the assessment of potential target achievement rates (activity 7). Different problem-solving techniques and mathematical methodologies for improved decision-making, as regards the identified alternatives, exist. Which of these methodologies should be applied for the analysis of potential target achievement rates, and thus for the final choice of an alternative, depends on many factors like complexity of the project. The alternative with the highest achievement rate is the alternative with maximized benefits against minimized costs, taking into account how critical success factors influence risks. This will also result in a feasible target system.

Thus in fact, activity 7 is the holistic approach as in previous activities (1–6), we take specific perspectives (e.g., only defining objectives; only defining CSF; only analyzing potential costs). In this activity 7, we check if, e.g., costs will probably be 10,000 € or if costs will be 50,000 € as most of the CSFs are not considered or that singularities show that specific equipment and training are needed. As particular example: during mBPR, we identify the singularities (e.g., team of 3 IT-hating blue collars). For the mobile technologies, we need 25,000 €and for trainings 5000 €–in total, 30,000 € (here, we consider only the best case/average costs). We, e.g., have identified CSF "technology acceptance." If we cannot reach technology acceptance with trainings of employees, we will probably need two additional trainings with additional 10,000 EUR. So in contrast to earlier planned 30,000 €, we will have to spend in this case 40,000 €.

16.3 Research Methodology

Next to the development of the integrative framework, a design science research execution implies that it is validated as an artifact (Hevner et al., 2004, pp. 85–87). Hevner et al. (2004) define three generic criteria for such validation: (1) utility of the artifact, (2) quality of the artifact, and (3) efficacy of the artifact. They explain these criteria by (appropriate) functionality, completeness, consistency, accuracy, performance, reliability, usability, and fit with the enterprise (Hevner et al., 2004, p. 85). As these criteria leave room for more specifications, Carvalho (2012) additionally mentions among others generalizability, novelty, and explanation capability (i.e., being able to explain the success of the framework, in comparison with alternative models). How to validate our framework, taking into account the large set of the validation criteria, can be considered a complex and wicked problem. In any case, a single validation approach would be insufficient. Therefore, we consider a multimethod approach—as presented and discussed by Mingers (2001) and Venkatesh et al. (2003)—as the most appropriate way to evaluate our framework. We adopt the four-step guidelines presented by Venkatesh et al. (2003, p. 41) to define the process and structure of the validation:

16.3.1 Step 1: Define the Goals to Clarify the Appropriateness of a Multimethod Approach

We define as the overall objective of our study to reliably show the validity and applicability of the framework that is designed for the preparation of mobile technology-enabled digital transformation. Following Mingers (2001, pp. 243–244), multimethod research is necessary "to deal effectively with the full richness of the real world. [Also advantages like] (i) triangulation—seeking to validate data and results by combing a range of data sources, methods, or observers, (ii) creativity—discovering fresh or paradoxical factors, (iii) expansion—widening the scope of the study to take in wider aspects of the situation [advocate multimethod work]" for reaching our goal.

16.3.2 Step 2: Define a Strategy for Multimethod Validation

The strategy that drives the evaluation of our framework's applicability is based on multiple methods. A primary method applied is an in-depth case study of a mobile technology-enabled digital transformation. As case studies only show applicability in a certain specific context, for generalizability, we conducted experts' opinions as well. As elaborated below, we execute a general retrospective case study in the healthcare sector to evaluate the full integrative framework. In addition, we conduct a specific case study focusing on the first activity of the framework dealing with definition of the target system (at the Dutch fire brigade). Finally, semi-structured interviews with experts are held (project managers and scientists in the field of digital transformation).

16.3.3 Step 3: Define a Strategy for Analyzing Data

The multimethod strategy as defined in step 2 naturally results in different types of data and results. Therefore, we set up, process, and analyze the results of the validation methods via separate techniques and protocols. In the next section, each validation method and its resulting data will be introduced and described, including the description of the protocols and analysis techniques. Also, we described how we can then draw inferences from the combination of the data analyses.

16.3.4 Step 4: Draw Meta-Inferences from the Separate Validation Studies

The framework evaluation is finalized by synthesizing to provide overall inferences. As we will show in the evaluation and conclusion sections, we extract the common validation criteria and reflect on them. The integrative framework we developed to evaluate mobile system implementations ex ante will be validated by these four steps. This research methodology will structure the remainder of this chapter, leading to conclusion and reflection in the last section.

16.4 Qualitative Evaluation of the Integrative Framework

In this section, we describe a qualitative evaluation of the integrative framework based on (1) a retrospective case study, (2) a case action study of the first activity (A1), and (3) an assessment by domain experts. The results of this evaluation, demonstrating the applicability of our framework, are presented in the following subsections.

16.4.1 Nursemapp: A Retrospective Case Study

The case study of Nursemapp, a mobile app for nurses that allows entering vital body functions of patients into mobile devices, provides documentation by which the framework can be validated from an integrative holistic view.[3] For a detailed case description and execution, see Versendaal et al. (2016). The thesis of Heerink (2014) is used as the main case study source to investigate which of the activities of the integrative framework can be recognized in this document of the Nursemapp prototype implementation. We focused on the main text, not on the appendices, and examined if the activities of the integrative framework be identified in the Nursemapp implementation case. We reflected on the activities not explicitly mentioned in the thesis and what this implies. The analysis of the thesis was executed iteratively: First, one researcher studied and coded the thesis, with a priori codes for each of the activities of the framework. Subsequently, this coding was checked and—if necessary—completed by a second researcher. All occurring discrepancies were discussed, and a joint decision was made if a change was needed. The second researcher confirmed in almost every situation the coding of the first researcher. Discrepancies occurred only in few situations and were

[3] See the title of a master thesis: *Should health records go mobile: exploring a mobile health record application in its support to process and quality improvement within hospitals?* (Heerink, 2014).

solved, by, in general, following the second researcher's opinion, who is also the author of the integrative framework (Högler et al., 2015). The coding of the original work of Heerink (2014) is detailed in Versendaal et al. (2016). It describes the agreed-upon results of the coding of Heerink (2014)'s observation of the Nursemapp implementation.

The retrospective case demonstrates the following:

1. We found all activities of the integrative framework—except for "costs"—in text fragments of the thesis, confirming completeness and appropriate functionality. Costs were indirectly addressed as Heerink (2014) mentioned "funding" as of key importance for the success of her case study.
2. The distribution of code occurrences is not even. Particularly, "target achievements" were mentioned only few times, whereas "success factors" were represented very often in the thesis. We explain the limited number of references to "target achievements" by the fact that Heerink's thesis uses the terms "targets" and "benefits" synonymously. In contrast to Heerink's thesis, we distinguish between different levels of abstraction and thus between benefits and targets.
3. We particularly checked whether activities could be recognized in the text, not how they were specifically executed. Yet in some activities similar techniques are used; for instance, the technology acceptance model is described in the thesis and suggested in the detailed description of our framework in connection with the definition of success factors. In this case, the validation criterion consistency is supported.

16.4.2 Electronic Learning Environment: A Case Study at the Dutch Fire Brigade

The second case study, evaluating the first activity (target system definition) of the integrative framework, concerns the introduction of an electronic learning environment (ELE), leveraging mobile components, at the Dutch fire brigade (see for detailed description and execution of the case study Versendaal & Högler, 2017).

1. It should address the validation of the first activity (target system definition).
2. It should relate to a major (mobile) system implementation, in a large enterprise, currently being prepared.
3. It should be easily made clear to the enterprise that carefully thinking about targets, upfront system implementation, is utmost important.
4. There should be willingness from the enterprise to participate in the validation activity.

The Dutch fire brigade had acquired "Three Ships N@tschool!", as a system for supporting lifelong (place- and time-independent) learning for firemen. Now it is to be decided, which goals have to be achieved with the system within the fire brigade organization. Having the goals and their prioritization clear, requirements can be derived, and the system can be configured accordingly. Therefore, the case study focused on the definition of the target system. Six of the 25 fire brigade regions (representing almost 6000 firemen), with the support of the educational department, decided to take a leading role in developing a showcase on how to implement the ELE.

We followed a structured case study protocol that guides in conducting the case study (Yin, 2013). We took the standard research design template of Maimbo and Pervan (2005) for describing our validation protocol. It contains information about chosen procedures, research instruments, and data analysis guidelines. While the case study was conducted, the proposed protocol was followed. The procedure and outcomes are described in the in-depth report by Versendaal et al. (2016). In performing the first activity of the integrative framework, we closely followed the steps identified in Sect. 16.2 and started with the brainstorming of the goals of the ELE implementation with the six region heads, including some bystanders (step 1). One of the researchers managed the process during the brainstorming, while a secretary of the fire brigade observed and took notes and created afterward the dependency matrix (steps 3, 4, and 5; see Fig. 16.2 for a snapshot of interdependencies between objectives). Based on these results, the researchers constructed a goals' hierarchy (step 2), which was checked and approved by the secretary. The researchers together performed step 6 (defining high-, medium-, and low-priority goals from the values in the dependency matrix, only considering lowest-level—so-called process—goals). In addition to the prioritization using the dependency matrix, one of the team managers (involved as a bystander in step 1) created an ad hoc prioritization; this helped in the discussions while executing step 7 (describing the final target system, with final prioritized goals).

To validate the framework with this pilot implementation as case study, we elaborate on the following validation criteria:

1. Can all steps of the first activity of the integrative framework be performed successfully? The fire brigade case shows that indeed all steps of activity 1 can indeed be applied successfully.
2. Is the execution of the first activity of the integrative framework considered to be accurate? In a reflection, the fire brigade's secretary states that he considers the model highly accurate, if applied following a robust procedure: he suggests to undertake steps 3–5 with multiple employees, so that consensus on the resulting goal priorities can be made. This confirms the procedure as applied by Högler and Versendaal (2016), in which multiple user groups created multiple dependency matrices, which were consolidated in step 7 of the framework's first activity. In addition, as demonstrated at the fire brigade's validation, an extra ad hoc prioritization helps in providing a reference for discussion on the prioritization through the dependency matrix.
3. Is the execution of the first activity of the integrative framework considered to be useful? In relation to especially steps 3–5, the remark of the secretary was that it was an "[...] extremely time-consuming execution; [...] it lets you focus on what

	Uniformiteit in verblende producten tussen regio's	Eenduidige context van opleiden	Content meer delen met regio's; inhoud/content gelijk over regio's	Verhoging kennisniveau door gebruik kennis verschillende regio's	Ontwikkelen content door en voor meerdere regio's	Landelijk ontwikkelen van leertraject ipv regionaal	Zelfde niveau opgeleid over regio's	Samenwerking intensiever mbt vakbekwaamheid	Leuk en uitdagende leeromgeving	State-of-the-art leeromgeving	Hoge kwaliteit leerproducten	Inspirerende leeromgeving
Uniformiteit in verblende producten tussen regio's		2	3	2	3	3	2	2	2	1	2	1
Eenduidige context van opleiden	2		3	3	3	3	3	2	2	1	2	1
Content meer delen met regio's; inhoud/content gelijk over regio's	3	2		3	3	2	2	2	2	1	2	2
Verhoging kennisniveau door gebruik kennis verschillende regio's	2	1	3		3	2	1	2	2	2	2	2
Ontwikkelen content door en voor meerdere regio's	3	2	3	3		1	2	2	2	2	2	1
Landelijk ontwikkelen van leertraject ipv regionaal	3	2	3	1	-1		2	0	1	0	0	1
Zelfde niveau opgeleid over regio's	2	3	3	2	3	1		1	0	0	0	0
Samenwerking intensiever mbt vakbekwaamheid	2	2	3	2	3	-1	1		1	1	1	1
Leuk en uitdagende leeromgeving	1	0	1	2	2	1	1	1		3	2	2
State-of-the-art leeromgeving	0	0	0	0	0	0	0	0	2		1	3
Hoge kwaliteit leerproducten	1	1	1	1	0	0	1	1	1	1		1
Inspirerende leeromgeving	0	0	1	1	1	0	0	0	2	2	1	

Legend:
-1 horizontally depicted objective A has a moderate negative impact on obtaining vertically depicted objective B
0 horizontally depicted objective A has no impact whatsoever on vertically depicted objective B
1 horizontally depicted objective A has moderate impact on vertically depicted objective B
2 horizontally depicted objective A has quite some impact on vertically depicted objective B
3 horizontally depicted objective A has major impact on vertically depicted objective B

Fig. 16.2 Snapshot of dependency matrix (in Dutch)

is really of importance, but it costs a lot of effort. Yet at the same time I admit it is very useful: it will help during the actual execution of the implementation project for the ELE to concentrate on the really important things!". What might help in saving time with steps 3–5 is the determination of the object hierarchy before (instead of "in parallel with") step 3, so that the dependency matrix only consists of process goals (the lowest-level goals that are drilled down from key goals and basic goals; see Fig. 16.2). Also presenting the dependency matrix in another format (e.g., as a list) may contribute to the speed with which values can be entered in the matrix. We end with the statement of the secretary saying that, although creating the goals' prioritization through the dependency matrix was time-consuming, the investment at the start of the project (in defining thoroughly the target system) would definitely pay itself back during the execution of the actual ELE implementation.

16.4.3 Validation of the Framework by Experts

In the timeframe of February and March 2017, six experts from research and practice were recruited to validate the framework as in a consultation round. The selection criteria for choosing experts were:

1. A high familiarity with the topics of IT project management
2. A high familiarity with the evaluation of economic efficiency of IT systems
3. A long-time experience in practice

To identify the experts efficiently, personal business contacts were used in order to identify potential candidates. To select them on experience related to the integrative framework, CV of each candidate was checked including their company web page. In a second step, a first group of seven evaluators was contacted. These experts were addressed personally via email, explaining the purposes of the study and asking them if they were interested in participating in the survey and an interview. Eventually, six evaluators were found to participate. Upfront, they have been provided a description of the framework and the related survey. With the first three experts, the survey was discussed during the interview. The other experts were asked to send the completed questionnaires back before the actual interview, so that in the interview, the authors could focus on particular parts of the questionnaire (i.e., the parts in which the expert did not (fully) agree). For details on this procedure, we refer to the report (Högler & Versendaal, 2017).

16.4.3.1 Structure of the Questionnaire

The questionnaire contained 11 sections (Fig. 16.3): 1 introductory section that contained general information like scope of the interview and questions as regards the general understanding of the procedure (questions 1–4 (Q1–4)). Section 16.2 gathered general information about the interviewee like personal data, experience in the topic of the integrative framework, and confidentiality/usage of the gathered data. The third section of the questionnaire provided specific assumptions, which we call axioms, of the integrative framework and validated them (Q5–7). Section 16.4 concentrated on the validation of the integrative framework as a whole (Q8–13), whereas sections 5–11 validated every single of the seven activities of the framework (Q14–32).

The questionnaire consisted of survey questions that could be answered by dropdown menus with pre-defined answers (yes/no/partly) and the possibility to enter free text for the case, if “no” or “partly” was chosen in the previous question as answer. With the questionnaire as an interview guiding instrument, the validity and applicability of the integrative framework were assessed, specifically by asking the experts about the (appropriate) functionality, completeness, correctness, usability, generalizability, and novelty of the integrative framework.

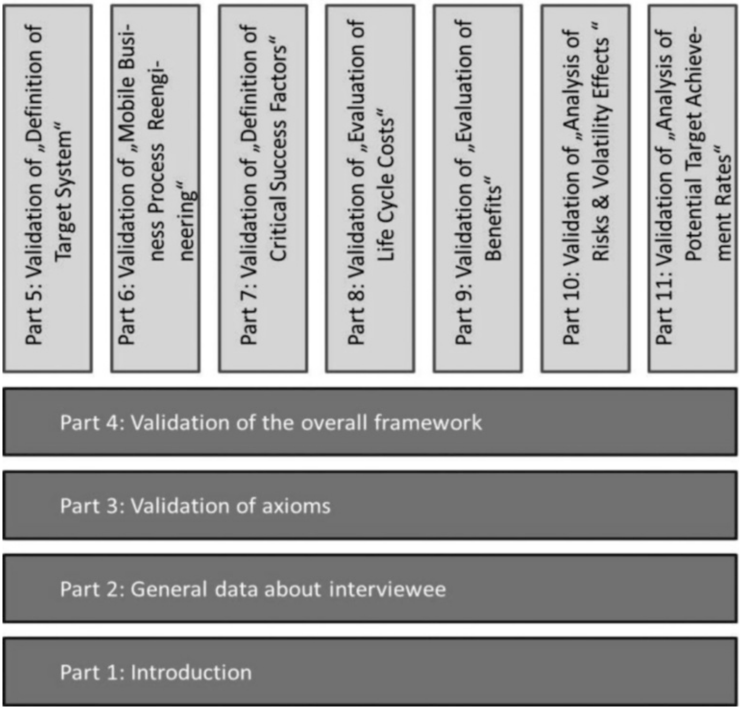


Fig. 16.3 Structure of the questionnaire

Table 16.4 Participants of the study

Name	Affiliation	Position	Familiarity	Practical experience since
Prof. Dr. Hans Mulder	Venture Informatisering Adviesgroep NV	Managing director	Very familiar	1995
	Antwerp Management School	Academic Director & Executive Professor		
Prof. Dr. Rainer Neumann	University for Applied Sciences, Karlsruhe	Professor	Very familiar	1998
Prof. Dr. Dieter Hertweck	HHZ Research Centre, Reutlingen University	Professor, Head of Research Group	Very familiar	1996
Dr. Asarnusch Rashid	Zentrum für Telemedizin Bad Kissingen	Managing director	Familiar	2004
Daniel Stucky	Keller Informatik AG	Owner	Very familiar	1988
Rüdiger Bäcker	@TOLL GmbH	CEO	Very familiar	2004

16.4.3.2 Analysis of Results and Suggestion for Improvement of the Integrative Framework

In terms of their business position, the six evaluators came from SMEs and research institutions. They all had positions like CEO or professor and many years of experience in the topic “IT project management.”See Table 16.4.

The detailed interview results are described in the report by Högler and Versendaal (2017).

From the comments and suggestions queried to validate the integrative framework by the experts, the following becomes clear:

1. The *axioms* of the framework were confirmed by the experts. Discussion came up as regards activity 2 (mBPR) which needs to be described more precisely for a better understanding. For example, it could be useful for readers to know if business (IT) strategy is taken into account during mBPR. Otherwise, it is difficult to judge, if all requirements (based on activity 1) and system specifications can be derived from mBPR.[4] In addition, it needs to be pointed out clearly that also stationary information and communication systems can be a result of activity 2 and not necessarily only mobile systems. "Singularity" as a term needs to be defined more precisely; particularly, it needs to be better elaborated that it contains also aspects of "stories," "context," and "personas."
2. The *framework* as a whole is confirmed by all experts as regards the correctness and usefulness of its set of three principles and its seven activities. Discussion came up as regards the term "completeness": Two experts stated that without being provided a more detailed description on how the activities are implemented, it is difficult to judge if they are complete as for different kinds of projects, different sub-activities and methodologies can become necessary. Due to the fact that the integrative framework delivers a guideline for evaluating (mobile) ICS which can take place in different contexts and within different sizes of projects, it does not provide a standard procedure that can be applied with the same quality of results for all kinds of projects. The following general statement can be made: The bigger and more complex a project is, the more detailed and extensive the evaluation has to be. This in turn means that additional methods within an activity can become necessary, which are currently not made explicit in the integrative framework (e.g., a culture analysis as proposed by H. Mulder, which can be part of mBPR). The main recurring feedback is related to the iteration of the seven activities. The framework can be significantly improved by explicitly stating that iterations between activities are possible. An iterative approach within the framework was in general possible, but the iteration mainly focused on activity 7 (analysis of the potential target achievement rates) and activity 1 (definition of the target system)—by figuring out which targets can finally be achieved with the given project framework (e.g., budget, timeframe), the target system (i.e., result of activity 1) can be adapted.
3. An enhancement of the framework would be reached by following a comment by R. Neumann: He suggested to define critical success factors in the description of the framework in a clearer way (keyword: "intersection," which success factors are general ones, when do they become critical, etc.), e.g., by applying a Venn diagram for the visualization of results of the matrices. This additional—

[4] Remember: System specifications are derived from requirements (which are outputs of activity 1) and based on outcomes of activity 2.

descriptive—sub-step would make it easier to understand how success factors turn into critical success factors. In addition, he also suggested to conduct the analysis of risks and volatility effects at an earlier stage; generally speaking, this analysis could take place at an earlier stage, but then the integrative approach would be lost as the analysis of risks and volatility effects considers results of the previous activities.

16.4.3.3 Concluding Remarks for Improvement

As regards their completeness of the seven activities, two experts pointed out that a more detailed description about how the activities are implemented and which methods are applied is necessary to be able to provide a profound validation as regards completeness. The analysis of all potentially applied methodologies and sub-activities could be part of a future study. To enhance the framework, it was suggested to allow iteration between single activities and to implement the analysis of risks and volatility effects at an earlier stage, though, by analyzing risks and volatility effects at an earlier stage, the integrative character of the approach would be lost as outputs of previous steps could not been taken into account anymore. Another suggestion for improvement was to provide a more detailed description on evaluation criteria ("when does a success factor become a critical success factor") which would enhance the framework/this activity.

16.4.3.4 Reflection on the Validation Criteria

The main objective of the chosen multimethod approach for validation was to reliably show the validity and applicability of the framework that is designed for the preparation of mobile technology-enabled digital transformation. To do so, we have used different validation criteria. In Table 16.5, we summarize the validation results as for support for these criteria.

16.5 Conclusion

In this chapter, an integrative framework to evaluate mobile system is presented (Högler et al., 2015) that delivers insight into its tangible and intangible effects before it is being implemented and thus represents an ex ante evaluation approach. The integrative framework was evaluated by applying a multimethod approach, i.e., a retrospective case study, a case action study, and a validation by experts in the area of project management. The results of these validations are shown in this chapter and can be explored in further detail in the respective separately available validation reports. Summarizing these findings, we can state that the framework has potential applicability in supporting decision-making

Table 16.5 Results from the case studies and expert interviews with regard to the validation criteria of the integrative framework

	Retrospective case study	Fire brigade case study	Experts
Consistency	Usage of same type of approaches in activities		
Accuracy		Following a robust procedure, the first activity (building the target system through a dependency matrix) is considered highly accurate	
Appropriate functionality	All activities, except for an explicit cost evaluation, were also identified in case	All steps of the first activity can be performed successfully	Appropriateness of activities is confirmed by all experts. One even mentioning to start applying it for his own anticipated projects (Högler & Versendaal, 2017)
Completeness	All activities, except for an explicit cost evaluation, were also identified in case		Only verifiable when the execution procedure for each activity is known. The more complex a project, the more detailed and extensive each activity needs to be executed.
Correctness	All activities, except for an explicit cost evaluation, were also identified in case	All steps of the first activity can be performed successfully	Underwritten by experts for principles and activities. Suggestion for leveraging specific techniques in third activity (Critical Success Factors identification). Suggestion for more interaction between activities and re-addressing certain activity based on outcome of other activity.
Usability		The execution of the first activity is, especially in the case of many identified goals, cumbersome	Underwritten by experts for all activities
Novelty			No similar integrative approaches known
Generalizability		Applicability for ICS that has limited mobile technology	Applicability of the framework even beyond mobile systems confirmed by experts

processes for mobile system evaluation in a comprehensible way. Its appropriate functionality was confirmed in both case studies and by all interviewed experts who assessed the framework based on their long-time practical experience in this field. For the Nursemapp case study, the general approach of the framework could be addressed explicitly for six out of seven activities. The fire brigade case action study confirms particularly the accurateness, utility, and effectiveness of the first activity of the framework, the preference-neutral definition of the target system. The set of the three principles (internal analysis, economic analysis, and integrative evaluation) as well as the seven activities were confirmed by all of the consulted experts. The *novelty* of the integrative framework as a whole, but particularly the proposed approach for defining the target system and critical success factors, was approved by all experts, stating that they did not know any similar approaches. Suggestions for improvement (from the expert validation) focus on the re-addressing of activities and more insight into the method of particular activity execution. Further research on the validity of the integrative framework can be done for the single methodologies and approaches applied in the seven activities. The lack of a detailed description was mentioned by several of the involved experts, meaning that there is room to improve the completeness of the framework.

Chapter 17
Conclusion

Kazem Haki, Bas van Gils, and Henderik A. Proper

The constituent chapters of this part seek to make advances in prescriptive EA.

As outlined in the introduction of this part, to contribute and give direction to the discussion of whether and how enterprise architecture (EA) can be employed to deal with digital transformation's associated challenges, we first need a thorough understanding of EA's theoretical foundations. While the notion of EA comprises both descriptive and prescriptive aspects (Haki & Legner, 2021), existing research predominantly focuses on descriptive EA and leaves the prescriptive aspect somewhat underserved. The set of chapters in this part seek to make advances in prescriptive EA (i.e., architecture *principles*), with a specific focus on EA in the era of digitalization and digital transformation.

Chapter 14 makes us aware of the fact that immaturity in understanding and in the establishment of architecture principles is not only reflected in research. But also, in practice, organizations employ diverging approaches to architecture principles and disregard the criticality of explicitly specifying architecture principles' rationales and implications. Chapter 15 discusses the necessity of a new approach to EA due to digitization and the exposure of contemporary organizations to new drivers such as *customer centricity*, *data-driven service innovation*, and *collaborative value generation*. To this end, this chapter raises the need for new architecture principles in organizations as well as new ways through which architecture principles are

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_17

applied. Finally, in elaborating on how architecture principles help in addressing the associated challenges of digital transformation, Chap. 16 employs a set of architecture principles (categorized in *internal analysis*, *economic analysis*, and *integrative analysis*) to derive a framework for mobile technology-enabled digital transformation.

To conclude, our argumentation for EA to address digital transformation's challenges is as follows. In dealing with hyper-turbulent and dynamic environments, EA needs to be sufficiently *agile* to constantly adapt IT landscapes to ever-changing organizational and technological requirements. While such an adaptation process is required to survive and thrive in the environment, it also bears the risk of making architecture's evolution inherently emergent and its outcomes inevitably unpredictable (Haki et al., 2020). As such, besides architecture requiring a *plastic core* (*descriptive EA*) to evolve dynamically with environmental changes, it requires a set of principles as a *robust core* (*prescriptive EA*), in order to purposefully guide its evolution (Haki & Legner, 2021). Thus, architecture principles are crucial to ensure the *guided*, instead of entirely emergent, architecture evolution to obtain EA's predefined value and outcomes.

Part IV
An Enterprise Modeling Perspective

Chapter 18
Introduction

Bas van Gils, Kazem Haki, and Henderik A. Proper

In our view, digital transformation is a *complex* venture. In previous work (e.g., Proper et al., 2018c), we took that stance that digital transformation requires *coordination* of activities and that the *architecture* discipline is positioned best to play a leading role in this coordination effort. In Part III of that work, we explored coordination constructs from an architecture perspective. One of the conclusions was that models can play an important role for achieving effective coordination and, as a result, contribute largely to the success of the transformation initiative.

This part picks up this thread, where we further explore the role of models/modeling in the context of digital transformation.

This exploration is kicked off in Chap. 19 by noting the parallel between cartography in the "real" world and the digital world. This chapter introduces the notion of enterprise cartography while illustrating this with five cases. It also includes some (brief) thoughts on building up and maintaining an effective knowledge base.

This sets the stage for Chap. 20, which zooms in on the evolution of enterprise architecture models. More specifically, it introduces a conceptual model that can be used to *evaluate/express* the evolution of models which is a first step in *managing* effective evolution of the models in light of the ever-changing needs (for models in digital transformation initiatives) in the enterprise.

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_18

Finally, Chap. 21 evaluates ArchiMate, the de facto standard for enterprise architecture modeling, in light of the modeling needs for digital transformation. Based on our analysis, we present the outline of an alternative approach which might set the stage for the next version of this open standard.

Chapter 19
Enterprise Cartography: From Theory to Practice

Pedro Sousa, José Tribolet, and Sérgio Guerreiro

Abstract We refer to enterprise cartography (EC) as the process of representing an enterprise as it is observed directly from reality. It differentiates from enterprise architecture (EA) because it focuses on producing representations based on observations and, therefore, does not include the purposeful design, as one expects in EA.

To abstract and represent an enterprise, one requires a full set of familiar concepts from the EA discipline, such as models, views, viewpoints, representation rules, etc. However, to abstract and represent the enterprise reality, one also needs principles and instruments to deal with the continuous and fast transformations going on in the enterprise. Without them, enterprises will likely change faster than one can represent them, and representations become obsolete long before they are completed.

We have been working on this issue over a decade and have come up with a set principles and instruments to enable effective EC in continuously changing enterprises. We have applied and improved our methods in several enterprises of different sizes, sectors, and countries and have had our share of successes, failures, and lessons learned.

19.1 Introduction

According to the Merriam-Webster dictionary (Meriam–Webster, 2003), cartography is *the science or art of making maps*. It is an old and established discipline of producing representations of an object from its observation and measurement. Traditionally, these maps require not only geographic or spatial referential but also scientific, technical, and even aesthetic (Wikipedia, 2019a) ones. Unlike traditional

P. Sousa (✉)
Instituto Superior Tecnico, Lisbon University, Lisbon, Portugal
e-mail: pedro.sousa@linkconsulting.com

J. Tribolet · S. Guerreiro
Instituto Superior Tecnico, University of Lisbon, Lisbon, Portugal
e-mail: jose.tribolet@inesc.pt; sergio.guerreiro@tecnico.ulisboa.pt

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_19

maps, enterprise representations do not have necessarily a geographic or spatial referential. Nevertheless, since the definition of *cartography* matches our concept, we adopt the term *enterprise cartography* (EC).

Simply put, EC is the process of abstracting, collecting, structuring, and representing architectural artifacts and their relations from the observation of enterprise reality.

In this chapter, the expression "enterprise reality" refers to the present state of the enterprise. Traditionally, the observation of this reality is a subjective perception of an observer, and consequently, it is probable that there are different actors that perceive different realities of the same enterprise. However, as we propose and will become clear along the text, the perception of the present state of the enterprise (the reality) is sustained on relevant facts captured in logs and represented through artifacts based on previously defined and agreed-upon models.

We refer to "architectural artifacts" as the enterprise's observable elements whose interdependencies and intradependencies express the architecture of the enterprise. Naturally, different institutions may consider various sets of artifacts as part of their architecture.

To abstract, structure, and represent the architecture-related artifacts, one needs the whole set of knowledge and concepts implied in architecture visualization and representation. For example, the concepts of semiotic triangle (Morris, 1928; Dietz, 2006), the model of architecture description presented in ISO IEEE 1471 (ISO, 2013), and a symbolic notation, such as the one used in ArchiMate (Band et al., 2016), are concepts necessary to the production of architectural maps.

Traditionally, these architectural maps can present the architecture at different points in time, namely, at present (*AS-IS*) and at a given time in the future (*TO-BE*). Within the *TO-BE*, the architecture that is emerging as the result of ongoing transformation initiatives is of particular importance for all sorts of planning purposes, from the strategic macro levels to the full synchronization of multi-project initiatives at tactical and operational levels.

Ongoing transformation initiatives and their plans are an essential part of enterprise reality because they define the near future *TO-BE* of the enterprise if no further decisions are taken that might have an impact on them. This is a fundamental concept that we call *emerging AS-IS*, which we define as the state of the enterprise after the successful completion of ongoing transformations.

Given that the transformations underway are in fact an endless succession of transformations that begin and end, the concept of *emerging AS-IS* is not only an essential capacity for the planning of new transformations but also for the monitoring of the enactments of ongoing transformations. When driving a car, the faster the car is moving, the farther ahead one should focus our eyes to match the longer time and distance context within which we need to steer it. Likewise, when steering an enterprise, the faster the enterprise is changing, the more important it is to know the *emerging AS-IS*, as time flows.

Continuing with the car metaphor, if one knows the car situational reality (position, speed, and direction), then the car future's position, speed, and direction can be estimated unless some unforeseen action (e.g., break or turn) is taken. Much like the car's speed and direction, ongoing transformation initiatives are, in fact, the core elements of enterprise inertia, since they pre-condition the future of the enterprise (the *TO-BE*), unless some decisions are taken meanwhile to change such core inertial elements. Since plans of ongoing transformation initiatives are observable artifacts, by observing in real time the enactment of the enterprise reality, one can predict the expected future enterprise states.

EC is a purely descriptive perspective, since it does not explicitly incorporate the purposeful design of the new enterprise artifacts, as one expects in enterprise architecture (EA). Such a difference is also evident in the definitions of the *architect* and *cartographer* roles. According to the IEEE Standard Glossary of Software Engineering Terminology (ISO/IEC/IEEE, 2015), the term *architect* is defined as "the person, team, or organization responsible for designing systems architecture," and in the Merriam-Webster dictionary (Meriam–Webster, 2003), the term *cartographer* is defined as the *person who makes maps*. So, an *architect* is essentially a person that designs and shapes intended changes to the architecture of the present enterprise reality, while a *cartographer* is essentially a person that aims at representing reality as it happens, including such changes as they are occurring.

Cartographic change as it is happening is the most challenging aspect in EC since it implies doing representations of evolving objects. The real challenge comes when the practices of abstracting and representing the evolving enterprise occur at a rhythm that is slower than the actual rhythm of enterprise changes.

Therefore, to abstract and represent enterprise reality, one also needs principles and instruments to deal with the continuous and fast transformation of the enterprise. Without them, enterprises will likely change faster than one can represent them, and representations become obsolete long before they are completed. This is, in fact, the application of the famous Nyquist's sampling theorem (Wikipedia, 2019b) applied to enterprises while following general systems theory (Le Moigne, 1985; Wikipedia, 2019c). As a result, the sampling rate of sensing events must be high enough to follow the underlying dynamics of changes occurring in the underlying physical environment.

In this chapter, we start presenting the key motivations for the practice of EC in real enterprises. Firstly, we argue that accurate maps of the enterprise are a key asset not only to support day-by-day operations but also to support the planning of transformation initiatives. Secondly, we argue that keeping such maps continuously updated is easier when the roles of architectural design are kept apart from the roles of cartographic representation, thus enabling the separation of concerns between architecture and cartography.

We now present an overview of similar approaches to cartography existing in different domains, from infrastructure to business processes. Next, we introduce the set of definitions and principles that shape our approach to EC. After that, we present some lessons and conclusions from our experience in real enterprises, and finally, we present our conclusions on the usage of EC.

19.2 Motivation for Enterprise Cartography

In general, simplifying and streamlining enterprises entails not only new processes and systems but also greater integration with existing ones. More and more integrated processes and systems result not only in an increase in enterprise complexity but also in a greater need to know the enterprise current and emerging processes and systems. Unfortunately, these two factors feed each other in a positive feedback loop: the more complex the enterprise is, the more one needs to know, the harder it gets to know it. Such positive feedback significantly contributes to the increase in costs, risks, and execution time for transformation initiatives, as well as the complexities of systemic enterprise steering and governance.

The problem of keeping a set of accurate representations (i.e., models) of an enterprise is not an easy one for large enterprises, which can have hundreds of transformation initiatives, of different scopes and sizes, each year. The origin of this difficulty is that the planning process originates in many of the enterprise's communities without a consistent, coherent, and complete systemic view of the enterprise to support them, individually and as a whole.

Based on our observations, the design of the enterprise's architecture is a distributed process in most enterprise, performed by architects from different domains (from business to technology), each planning, designing, and deciding based on partial and often conflicting or even incompatible views of the enterprise. This state of affairs is entirely different from the one it is observed in some industries, in particular, those for which real-time systemic integrity is essential to their survival, such as defense, where all the efforts are made to ensure at all times the enterprise transformations are steered by a centralized and fully controlled design process (Dam, 2015).

The list of reports on failed EA programs in enterprises is large (Jacobson, 2007; Roeleven & Broer, 2019; Gunther, 2014). Nevertheless, in spite of high levels of EA program failures, enterprise transformations continue to lack purposeful designs, according to systemically coherent architectures! In fact, most enterprises today are still the result of the "natural evolutions" that happened from their original birth configuration, and present transformations keep being architected by enterprise architects according to partial needs and points of view, regardless of the eventual existence of global EA programs to guarantee the global integrity of the *TO-BE* enterprise that eventually emerges from such efforts.

What EA programs fail to produce is not the architecture of an enterprise, but the establishment of the processes and tools to provide complete and accurate information that sustain a good architecture design, where alignment between the different elements was taken into consideration and resulted from a conscious design. In other words, they cannot create a single, consolidated view of the enterprise because this requires coordination between different areas, from coherent and non-conflicting goals, rational and implementable use of resources, cost sharing, work methods, languages, tools, etc. (Jacobson, 2007; Roeleven & Broer, 2019).

Such a consolidated model can be obtained and sustained by forcing all architects to use and share the same modeling notation, the same level of detail, and probably also the same modeling tool. However, this is very hard to achieve in real enterprises, because enterprise architects are in fact a very heterogeneous group, in what concerns their backgrounds, their knowledge, and their needs. In practice, different architects can use different notations, different levels of detail, or even different tools, making it harder to support a design process that enables consistent and coherent views of the enterprise architecture.

With the introduction of the concept of EC, we can decouple the problem of design from the problem of building a consolidated set of representations of the enterprise. Different architects can use their preferred design approach, leaving up to the cartographer the task of consolidating each partial architecture into a single global set of maps representing the enterprise architecture of the *AS-IS* as well as of the *emerging AS-IS*. In the course of this consolidation process, many inconsistencies may arise. However, they should pass on to the enterprise architects involved so that any inconsistencies and opposing points of view can be dealt with the adequate governance mechanisms, which form the essential core of the entire organizational governance (Hoogervorst, 2009). As a matter of fact, enterprises do not have explicit governance mechanisms aimed at maintaining global systemic integrity, while subject to the multitude of changes that are in progress in a distributed fashion throughout the enterprise. To realize such governance mechanisms, they need a systemic, coherent, and complete view of the enterprise as provided by EC.

Another important aspect is that EC is less intrusive than EA and, therefore, more easy to implement in real enterprises, the reason being that defining the shape of the enterprise is naturally perceived as a more relevant role than representing it. Enterprise architects are naturally more willing to cooperate with others when the ownership of the design remains theirs than with other architects with whom they would have to negotiate and share the ownership of new designs.

Enterprise *AS-IS* and *emerging AS-IS* are used for different purposes. The first is a key asset to operate the enterprise, for example, to analyze the impact of a response to an event, whereas the second is a key asset for the planning of the transformation initiatives. For example, consider that today's date is January and one want to plan a transformation initiative that is expected to start in April and to conclude in September. Assuming there are other ongoing transformation initiatives, the enterprise might change from January to April, and therefore, the *AS-IS* state in January is not sufficient for the planning of the actions that will occur in April. In fact, to do the job properly, one needs to keep estimating as best as possible the enterprise future *TO-BE* states from April to September, based on the best available *AS-IS* at any point in time during the transformation time frame.

Transformation initiatives create, remove, or change artifacts and their interdependencies that might result in changes to architecture, as stated in the description of their outcomes. In IT, where transformation initiatives are typically called projects, such descriptions are either stated in natural language or models such as BPMN (Grosskopf et al., 2009), ArchiMate (Band et al., 2016), and UML (Booch et al., 2005) among others. The conciliation of all these descriptions in an integrated and consolidated vision is a task that entails a massive human effort and is therefore far beyond the reach of the vast majority of enterprises. Thus, the key motivation for EC is to provide architectural maps of the enterprise *current AS-IS* and *emerging AS-IS* with the necessary architectural information to those planning and executing the changes. We claim that EC allow us to provide a generic and systemic approach, where only a minimum effort is necessary to create and maintain *current AS-IS* and *emerging AS-IS* architectural maps.

19.3 Approaches to Enterprise Cartography

The idea of creating maps from observation of the enterprise reality is not new, in particular if one considers data extracted from systems in an efficient manner to observe the enterprise reality.

At the infrastructure level, configuration management databases (CMDB) as defined by ITIL (Office of Government Commerce, 2007; Axelos, 2015) represent the configurations and relationships of the IT infrastructure components. Such information can be used to produce architectural maps of the enterprise infrastructure. For example, some solutions provide auto-discovery techniques that detect nodes, virtual machines, and network devices to create infrastructure architectural views. Auto-discovery is a cartographic process and requires that the type of the concepts to be discovered be known in advance (Filipe et al., 2011). The resulting CMDB instance is a partial model of the enterprise's infrastructure. This model can be communicated through different visualization mechanisms, such as textual reports or graphical views that require a symbolic notation and design rules (Band et al., 2016).

At the business and organizational layers, cartography techniques are found within the discipline of business process management, especially in process mining. These techniques make use of event logs to discover process activities and control and data flow and also to assess the conformance of existing processes against constraints (Agrawal et al., 1998; van der Aalst, 2011; van der Aalst et al., 2012). Mined processes correspond to the actual instances of processes, not to the designed or modeled processes. Another example of enterprise cartography is the inference of inter-organizational processes based on EDI event logs (Engle et al., 2012).

Business intelligence techniques that collect data from enterprise systems to produce reports and dashboards may be considered yet as another example of cartography, even though it is mostly concerned with the analytic aspects rather than with architectural (relational) aspects of the enterprise. However, we can still envisage many analytical views within an architecture, and therefore, we may also consider business intelligence techniques as another example of cartography and the traditional ETL (extract, transform, and load) process as a cartographic one.

One may say that EC has been a reality in many specific domains, concerning specific variables and hard-coded with limited and pre-defined meta-models and concepts. Most commonly, referred techniques aim at producing *current AS-IS* views of the enterprise. Even though not so common, some also address the *emerging AS-IS* views of the enterprise, especially the analytical approaches.

We aim at a generic and systemic approach, where the effort to produce and maintain such *current AS-IS* and *emerging AS-IS* enterprise architectural views is kept to a minimum. Earlier results of our approach have, e.g., been reported in Sousa et al. (2009).

19.4 Enterprise Cartography Principles

We start by presenting key definitions upon which we then present our cartography principles.

19.4.1 Definitions

- *Enterprise meta-model*—An enterprise meta-model refers to a set of artifact types and set of allowed relations between them used in the definition and visualization of the enterprise architecture. It establishes the subjects (artifacts), the verbs (relations), and the states (adjectives) used to express the architecture.
  Examples of artifact types are *department*, *information system*, *service*, and *process*, while examples of relations are *realizes* and *uses*, and examples of states would be *productive* and *decommissioned*.
  We are only considering artifact types, relations, and states that are relevant to describe the enterprise's architecture.
- *Architectural statement*—An architectural statement is a well-formed proposition regarding the organization architecture. It must necessarily be expressed using the artifact types and relations defined in the meta-model.
  Using the above examples, the following are architectural statements: 'Sales *department uses* CRM *information system*" or "*process* complain management realizes *service* customer complain." Architectural statements can also refer to artifact's states as "CRM *information system* is decommissioned."
- *Architectural map*—Since we are considering only the artifacts that are relevant for the enterprise architecture, architectural maps are simply a graphical representation of a set of architectural statements over a period of time. Having multiple statements referencing different points in time implies that architectural maps should allow for the visualization of changes in artifacts and their relations and states.
- *Alive artifact*—An alive artifact is a productive one, in the sense that it can play roles in the organizational transactions and processes to create value. This means that the artifact is reachable via the dependency graph of enterprise alive products and services. So, the aliveness state of an artifact is defined from a graph traversal

and is not an intrinsic property of the artifact. The roots of this graph are usually the enterprise products and services that are active at a given moment in time.

- *Transformation initiative*—A set of planned and purposeful activities that might yield changes in the enterprise artifacts, such as projects. A transformation initiative holds a set of architectural statements to become true after its successful completion. We consider ongoing transformation initiatives to be alive artifacts of the enterprise. It is not within the scope of this chapter to fully justify this assumption. Suffice is to say that ongoing initiatives are an observable part of the enterprise reality. Depending on the enterprise meta-model, the following might be transformation initiatives: the creation of a new department, the deployment of a new information system, and the hiring of a new employee.
- *Enterprise observation*—An enterprise observation relates to the ability to formulate architectural statements from the observation of the enterprise reality, including both the alive artifacts and the items of the transformation initiatives.
- *Knowledge base*—A knowledge base, hereafter designated as KB, is the repository holding the meta-model, the architectural statements, and the definition of the conceptual maps.
- *AS-IS state*—Is the current set of all alive artifacts in an enterprise and their relations and states.
- *AS-WAS state*—Is the set of all alive artifacts in an enterprise and their relations and states as observed at a particular point in time in the past.
- *Emerging AS-IS state*—Is the set of all alive artifacts and their relations and states as observed after the successful completion of ongoing transformation initiatives. If we consider that the date of completion of the last transformation in the transformation pipeline is lastDay, then the emerging *AS-IS (Td)* corresponds to *TO-BE (Td)* from any day *Td* between the present day to lastDay. That is, the emerging AS-IS corresponds to TO-BE assuming that there are no new decisions and that the ongoing changes continue as planned. Taking into account that ongoing transformation initiatives fulfil their current plans, then the *emerging AS-IS* state is the same as the *TO-BE* at the time of lastDay.
- *TO-BE state*—Is the set of alive artifacts and their relations and states in point in time in the future. The *TO-BE* considering only the outcomes of transformation initiatives corresponds to the emerging *AS-IS*.
- *Enterprise architect*—A person who makes architectural statements regarding some point in time in the future.
- *Enterprise cartographer*—A person who collects architectural statements from observations of the enterprise reality and produces architectural maps.

19.4.2 Principles

Our approach for EC is grounded on the following principles:

- *Principle 1—Transformation initiatives are enterprise artifacts.*

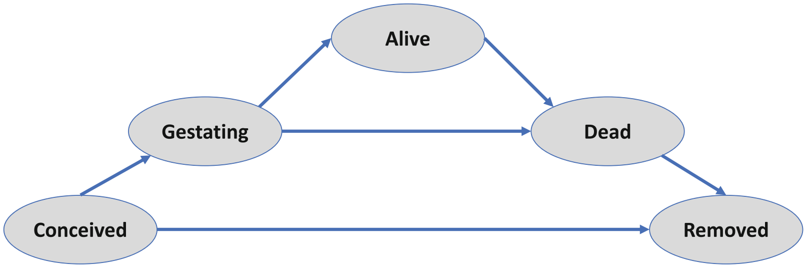


Fig. 19.1 Fundamental artifact lifecycle

A transformation initiative is a key artifact type in the meta-model. Thus, its architectural statements can be observed in the enterprise reality assessment and pinned down to a given point in time. Making the transformation initiatives an enterprise artifact is in line with most architecture guidelines such as ISO 42010 (ISO, 2013) or with the Work Package concept in both TOGAF (The Open Group, 2011) and ArchiMate (Band et al., 2016).

- *Principle 2—Changes in the set of alive artifacts are planned ones*.
  This means that an artifact does not become alive or non-alive randomly, but only as a result of purposeful transformation initiatives, whose plans include architecture statements that lead to that state. Therefore, one can foresee the set of alive artifacts at some point in time in the future by consolidating the AS-IS with the architectural statements in the transformation initiative whose completion date precedes the desired moment in time.
- *Principle 3—All enterprise artifacts have a five-state lifecycle: conceived, gestating, alive, dead, and retired.*
  These states and their fundamental transitions are presented in Fig. 19.1.

1. *Conceived* is the first state of existence. It corresponds to a state where the artifact is planned but before its gestation starts. A conceived artifact can also be inferred from statements in the transformation initiatives that have not started yet.
   For example, if a planned initiative holds a statement to put an artifact into production at some point in the future, that artifact is in the conceived state today.
2. *Gestating* is the state when an artifact is being constructed or acquired to become alive. It only differentiates from conceived state by the fact that the transformation initiative that aims at putting the artifact into production has already started. Much like conceived artifacts, gestating ones do not yet exist as an alive element of the enterprise but can be referred by alive artifacts.
3. *Alive* is the state when, as defined before, an artifact can play roles in the organizational transactions and processes to create value. From Principle 2, an artifact cannot be brought into existence as live. It always exists first as conceived of gestating before being alive.

4. *Dead* is the state where an alive artifact is no longer able to play roles in the organizational transactions and processes to create value. As in the conceived and gestation states, a dead artifact may still be referred by alive artifacts. In fact, even if it does not create behavior or value to the enterprise, it may be the target of several housekeeping activities that are necessary after being dead.
   The dead state can be achieved directly after gestating state, without becoming alive. An artifact planned to become alive by a given transformation initiative might never be alive either because the initiative was canceled or because it simply changed plans and decided to no longer put that artifact into production.
5. *Removed* is the state which represents the "post-dead" state where the artifact has no impact on the remaining artifacts. A retired artifact is unable to interact with alive enterprise artifacts. An artifact can move from conception directly to removal when it never materialized in a gestation, meaning that it never went beyond an idea.

- *Principle 4—The TO-BE state precedes the AS-IS state.*
  Since artifacts in the *AS-IS* enterprise state are alive ones, they exist first as conceived ones in the *TO-BE* state, then as gestating ones in the *emerging AS-IS* state, and only after they can exist as alive artifacts in the *AS-IS* state of an enterprise.
- *Principle 5—The emerging AS-IS can be inferred by observing the AS-IS of an enterprise*.
  The emerging AS-IS state differs from the AS-IS state by the artifacts planned to be brought into production by ongoing transformation initiatives. Since ongoing initiatives are alive artifacts, the architectural statements that exist in their plans are in the scope of the cartographer observations of the enterprise reality.

19.5 Enterprise Cartography in Practice

In this section, we present the general steps for enterprise cartography projects that we have been using for many years now, even though initially we did not refer to them as such (Serasa Experian, 2017; Sousa et al., 2014; Sousa & Vasconcelos, 2014).

19.5.1 General Approach

The purpose of an EC project is to deploy processes and tools to produce up-to-date architectural maps of the enterprise architecture with near-zero effort. In the following figure, we can see an illustration of the intended projects, using the *Atlas* (originally EAMS) (Link Consulting, 2017) tool as the KB and architectural maps generator.

The upper part of Fig. 19.2 illustrates a scheduling of projects, where one can see that project *X* execution is scheduled to begin and end at *Tm* and *Tn*, respectively, and project *Y* is expected to end between those dates. The lower part of the figure shows the EA KB receiving information from various sources of information and producing architectural maps. Each map has a time slider that lets you see how its contents evolve over time. By pre-assigning a color to each artifact lifecycle stage, one may see the lifecycle state of the artifacts appearing in the map at any point in time.

Consider now that project *X* intends to add a new artifact and that project Y aims at replacing one artifact by another. Since project *Y* ends before project *X*, the architects of project *X* should take into account the replacement done by project *Y*.

By loading project *X* plans, architects can see the impact of this project in the enterprise architecture in the generated maps. The left corner of Fig. 19.2 shows a single map with the time slider at *Tm* and *Tn* positions, corresponding to project *X* begin and end dates, respectively. The map contents show the impact of all the projects that finish its execution until the date selected in the time slider. So, when the time slider is set to *Tm*, the map shows the component to be created by project *X* and *Y* as under-development (gray), and, when the time slider is set to *Tn*, the map shows created artifacts as alive (light blue) and project *Y* removed artifact as dead (red).

The time slider can also operate in gap analysis mode, in which it presents the evolution of each artifact between the dates selected in the time slider. In Fig. 19.3, we can see the same map in gap mode between the time *Tm* and *Tn*. The left part of the artifacts is colored with the state of the artifact in *Tm* and the right part of the state in *Tn*. This analysis can also be done in depth taking into account the dependency graph of each artifact.

It should be clear that KB cannot be fed only from the promises (plans) of the projects. At least at the end of each project, they should upload their revised plans to the artifacts that were actually produced.

As an example, consider again project X, whose plans include putting into production an artifact on the date *Tn*. This promise is made at the beginning of the project by setting the alive date of the artifact to be created to *Tn*. The loading of this artifact at the beginning of the project (before *Tn*) is always a promise regarding the artifact aliveness because it refers to a future event. But, loading this same artifact on the date *Tn* is no longer a promise but a statement about the reality, for it refers to the present.

Thus, for objects whose plans were fulfilled, nothing needs to be done. In fact, loading them again at the end of the project is an operation that does not affect the state of the KB, since one is only repeating an architectural statement that was already in the KB.

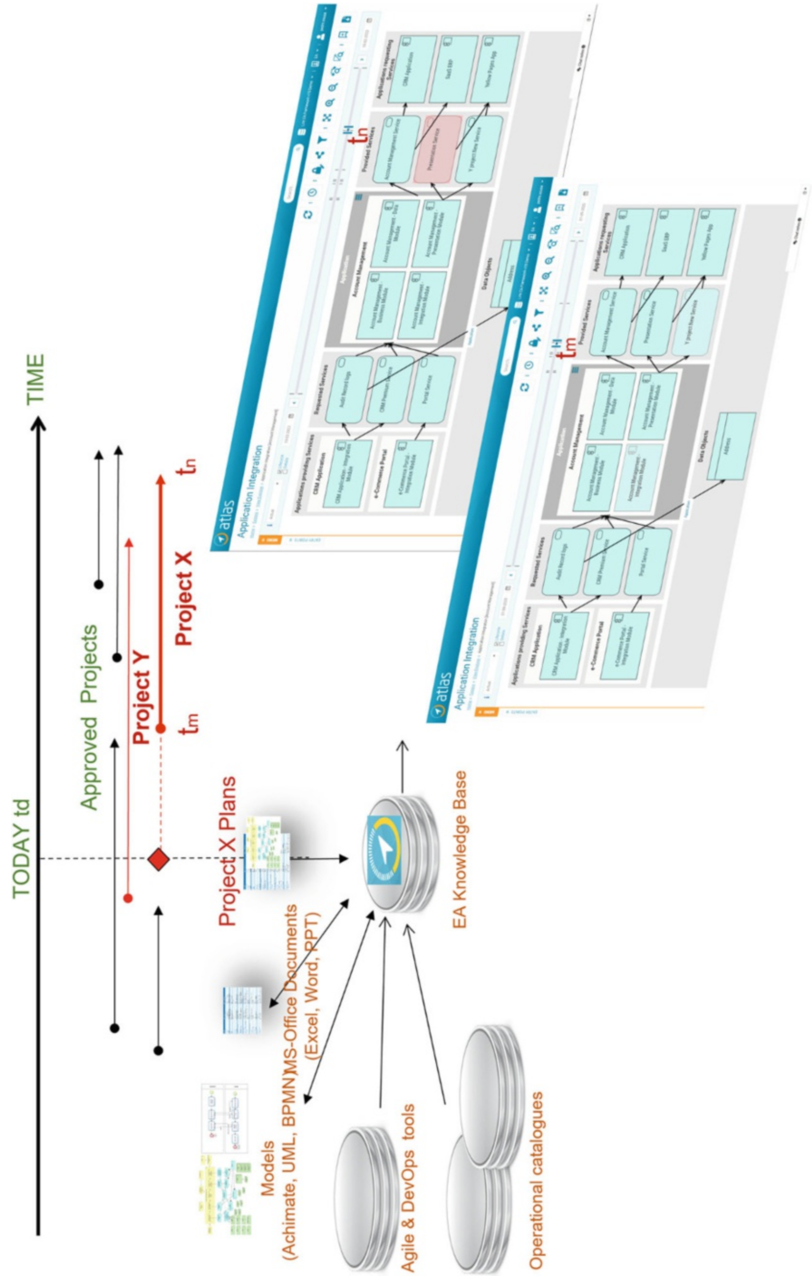


Fig. 19.2 An overview of the intended EC project

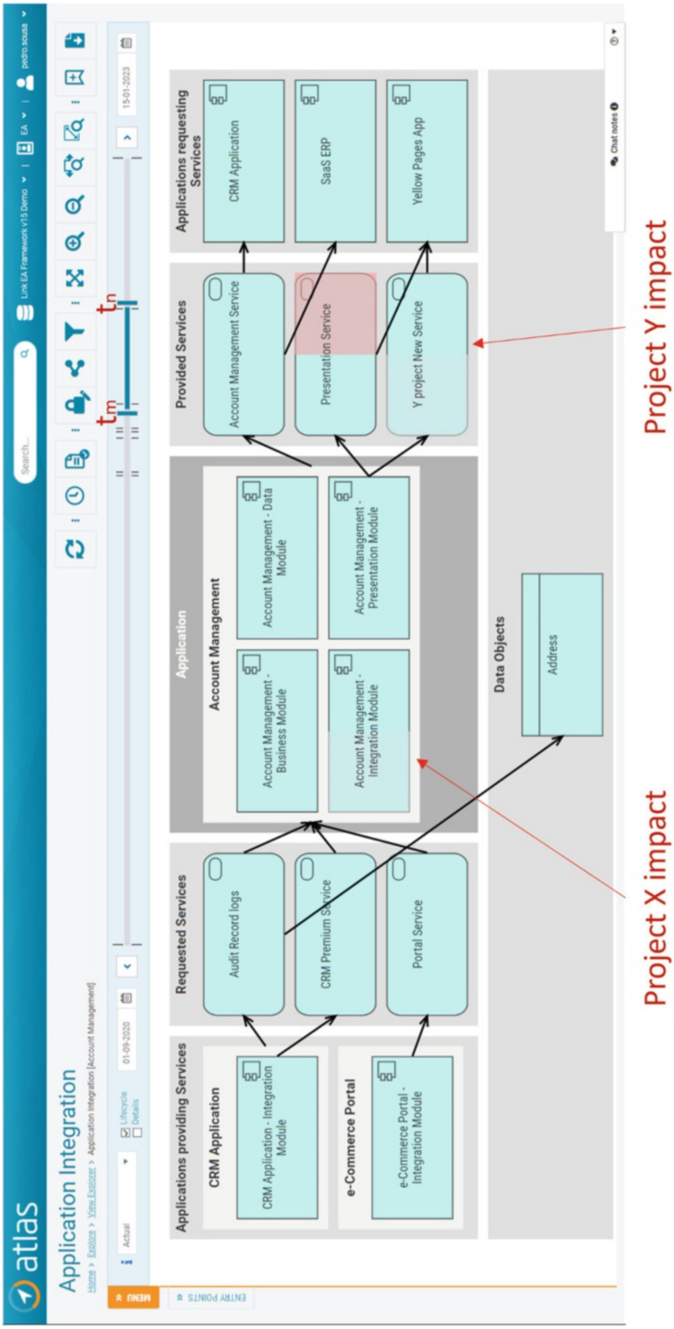


Fig. 19.3 The time slider in gap mode in an architectural map example

For the remaining artifacts, the revision of the project plan leads four possible situations:

- For the artifacts that were made productive without known and explicit plans, one needs to add the artifact with the alive date set to *Tn*, being clear that the object has a date of aliveness but does not have a date of conception or gestation. This does not mean the artifact was not conceived before; it just means their conception information and date are unknown.
- For the artifacts that were to be discontinued in the plan but they remain in production, one needs to clear their date of death.
- For the artifacts that fail to born at *Tn* as initially planned, one needs to set the date of death to *Tn*. This artifact has a date gestating and a date of death but no date of aliveness. It corresponds to what in nature is known as abortion.
- Finally, for the artifacts that were to be decommissioned in *Tn*, but remained in production, one just needs erasing the *Tn* value from the date of death.

In the rest of this chapter, we will focus on the issue of information capture, which has many issues that need to be sorted out, to ensure the project success. We start by presenting a vision of the activities of a typical EC project, followed by the main risk factors associated with them.

19.5.2 A Template for Enterprise Cartography Projects

EC projects are consultancy projects developed together with the client's EA unit. Nowadays, these projects involve an effort between 30 and 40 men * day and have an average duration of 2 to 3 months, with the agenda of the client's senior elements having to be involved the primary factor that determines this period. These numbers do not include the effort in Phase 6 regarding data cleansing.

An EC project is structured around the phases as discussed below.

19.5.2.1 Phase 1—Identify Project Goals

In general, the number of types of artifacts that can be qualified as relevant to include in the meta-model is quite large. So, it is fundamental to bound the problems that we intend to respond to with the EC project, as a way to limit the complexity of the meta-model to the minimum strictly necessary. So, this stage boils down to a set questions whose answers the EC project should provide.

19.5.2.2 Phase 2—Define the Meta-Model

After limiting the scope of the project, we can define the meta-model by identifying the most relevant types of artifacts and their interdependencies required to produce the desired answers. This is an entirely straightforward exercise that starts with a standard meta-model (TOGAF's Content Model and ArchiMate, among others) and extends or reduces it as required. The result is the definition of the knowledge base meta-model and, consequently, the set of possible architectural sentences one needs to capture into the knowledge base to be able to provide the expected answers.

As enterprises evolve, the KB meta-model also evolves and follows the maturity of the enterprise in these domains. Traditionally, evolution is in the sense of extending the meta-model to new areas, with more detail and with more sources of information (Silva et al., 2016).

Thus, the meta-model must be thought in such a way that it can easily be extended by areas, as the TOGAF Content Model allows it, foreseeing enrichment in different areas (The Open Group, 2011).

19.5.2.3 Phase 3—Identify the Best Sources of Information

Once the artifact types and their possible interdependencies are known, one needs to identify the sources of information that allow the collecting of information about the artifacts as effortless as possible. For each artifact, we identify the best information sources to capture state changes in the artifact's lifecycle. Depending on the artifacts, one might consider a broad set of information sources.

Structured repositories' systems are natural candidates for alive artifacts, for example, products' price list, staff in the payroll system, application services in the services bus platform, applications in the helpdesk support system, and so on. Such systems can be a trustful source of the AS-IS, but they rarely hold information regarding the emerging AS-IS. An exception are the systems supporting the IT developing and deploying process, namely, those supporting Agile development paradigm and DevOps tools, such as DevOps (Jenkins, 2017) or Sonar (Sonar, 2017). Another exception is the test and quality environments since artifacts being tested will likely become alive.

Plans for transformation initiatives are possible sources of the *TO-BE* architecture. Such plans are often found either in office documents or in models in some structured notation/language, such as UML, ArchiMate, BPMN, or others. In the first case, Office documents hold mostly natural language statements or images that cannot be automatically processed, for example, "the log of a board meeting where it was decided to start selling the product X in 6 month time" or "to replace a system X by system Y in the next eight months." In the second case, the meta-models of such notations used are likely to be different from the one used in the KB, both in the artifact types and the relationship types. Thus, it is necessary to define the mapping rules to use during importation and exportation. A real example was the case were the KB has the *Platform* and *System Software* artifact types. However, the models were produced in ArchiMate, where artifacts of both *Platform* and *System Software* types were modeled as instances of *System Software*, being distinguished by the fact that *Platforms* artifacts always aggregate more than one *System Software* artifact and the latter does not aggregate other (*System Software*). Thus, one had to load them as *Platforms* or as *System Software* depending on the defined rule outcome.

19.5.2.4 Phase 4—Structure the Processes and Tools to Capture Information

EC projects are usually conducted by the enterprise's architecture unit with the participation of other units or roles, such as project managers, technical leaders, project architects, or solution architects, acting both as a consumer (stakeholders) and as a producer. In fact, they consume architectural maps and produce the information necessary to generate them.

In our experience, the participating teams are quite skeptical about the possibility of generating updated maps without a significant effort to capture the necessary information. Thus, one has to ensure that the EC project does not require the remaining actors to do more work than they already have today. Our approach is, firstly, to look at the tools the teams use to produce and register architecture-relevant information and propose the necessary changes so that it can be automatically loaded into the knowledge base. This approach allows the loading of information and the production of maps for the teams to use with a minimum negative impact on them. For example, in an enterprise that uses MS Office documents for internal project presentation and description, we can include tables in those documents to automatically load the information (the architectural statements) about such project. Figure 19.4 shows an example for project *X* (used as an example earlier), with a first table for general data and a second for application components affected, according to the map shown in Fig. 19.3.

Later, after the remaining teams are using the maps generated with the information collected, we can propose more structured ways of obtaining the required information, either through the models or direct introduction in the KB.

19.5.2.5 Phase 5—Define and Configure the Architectural Maps

The definition and configuration of the intended maps are fundamental for their adaptation to the needs of the various stakeholders. The maps generated are hierarchically structured since, in our experience, they tend to be one of the preferred ways to visualize architectural maps. This is also confirmed in Roth et al. (2014), where such maps are named clustered maps. Despite the hierarchical structure, the map generator can be configured to produce all ArchiMate viewpoints (Band et al., 2016). As an example, besides the integration map presented in Fig. 19.3, we show a layered map in Fig. 19.4, corresponding to the ArchiMate Layered Viewpoint (Band et al., 2016).

It is important to clarify that not everyone wants to deal with the complexity of the KB meta-model. Therefore, our architectural maps act as a *View* in a relational DBMS, allowing navigation and exploration under a perceived meta-

Project X Architecture Data

1 – Project Card

Name	Description	Project Manager	Start Date	End Date
Project X	...	John	tm	tn

2 – Application Components

Name	Desciption	Component Of	Action	Dates
Account Management Integration Module	...	Account Management	Create	01-02-2023

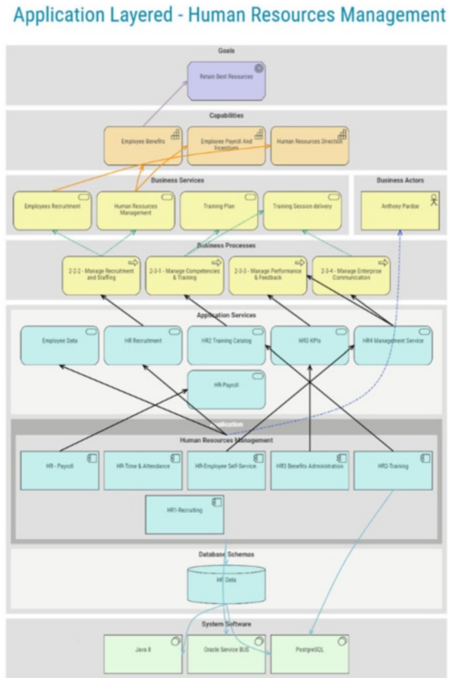


Fig. 19.4 On the left: an example of the Word file to be imported. On the right: an example of a layered architectural map

model specifically designed to each stakeholder in that map. A typical situation is to reduce the number of entities one need to cross to find the dependency between two concepts, showing only the derived relations instead of the existing ones in the meta-model. For example, in a KB with the ArchiMate meta-model, users of a given architectural map could experience and navigate over simplified meta-model where business processes are directly related with applications without going through the services and functions in between.

19.5.2.6 Phase 6—Populate the KB with an Initial Baseline

The initial loading of the KB is usually done by importing other systems or even existing Excel sheets that the client may have with relevant information. However, loading AS-IS can be a very challenging step in enterprises with a low maturity level on these subjects. As a matter of fact, if enterprises had their AS-IS ready to be loaded, they probably would not need an EC project to start with. The main problem is usually in the quality of data of the information to be loaded, as well as the lack of homogenization at the conceptual level. Concepts such as information system, application, service, and platform are among the most complex to grasp.

The effort of loading information into KB is tiny. On the contrary, the work needed to prepare the information to be loaded (data cleansing) can be very high. In practice, there are only a few catalogs that are to be loaded, in a Pareto law approach.

Obviously, the more complete and rigorous the KB is, the more useful and savings one gets from the EC solution. Therefore, this initial loading is often perceived as an essential step to the success of the project. In our view, this initial upload is actually optional and should not be placed as a critical success factor for the project. We argue that the true critical success factor is the ability of uploading and updating the KB on a project-by-project basis. The more the projects, the more rapidly the KB will be populated and kept updated. So, in the approach presented, we will consider that the KB is loaded incrementally, project by project. This task can be done in parallel with the architectural maps configuration.

19.6 Enterprise Cartography Cases

19.6.1 Case 1—Enterprise Cartography as a Monitor to the Progress of the Strategic Roadmap

The Serasa Experian (hereafter designated as Serasa Serasa Experian, 2017) is among the biggest credit bureaus in the world outside of the USA, holding the largest database of consumers, companies, and economics groups in Latin America. The company is involved in most credit decisions taken in Brazil, corresponding to 4 million queries each day, done by over 400 thousand clients. With a strong transformation plan, Serasa needed tools to monitor the progress of the strategic roadmap.

Serasa had a strategic plan for the entire IT that is embodied in roadmaps for the key business and technologic areas of the company. These roadmaps set the targets to achieve for each area over the next 5 years. These targets are defined regarding a maturity level measured on a five-level scale (Maier et al., 2012). At the end of each year, each area submits a plan with the initiatives proposed for the coming year. If approved, these initiatives become real projects. Then, 1 year after, each area evaluates the progress achieved and updates the roadmap accordingly. Roadmaps were manually maintained in many Microsoft Excel sheets, requiring a significant effort to update the roadmaps according to the results of the projects.

Our task was to design, plan, and implement an instrument that enables the integrated management of architectural views, project results, and the roadmap progress. More specifically, this tool should allow Serasa to:

- Have an integrated and up-to-date view of the *AS-IS* of business and IT architecture, by the consolidation of information from different sources
- Have a view of the emerging *AS-IS* state of the business and IT architecture based on the foreseen results of ongoing and planned projects
- Have the ability to identify the progress of each roadmap updated according to the project's progression

Following our project template, the different areas had identified 82 relevant questions, leading to a meta-model with 32 types. Each of the areas provided the information to be loaded in the KB, both regarding the roadmaps and the reality of the enterprise at different levels: processes, skills, software, and infrastructure. The architectural maps were defined in sessions with each of the areas.

The immediate result of this project is to provide to the various areas of IT a consolidated and updated view of the architecture of Serasa: consolidated, because it results from the information provided from different areas, and updated, because it results from information sources maintained by the various areas with transparent processes in an automated manner. Since each group shared its information with other areas, the project entails a substantial transformation in the enterprise, as it homogenized languages and tools. In this regard, the KB meta-model is a valuable asset because it identifies the concepts familiar to the various areas. A more detailed description of this case was published in co-authoring with the architecture team of Serasa in Sousa et al. (2011).

19.6.2 Case 2—Supporting a Prescriptive Approach for Enterprise Architecture to Reduce IT Spending

This case presents the setup of an IT architectural KB for the public bodies of the Portuguese central government. The Agency for the Public Service Reform

(AMA, 2017) is a public institute that has the mission to roll out modernization initiatives in the administrative and regulatory areas focused in the simplification and optimization of public services, within the policies defined by the government.

AMA had the task of achieving an EA at the Portuguese central government, aiming at a more rational IT spending within the public organizations. The project aims at the creation of a central KB to allow a descriptive and prescriptive approach for EA (Greefhorst & Proper, 2011a; Hoogervorst, 2009) over the entire set of enterprises belonging to the Portuguese government.

The adoption of a simple KB meta-model and intuitive maps seems critical to achieving an active EA over such a broad scope. Even so, the meta-model contains 24 artifact types, from business to infrastructure domains. The KB was initially loaded with the information from a previous assessment of about 200 public agencies from 7 different ministries. About 87K devices, 15K servers, 14K databases, plus 20 other artifact types were loaded at this stage.

After the initial loading, the KB is updated on a project-by-project basis, since public bodies are required to submit to AMA the architecture of the IT projects over 10K €. Each submission is validated against a set of regulations applied to public bodies. Such ruling aims at the implementation of a prescriptive and normative approach for EA by promoting the adopting of a reference architecture and design principles and the sharing of available resources within the public administration. This project was presented in co-authorship with the AMA architecture team in Sousa and Vasconcelos (2014).

The project is still ongoing in some ministries that followed a federative view of the KB, where the ministry has its KB with a meta-model enriched to address specific concerns in that domain, namely, health and defense. First, the public bodies feed the KB of their ministry, by submitting the proposed architectural scenario, and then, upon approval against the ministry's own rules, it can be sent to AMA for its approval and KB update.

19.6.3 Case 3—EC as the Provider of Organizational Views for ITIL Processes

The IT Department of the Portuguese Defense (CDD, 2017) (or CDD for short) provides IT services and support to Portuguese Defence forces. CDD adopted ITIL (Axelos, 2015) processes to improve the quality of its support services. Although the processes were defined, there was still much to improve in the description of information and concepts among the different teams involved in the execution of these operations. Each technical area managed information relating to their respective functions and technical competencies. This information was not integrated or even shared, as each technical area had its own database with related information. One example was the lack of a coherent service catalog, since each technical service area had identified their services at different levels of granularity and failed to integrate them into a single one.

The purpose of the project was to provide the CDD with a KB that was common to the various technical teams, homogenizing terms and concepts and ensuring the coherence of information used in the execution of ITIL processes. To structure the information sources, CDD had to clarify the semantics of each concept and their relationships. They identified the information sources from each technical area in the meta-model. From the identified information, CDD developed the architectural layers: infrastructure architecture involving the servers and networking views, data architecture with databases and instances, and applications architecture.

The used meta-model (Gama et al., 2011) was based on service orientation, resulting in a meta-model with 16 concepts. Hence, they identified the information sources from each technical area in the defined meta-model. Together with each of the 4 technical areas, CDD identified, clarified, and selected 12 primary sources of diverse information that need to be integrated and loaded into the repository. ArchiMate models with the ITIL processes and resources were also loaded into the KB. Nowadays, the information serves the CDD interest as a whole instead of relating to a unique technical area. The achieved results allowed the CDD to provide consolidated and updated information to the various technical areas through views of the architecture.

The project was fully planned and executed by the CDD architecture team, including the design of the KB meta-model, the configuration of the loading rules from the CMDB and ArchiMate models, and the design and configuration of the architectural maps made available to the technical teams (Gama et al., 2015).

19.6.4 Case 4—Architectural Maps in Enterprise Intranet and Wikis to Support Development Teams

The client referred to in this case study is a retail bank that operates in the Iberian Peninsula and serves almost 2 million customers (individuals, companies, and institutions) through its multi-channel distribution network comprising around 650 retail branches.

The bank already had an EA portal on their intranet to support the registration of information regarding application and technology architectures. The intranet portal included a large number of features such as a wiki, with a considerable amount of articles that described the processes, system architecture, and which established an entry point for all documentation deemed more technical.

With this project, the bank was addressing several challenges. The first one was to develop the bank's EA practice, by improving the EA intranet portal to support the communication and awareness of the architecture between all stakeholders, from infrastructure teams to business areas.

The second challenge was the need to effortlessly maintain the architectural representations. The bank had an experience in EA, since it went through the process of using a modeling tool to produce representations and models of its IT architecture. The approach followed was based on a central repository holding all models, each designed manually with an EA tool. This approach required a substantial effort to keep them up to date, in particular if one considers the consolidation of the models produced in different projects into a single and enterprise-wide view of the IT landscape. So, the maintenance effort to keep the EA representations was a key concern of this project, and the bank stated that "if an architectural view cannot be generated automatically with up-to-date information, then it cannot be presented in the EA portal." The third challenge was the need to keep audience's learning curve regarding the usage of the architecture intranet portal as lean as possible. The fourth challenge was the integration of the SoA (service-oriented architecture) initiative in the enterprise architecture initiative.

The project aimed at the implementation and deployment of an EA solution fully integrated into the existing intranet, so that all but back-office housekeeping activities could be done in a seamless manner on the bank's intranet. The bank also wanted to address other architectures in domains they considered relevant, even though they did not have much information to start with, namely, business, information, user experience, and normative architectures. Both information and business architectures were domains that the bank planned to address in this project, starting with the definition in the meta-model, where it became clear the level of detail needed and how to establish the relations with the remaining architectures.

The user experience architecture concerns with the definition of patterns to allow the same concept (client, address, account, and so on) to appear to the user with the same paradigms, regardless of the interface and application where such experience occurs. Finally, the normative architecture structures all information regarding architectural principles, rules, best practices, and technical articles among other information. Articles represent a real knowledge base, and a wiki platform allows live interaction with end users.

The architecture views were generated on the fly and embedded in a seamless manner with the bank's intranet portal. Architectural views are the entry point to access any IT documentation and knowledge kept in wikis. The information necessary to generate the architectural maps had to be harvested from various sources, such as Microsoft SharePoint (MS, 2017) and Oracle Enterprise Repository (Oracle, 2017), among others.

This approach enables not only the role of the architecture views in understanding the complexities of the business and the IT underneath but also a better and more efficient collaboration between the different stakeholders. This has a positive feedback on the use of the EA internet portal as cooperation and communication platform, and it plays a fundamental role in people communication and collaboration. This resulting case has been published in Sousa et al. (2014).

19.6.5 Case 5—Clarification of the Concept of IT Application

The case presented here focuses on the issue of standardization and structuring of project submission documents, addressed in “Structure the Processes and Tools to Capture Information” presented in the template as discussion in Sect. 19.5.2.

The IT industry has multiple terms with more or less obvious meaning; words such as application, information system, and business solution tend to be used indistinctively in various scenarios and can become the source of confusion in many situations. The existence of an application/system/solution catalog is common in many enterprises, and it is considered a fundamental element that draws attention from the business, the IS, and the IT area.

With over 20,000 employees spread throughout the world and with more than 4 million clients, the Caixa Geral de Depósitos (CGD, 2017) (CGD for short) is the largest Portuguese financial group, encompassing banks and insurance and medical companies. The Sogrupo - de Sistemas de Informação (or SSI) is the company responsible for supporting all information technologies in the group. SSI has approximately 900 workers. The SSI has an ongoing EA program which enforces both the descriptive and prescriptive perspective of architecture. It follows TOGAF in its main ideas.

The project in question had the mission of clarifying the application concept to the primary stakeholders across the SSI and to restructure the key application technical architecture accordingly. With over to 500 so-called applications to classify, the goal was not to provide formal and theory-driven definitions but instead to establish a concept that was useful and very pragmatic to be used by the multiple stakeholders involved.

As the result of the project, one realizes that the term application embodies aspects of the three different perspectives (business, information systems, and infrastructure) and that it would be better to decompose the application into three distinct concepts, each covering only one perspective and with a single responsibility.

- At the level of the business, the concept of application relates to functional issues. Here, the term business solution was the one that garnered more acceptance. The person in charge of a business solution decides only the functional aspects and who can access.
- At the level of information systems, the application concept unfolds in a set of components to support the defined requirements. The person in charge of an application focuses on the architecture and engineering of these components.
- Finally, at the level of the infrastructure, the concept of application materializes in the capacity of execution of the defined components. Here, the term chosen was platform, and the person responsible for it is fundamentally concerned with ensuring the continuity of its operation.

As a result, the architecture and catalog of an application may lead to three distinct artifacts, each with its domain and concern. But, of course, there is a many-to-many relationship between the artifacts of these three levels, and an application component can support different business solutions and a business solution to be supported by many application components. The same applies for applications and platforms. This work is published in co-authorship with the architecture responsible at CGD in Sousa et al. (2012).

19.7 Discussion

Although there are many complex issues regarding the generation of architectural maps automatically and even on the fly, we keep the discussion focused on issues centered on information capture. We select the following topics:

- Start using the EC solution with the empty KB.
- Projects as information sources.
- Corrective and evolutive maintenance.
- Business cartography.

19.7.1 Start Using the EC Solution with the Empty KB

In cases where the KB is not loaded from the beginning, it will not be able to feed the transformation initiatives with information necessary for their planning and impact analysis. In this situation, the projects feed KB instead, both with the *AS-IS* and the *TO-BE*.

Whenever a project proposes to do some form of integration between the new artifacts and those existing in the enterprise, the analysis and estimation of the time, cost, and risk associated with that integration require some assessment of *AS-IS*. If this information does not exist in the KB, the effort to survey the *AS-IS* should be included in the project planning.

Therefore, at first, while the KB is practically empty, the projects have to consider the effort and time to do the necessary assessment of the *AS-IS*. However, as KB becomes loaded, projects become information consumers and take full advantages of the KB and related architectural maps to support their analyses and planning. The faster the enterprise changes, the more project will be loading the KB. After a few iterations, only *AS-IS* systems that have never been involved in any project will remain unknown.

19.7.2 Projects as Information Sources

As mentioned before, in our approach, the KB should be uploaded project by project, to ensure that its impact is present on the maps and in this way is known by all. We argue now that each project should provide information in two key moments:

- The first one is when the project architecture already contains the main elements to become productive (alive) or decommissioned (dead), consequently, they should be loaded into the KB so that its impact on the other projects can be known and evaluated. The information uploaded is just a promise that will or will not come to fruition. This moment usually occurs when the project is evaluated for go/no-go decision-making since it is at this time that it is necessary to assess the costs, times, and risks associated with its execution.
  The reader could argue that architecture is substantially independent of cost, time, and risk issues. But our experience is just the opposite: the more rigorous the evaluation of cost, time, and risk, the more complete and detailed the architecture is known. In fact, this finding is perfectly aligned with the most common definition of architecture, in which it includes knowledge that is necessary for the maintenance and transformation of enterprises (Zachman, 1997; Op 't Land et al., 2008a; Dietz, 2006; ISO, 2013). Consequently, architectural knowledge is a necessary asset to estimate costs, times, and risks associated with transformation initiatives. The close relationship between the EA and the management key variables associated with transformation is a reliable driver to force project leaders to provide relevant architectural information early in the project lifecycle. Furthermore, assessing the impact on cost, time, and risk management is a way of determining the appropriateness of an artifact in the enterprise's architecture. Regardless of the theoretical discussion that this line of thought can take us, the more impact an artifact has on those three variables, the more likely it will be identified and known in the plans of a project.
- The second moment is at the end of the project, since at this time, the artifacts and relationships created or eliminated by the project are known for a fact. The importance of updating the project plans and reloading them in KB is obviously to differentiate what are plans (promises) from what is the reality (*AS-IS*).
  As seen earlier, artifacts that were not initially foreseen should be created with the gestation date and alive coinciding with the beginning and end of the project, respectively. These dates are automatically filled in, during the loading of the project information according to the configured rules. These artifacts are distinguished from those that were originally planned because they do not have an existence before the project. By contrast, artifacts planned from the beginning have an existence (conceived) before the start of the project.
  The artifacts that were planned but not put into production at the end of the project become dead even though they have not gone through the alive state. It is a situation of abortion, in which one passes from gestation directly to death. It is very important to record this evolution, for as long as they were in the gestation state, these artifacts were alive in the enterprise emerging *AS-IS* and therefore may have influenced decisions that are only justified in the assumptions of the existence of these objects. For example, consider that project Y made decisions based on the assumption that project X, in progress, would create a particular

artifact. If this will never be brought into production, it is important to keep such fact in architectural maps so that project Y can justify the decisions made.

A second reason to load the project information at the project completion concerns with updating the dates related to objects' lifecycle automatically. Since, most probably, projects fail to end at the dates initially planned, it is important to automate the setting of the dates of the artifacts based on the dates of the projects that create or discontinue them so that they can become correctly with no effort.

19.7.3 Corrective and Evolutive Maintenance

The assurance that the information residing in the KB remains up to date is supported by the assumptions of Principle 2, in which it is assumed that *AS-IS* is portrayed in some way in the *TO-BE* view. If the evolution of the enterprise is fundamentally based on projects, the KB is kept up to date. However, it may happen that the evolutive and corrective maintenance also changes the enterprise architecture. Since maintenance actions tend to follow a lighter approach than projects, one must assure that architecture changes are in some manner propagated to the KB.

The simplification of the process involves uploading the changes made at the end of the action, thus omitting the loading phase of the plans. Assuming that maintenance actions are mostly functional, thus do not cause changes to the architecture, the need to report changes to the architecture is more an exception than the norm. Even so, it is important that maintenance teams have a list of the types of artifacts that are being tracked in the context of the EC. A simple list of meta-model artifacts (typically with 12 to 20 entities) allows maintenance teams to mitigate this issue.

19.7.4 Business Cartography

Throughout this chapter, our focus and example have been mainly on the evolution of IT, although we consider our approach to mapping to be generic and also applicable to business transformations. But there is indeed a crucial factor that distinguishes transformations at the level of business from those at IT. Project management is a mature discipline in most enterprises, involving virtually all changes in the IT architecture. All enterprises seek to know the time and risk costs of IT projects, thus sustaining our approach. On the contrary, transformations at the business level often happen without a project being formed. A typical example is the creation of a new department, which can arise only from the course of a series of meetings without any *TO-BE* document or explanatory model. This is a difficulty that needs to be mitigated, for example, by defining structured templates for meeting minutes so that they can be uploaded and can be valid sources of information.

19.8 Conclusions

Having discussed the most critical points of information gathering in enterprises, in this conclusion, we explore the role of EC in governance and organizational self-awareness (OSA) (Tribolet, 2005; Abraham et al., 2013b; Tribolet et al., 2014b,a; Tribolet, 2014).

The formal and rigorous emergence of the discipline of enterprise engineering, scientifically based on systems theory (Wikipedia, 2019c) and solidly grounded on social sciences theories, has provided a fertile ground for interdisciplinary research involving key aspects of the full lifecycle of dealing with enterprises as complex artifacts populating and serving the human habitat. The general engineering life-cycle of enterprises involves aspects of architecture, engineering, governance, and dynamic steering of the time transformations that always occur in all living entities. The present paper focused the role and relevance of EC and enterprise cartographers and clarified the distinction between those and EA and enterprise architects.

In a previous paper entitled "The Role of Enterprise Governance and Cartography in Enterprise Engineering" (Tribolet et al., 2014b), a more general view of EC was presented, by taking into consideration the view of enterprises as organized complexities, whose existence at any point in time is the result of the dynamic synergies between purposeful, intentional, goal-oriented designs, such as those generated by EA, with the emergent, opportunistic, individualistic, non-orchestrated, and often incoherent and chaotic actions that naturally occur as the result of the free will of the human actors that are, after all, the true core of any organization. We conclude this chapter by connecting the dots between purposeful EA designs leading to intended pre-determined enterprise transformations and the realities of real-time events, with the emergent phenomena that in fact end up being the reality of the enacted enterprise as de facto is.

The key to connect, integrate, and make sense of these two powerful reality drivers, one more top-down and the other clearly bottom-up, is precisely the capability to sustain, at the systemic organizational level, commonly shared representations of the *AS-IS* of perceived reality as accurate as possible. We have coined the concept of "organizational self-awareness (OSA)" to denote this systemic capability, whose fundamental technical basis is precisely the one addressed in this chapter: enterprise cartography.

By collecting the sensory data of the enacted reality as it happens and is perceived by those that are involved and immersed in the respective action context; by feeding such data into semantic organizational models capable of being read, understood, and interpreted by the organizational actors, according to their multiple and specific points of view; and by providing the dialectic contradictory mechanisms to challenge, verify, and consolidate the sensory data being reported as true and systematically building a verifiable and verified explicit knowledge base of the enterprise reality and its dynamics on the fly, one is providing to all involved—from the lowest level of operational workers to the top level of enterprise executives and strategists—a formal, updated, common view of reality which is key to sustain an essential feature of any organization: meaningful, informed, objective, explicit, non-ambiguous means of support to human-to-human communication.

The capability of an organization to be aware of itself at any point in time, in space and in context, is the equivalent to what we, single humans, do possess as most valuable to steer ourselves in life: our own self-awareness, on top of which we build higher-order levels of self, such as conscience, values, and ethics. EC is, at the very root, the basic mechanism for us to equip modern-day organizations with the artifacts we need to deal with the tremendous challenges we face today: exploding complexity, increasing speeds of interactions, gigantic volumes of information flows, shorter and shorter decision times, and increased requirement to build up on-the-fly agile and realizable answers to previously unknown challenges.

Acknowledgments This work was partially supported by the project 11419 (IT-Atlas) of the Portuguese PO CI operational program. The authors would like to thank André Sampaio and Ricardo Leal from Link Consulting SA that contributed for most of the projects presented here.

Chapter 20
A Conceptual Model for Expressing the Evolution of Enterprise Architecture Models

Nuno Silva, Pedro Sousa, and Miguel Mira da Silva

Abstract The role of enterprise architecture (EA) is to assist enterprises in achieving a successful strategy execution by conducting enterprise analysis, design, planning, and implementation. EA projects enable enterprise transformation by altering the current state of the organization's architecture model, which, in turn, impacts the digital resources and capabilities of the enterprise. Therefore, to support digital transformation, a sound conceptual understanding of the architectural aspects leading to the evolution of EA models is required. This chapter introduces a conceptual model which can be used for describing EA model evolution, hence clarifying misunderstandings concerning the evolution of EA models. Design science research was the methodology of choice for addressing this research.

20.1 Introduction

Enterprises are complex adaptive socio-technical systems that continuously consider and pursue a fundamental change to maintain or gain a competitive advantage (Rouse, 2005a). This change is nowadays more and more connected to digital transformation (DT), i.e., enterprise transformation impacting the enterprise's digital resources and capabilities. Therefore, in order to address the underlying challenges of organizational change, DT initiatives are conducted with the purpose of seizing fleeting market opportunities and reconfiguring the business to be in line with the shifting value propositions.

The enterprise architecture (EA) discipline describes a set of principles, methods, and models used in the design and realization of an organization's structure, business processes, information systems (IS), and infrastructure (Lankhorst et al., 2017),

N. Silva (✉)
NOS SGPS, Lisbon, Portugal
e-mail: 9nmcs0@gmail.com

P. Sousa · M. M. da Silva
Instituto Superior Tecnico, Lisbon University, Lisbon, Portugal
e-mail: pedro.sousa@linkconsulting.com; miguel.miradasilva@gmail.com

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_20

which has been suggested to facilitate DT. The focus, however, has traditionally been on process standardization and integration rather than on continuous adaptation to the changing business and technological landscape (Korhonen & Halén, 2017). Hence, for EA to have the desired impact, more adaptive conceptualizations of EA that address the requirements of the new digital environment are called for (Korhonen & Halén, 2017).

Moreover, a fundamental aspect of the relation between DT and EA, as well as EA management (EAM) in particular, is the evolution of EA models with the purpose of addressing strategic organizational drivers that motivate organizational change. By consolidating architecture decisions along with migration and implementation plans, the management and evolution of the EA model take as input a consistent baseline of information concerning the *as-is* landscape of the organization and from there plan the envisioned *to-be* state.

In line with the above, a particular issue has to do with the lack of architectural understanding, namely, regarding the underlying concepts that motivate the evolution of the architectural models. As a means of clarifying such understanding and thus providing support throughout the EA model evolution task, a conceptual model was proposed, framing a set of concerns regarding EA model evolution. The artifact was then implemented as an EAM tool add-on and used inside a European financial organization in the scope of an EA project.

Design science research (DSR) proposed by Peffers et al. (2007) was chosen as the research methodology to be used throughout this research. The methodology aims at creating and validating IT artifacts intended to solve identified problems from an organizational scope. This IS research methodology is widely adopted by IS researchers due to its appropriateness to research that seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts (Hevner et al., 2004). DSR is iterative and composed of six well-defined activities (Peffers et al., 2007):

- *Problem identification and motivation*—In this first activity, the specific research problem is defined and the value of a solution justified.
- *Define the objectives for a solution*—The objectives of a solution, whether quantitative or qualitative, are inferred from the problem definition and the knowledge of what is possible and feasible. Furthermore, the objectives should be inferred rationally from the problem specification.
- *Design and development*—Creation of the proposed artifact. “A design research artifact can be any designed object in which a research contribution is embedded in the design.” In this activity, the artifact’s desired functionality and its architecture are devised and the actual artifact implemented.
- *Demonstration*—This activity demonstrates the use of the artifact to solve one or more instances of the problem. Examples of demonstrations could be experimentations, simulations, case studies, proofs, or other suitable activities.
- *Evaluation*—In this activity, the artifact supporting the solution to the problem is observed and measured. The evaluation of the artifact involves comparing the objectives of a solution to the empirical results from the artifact’s usage in

the demonstration. Considering both the nature of the problem and the artifact, evaluation can take many forms. For instance, it could include a comparison of the artifact's functionality with the solution objectives, objective quantitative performance measures, satisfaction surveys, client feedback, or simulations. Conceptually, such evaluation could also include any appropriate empirical evidence or logical proof.
- *Communication*—In this final activity, the problem and its importance are communicated as well as the artifact, its utility and novelty, the rigor of its design, and its effectiveness to researchers and other relevant audiences.

The remainder of this chapter is structured as follows: Sect. 20.2 provides an overview and discussion of the literature addressing and motivating the research. Afterward, Sect. 20.3 presents the conceptual model for expressing EA model evolution. In Sect. 20.4, a demonstration of the artifact in practice is presented. Section 20.5 describes the evaluation approaches used to validate the proposed solution, and, finally, Sect. 20.6 presents the research conclusions and themes for future work.

20.2 Related Work

Enterprise transformation is known as a set of initiatives that change the organization's domain (its structure and dynamics) from its current *as-is* state to a predetermined *to-be* state. Keeping up-to-date architectural views with organization changes has brought up issues in EA projects, as well as misalignment between both business and IT spectrum. Approaches presented by Sousa et al. (2009), Tribolet et al. (2014b) proposed that all organizational artifacts have four fundamental invariant states (gestating, alive, dead, and retired) in their life-cycle as if each organizational artifact was a living entity, instead of using a versioning schema that handles the evolution of architectural views.

Hence, an organizational artifact is conceived as the future result of an EA transformation project, thereby entering the gestating state. Such artifact remains in gestation throughout the project's timeline becoming alive after the project ends. The artifact dies after the completion of a decommissioning project or when a transformation project is cancelled. In the end, the dead artifact is then retired when a retirement project explicitly removes it from the organizational infrastructure.

Furthermore, when discussing and clarifying the topic of EA evolution in the scope of enterprise transformation, the analyzed literature identifies four main concepts:

- *IT Project*—perceived as the enabler of EA transformation (Buckl et al., 2009b; Sousa et al., 2009; Wegmann, 2003) in which, given a baseline architecture, a target architecture is envisioned, planned, and documented to satisfy the strategy and vision of the enterprise

- *Driver*—"represents an external or internal condition that motivates an organization to define its goals and implement the changes necessary to achieve them" (Band et al., 2016)
- *Transformation*—"represents an intended architectural change" (Buckl et al., 2009b). Understood as a set of changes that alter the life-cycle of EA description elements
- *Life-Cycle*—"prescribes the stages and manner in which the contents of a conforming architecture description are to be produced" (ISO, 2013), i.e., defines the different states that EA description elements (entities and relationships expressing the EA) may have over the period of time in which they are part of the EA

As enablers of DT (and consequent organizational change), the aim of IT projects is to define and implement the strategy guiding the enterprise toward its evolution (Wegmann, 2003). These projects report back the changes in a normalized form, by providing a work plan to produce the intended architectural transformations and rules that together enable the evolution of EA models (Sousa et al., 2009).

An issue that typically compromises the success rate of IT projects is the complexity of EA models. EA models' explicit traceability across multi-layered elements representing the various levels of the enterprise (Wegmann, 2003) reinforces their tendency to focus on holistic views of the enterprise rather than on the subset of relevant artifacts for a given project, thus making the task of reading and updating these models more complex (Sousa et al., 2009).

The role of transformation in keeping EA models updated involves essentially two points (Sousa et al., 2009):

- Information about what has changed
- Rules to update the models accordingly

Managing each transformation requires awareness over what has been changed and the compliance of those changes with the rules in which EA models can be updated. Hence, robust change management processes and procedures are essential to maintaining the architecture of the enterprise (Kaisler et al., 2005).

Existing approaches that deal with EA transformation concentrate their efforts on the architectural concepts rather than on the project itself (Buckl et al., 2009b), resulting in a sequence of architectural snapshots indicating the architecture *to-be* state at a certain point in time. Moreover, these concept-based approaches do not express meaningful information like how were the EA models changed (i.e., what transformations were applied to the EA description elements composing the EA model under a specific IT project), who were the people with concerns about those changes, when those changes happened, and the motivations behind them.

Buckl et al. (2007) further identify the problem concerning the complexity of evolving information models (meta-models) by using existing models. EA projects seem to prioritize the development of new models instead of improving existing ones as it simplifies the project execution.

20.3 Research Proposal

In this section, the proposed conceptual model for expressing EA model evolution is presented. The model specifies the concepts and relationships that must be taken into account when analyzing and discussing evolution-specific aspects of EA models.

20.3.1 Objectives of the Solution

With this approach, we aim to provide a means to express and reason upon the underlying elements concerning the evolution of EA models. Another objective is aiding in the communication and decision-making toward EA evolution. Finally, the model aims at addressing a set of evolution-specific concerns.

A set of concerns were inferred from a landscape and project portfolio management survey presented in (Buckl et al., 2009a). The reader should be noted that the questions presented in Table 20.1 are a subset of the entire catalogue described in (Buckl et al., 2009a); hence, only layer-independent questions were regarded since the model's purpose is to cover the evolution of EA from a multi-layered perspective and not just a specific domain in isolation.

20.3.2 The Conceptual Model

The conceptual model was designed based on both Table 20.1 concerns and the evolution-specific concepts retrieved from the literature analysis. The model consists of eight core concepts and relationships that extend the EA model with evolution-specific aspects concerning the *to-be* state of the organization's landscape. In order to ground the proposed viewpoint in good practice, Greefhorst's architecture principles (Greefhorst & Proper, 2011a) were considered throughout the model's design.

Table 20.1 Stakeholder concerns derived from the stakeholder questions, taken from as suggested in the framework were Buckl et al. (2009a)

Stakeholder Question	Concern
Q5, Q6	C1: Gap between the baseline and target EA.
Q5, Q8	C2: Drivers realised by transformations to the EA.
Q7, Q9	C3: Projects applying the transformations to the EA.

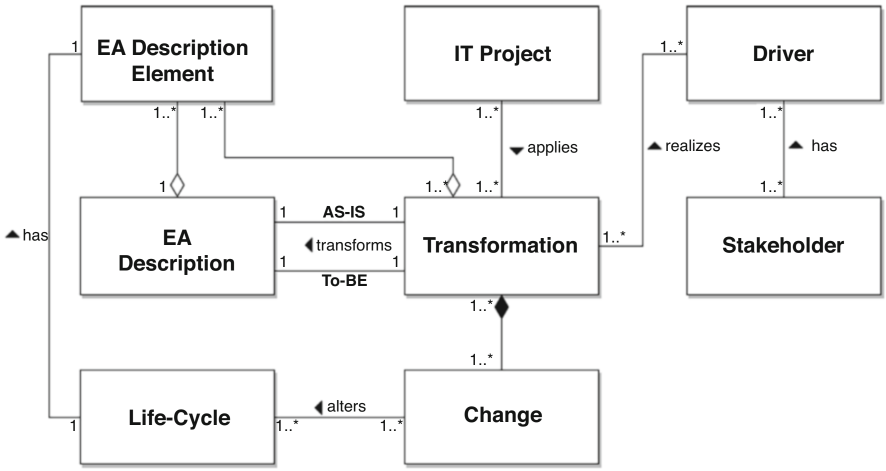


Fig. 20.1 The conceptual model

Figure 20.1 illustrates a UML representation of the proposed conceptual model. A *Transformation* (see Sect. 20.2) has a list of changes that take a baseline *EA Description (as-is)* state and transforms it into a target *EA Description (to-be)* state. A *Change* is an evolution-specific element that alters the *Life-Cycle* (see Sect. 20.2) of one or more *EA Description Elements* (entities, attributes, and relationships). A change has a specific type which relates to its nature, whether destructive, constructive, or refactoring. Transformations are then applied by an *IT Project* (see Sect. 20.2) and realize one or more *Drivers*. A *Stakeholder* (ISO, 2013) can have one or more drivers.

The IT Project, Transformation, and Change concepts express the information regarding the initiatives, and actions are taken to evolve an EA model. As for the life-cycle property of each EA description element, it expresses the time when the actions responsible for an EA change were taken.

The life-cycle role is indicative of the active EA description elements within the EA and of those who are no longer part of it. This type of information allows for the creation of multiple EA snapshots that, together, create the history of the enterprise. Furthermore, the proposed Life-Cycle concept follows the same principles and states as (Sousa et al., 2009)'s approach regarding the *state_of_existence*. Therefore, the states of *conceived*, *gestation*, *alive*, and *dead* were used as the chosen representation of the life-cycle property.

The Driver concept allows one to infer about the motivation of a specific EA change. Finally, the Stakeholder concept pertains to the interested parties that were directly (or indirectly) involved in an EA transformation.

Table 20.2 Number of EA changes – first iteration

EA Change	Number
Create EA Class	1
Create EA Property	216
Update EA Class	–
Update EA Property	2

20.4 Demonstration

This section describes the demonstration phase of the research method, in which the proposed artifact was used to address an instance of the research problem. The artifact's demonstration was performed as part of an EA project within a private organization that, due to non-disclosure reasons, shall be called from now on organization X.

Organization *X* is a European financial organization with a large set of both corporate and private customers. Organization *X* employs proximately 800–900 people and saw in EA a suitable approach for addressing their business needs. Furthermore, due to its large infrastructure, questions regarding the complexity of architectural changes during the development of its architecture were raised, which emphasized the need and consequent opportunity to apply our approach in practice.

One of organization *X* goals was to plan a beta version of its EA and, from there, iterate and evolve their EA model accordingly. So, an interface add-on was implemented within a vendor-specific EAM tool[1] using the conceptual model as a basis for the tool's EA repository schema. The main purpose of such interface was to allow for a practical assessment of the solution's design in supporting the EAM process of evolving organization *X* EA model (and the EA meta-model), thus clarifying understanding issues regarding EA changes.

A first version of the organization's EA meta-model was used as input for the set of applied EA changes. Table 20.2 presents the number of EA changes applied in the first iteration of the project.

Figure 20.2 illustrates the outcome of applying (using the interface add-on) Table 20.2 changes to the organization's EA meta-model. Two lists containing the project's EA transformations' respective requirements are presented as data grids in the left side from top to bottom, respectively. The data point's size in the chart illustrates the number of applied changes by change type. In this scenario, both the Create Property and Remove Property changes comprise almost all of the applied changes. The viewpoint describes the gap between organization *X* EA meta-model *as-is* and *to-be* views.

The change type can be deducted, by analyzing the highlights on each meta-model element. The class highlighted in green (Usage Agreement) was added to organization *X* meta-model. All other classes are highlighted in orange, denoting

[1] http://www.linkconsulting.com/eams/.

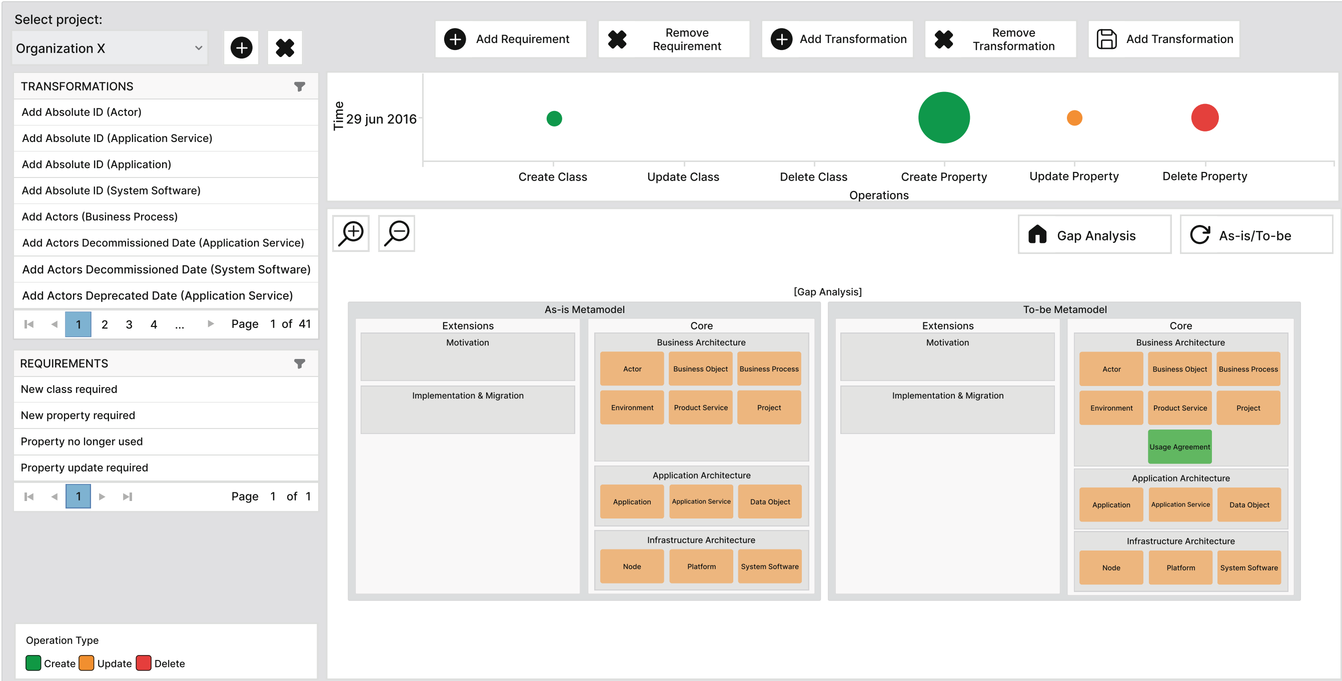


Fig. 20.2 Interface illustrating the *as-is/to-be* gap of organization *X* meta-model

changes in their properties. Correlating the point chart information with changes visualized in the meta-model, stakeholders can conclude that the majority of these operations are Create Property and Remove Property. This inference can be verified in Fig. 20.3, displaying as an example the "Business Process" class properties viewpoint.

The stakeholders (i.e., the organization *X* project team) claimed that this tool-based approach for describing EA change (by applying the proposed conceptual model) allowed them to have a clearer, more holistic view and understanding of the architectural changes (in this case concerning the EA meta-model).

20.5 Evaluation

To validate the proposed DSR artifact, a generic DSR evaluation model was used (see Sect. 20.5.1). Furthermore, in order to assess the conceptual model from a qualitative perspective, a data model evaluation framework for assessing data model's quality was also applied (see Sect. 20.5.2).

20.5.1 Generic Evaluation Model for DSR Artifacts

Prat et al. (2014)'s generic evaluation model for DSR artifacts was applied as a means of assessing the proposed DSR artifact. Table 20.3 presents the generic evaluation methods used throughout the evaluation phase.

The demonstration of the artifact in use (see Sect. 20.4) was the first evaluation method M1 applied in order to assess the artifact's efficacy in addressing an instance of the research problem. Also, in order to validate its utility in a real environment, the model was assessed by means of the implemented interface usage, with user tests (M3) and a utility questionnaire (M5).

To obtain statistically significant numbers, user tests should be done with at least 20 test users (Nielsen, 2012). To that end, a sample of 20 people (6 practitioners and 14 students with knowledge of EA) was used to assess the artifact's usability. A total of 16 tasks were performed, ranging from the creation to the deletion and refactoring of EA description elements, using different types of EA changes, as well as exploring the interface using the supported information visualization features. The average time to perform all tasks was 15 minutes and 5 seconds, the lowest time being 11 minutes and 37 seconds and the highest 17 minutes and 43 seconds. Similar task's time reduced exponentially after the completion of the initial tasks. No user failed to perform any task.

A questionnaire was then given to the users asking questions regarding the artifact's utility relative to its absence. The questionnaire used a scale of 1 to 5, with 1 meaning that the user strongly disagrees and 5 meaning he/she strongly agrees with the statement. The users understood the artifact goal (45% agree and 50%

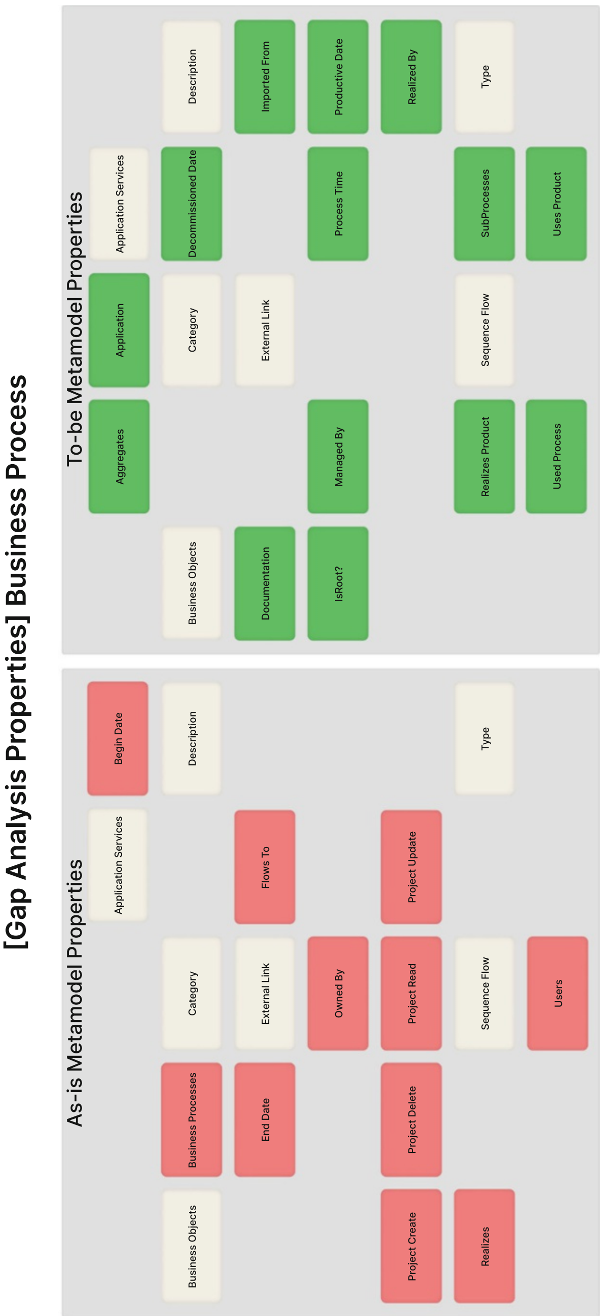


Fig. 20.3 Organization *X* meta-model *as-is/to-be* gap (Business Process properties view)

Table 20.3 Evaluation methods used in validating the conceptual model (based on the framework provided in (Prat et al., 2014))

Evaluation method	Assessed Criterion	Form of evaluation	Secondary Participants	Evaluation level	Evaluation relativeness
Demonstration of the artefact in use with a real example. (M1)	- Goal - Efficacy	- Analysis - Logical reasoning		- Instantiation - Real example	- Absolute
MMeasurement of the performance of students and practitioners in tasks using the artefact. (M3)	- Environment - Usability	- Quantitative	- Students - Practitioners	- Instantiation	- Absolute
Qualitative feedback from students and practitioners on the utility of the artefact. (M5)	- Environment - Utility	- Qualitative	- Students - Practitioners	- Instantiation	- Relative to absence of artefact

strongly agreed), found the model to be intuitive and easy to understand (65% agreed and 30% strongly agreed), acknowledged that the interface features were relevant and properly integrated in the interface (65% agreed and 30% strongly agreed), and disagreed that support was required from someone specialized to use the interface (25% strongly disagreed and 45% disagreed).

20.5.2 Moody and Shanks Framework for Assessing the Model's Quality

The quality of data models is an important aspect to ensure since the modeling of data poses a great impact on the overall system's quality (Simsion & Witt, 2004). Moody (2003), Moody and Shanks (2003) have developed a data model quality management framework that evaluates and improves the quality of data models based on nine quality factors (Moody & Shanks, 2003):

- *Completeness*—refers to whether the data model contains all user requirements
- *Simplicity*—means that the data model contains the minimum possible entities and relationships
- *Flexibility*—the data model's ease to cope with business and/or regulatory change
- *Integration*—the data model's consistency with the rest of the organization's data
- *Understandability*—the ease with which the concepts and structures in the data model can be understood
- *Implementability*—the ease with which the data model can be implemented within the time, budget, and technology constraints of the project

Each quality factor was evaluated using a set of quality metrics (Moody, 2003, 1998). The model's *completeness* was assessed by counting the different types of completeness mismatches according to both the model and user requirements (specializations of the inferred concerns—see Sect. 20.3.1) from organization *X* (see Sect. 20.4). The proposed model proved to be expressive enough to address each requirement.

As to *simplicity*, the authors assessed the model's number of entities ($E = 9$) and the system's complexity ($E+R = 9+10 = 19$). The low number of entities and the system's complexity value prove the artifact contains the minimum required entities and relationships.

Regarding *flexibility*, the elements were reviewed according to future changes. Due to its simplicity and few relationship constraints, the model can be considered flexible enough since few re-structuring would be needed. Nonetheless, a cost impact analysis of such changes would further strengthen the assessment of this quality factor.

For evaluating the *integration* quality factor, an assessment relating to data conflicts with organization *X* data model and existing systems was made. The assessment results were proven to be satisfactory since no apparent conflicts were detected.

To assess *understandability*, a sample of 20 users performed a set of user tasks using the implemented interface (and the conceptual model as data schema) as well as a utility questionnaire relative to the artifact's absence. During the execution of the tasks, the users made no errors, thus understood the model properly since no ambiguity or confusion was raised.

Finally, with respect to *implementability*, no major implementation issues were identified. However, implementability issues will always be related to the complexity of the system which the model is associated with. Therefore, providing an empirical assessment with respect to this quality factor would assume an intensive EAM tool analysis concerning this matter.

20.6 Conclusion

In this chapter, a conceptual model was proposed to address a set of stakeholder concerns regarding the underlying aspects of evolving EA models. The model was implemented within an EAM tool, with the purpose of assessing its utility in addressing the research problem.

The tool was then used as part of an EA project inside a European financial organization. Results showed that the solution artifact came in line with its objectives (see Sect. 20.3.1). Furthermore, to validate the proposed artifact, a DSR evaluation model was used. Finally, to assess the artifact's quality, a data quality framework for evaluating data model's quality was applied.

20.6.1 Limitations

According to Moody (2003), Moody and Shanks (2003) (see Sect. 20.5.2), both flexibility and implementability factors lacked a more circumstantial assessment. First, flexibility was evaluated in a qualitatively way, based on the generic and simple nature of the conceptual model. This type of evaluation can be questionable, whether or not the quality factor is actually achieved. A cost estimate of implementing real or hypothetical changes could reinforce the assessment, in a solid manner.

To fully evaluate implementability, a detailed analysis of EAM tools, which support the evolution of EA meta-models and EA models (using the proposed approach) would be needed. Such an analysis should also take the cost and risks associated with the implementation of such tools into account. To conclude, the interface still lacks the means to apply the required changes to the EA model when the EA meta-model evolves.

20.6.2 Future Work

Efforts are being conducted toward a complete version of the EAM tool interface add-on, namely, impact indicators of EA changes concerning the EA model during the migration phase. The migration logic of the EA model (after applying the changes to the EA meta-model) is also in current development.

Acknowledgments This work was partially supported by the project 11419 (IT-Atlas) of the Portuguese PO CI operational program. The authors would like to thank André Sampaio and Ricardo Leal from Link Consulting SA that contributed for the demonstration and validation of this research, by providing the necessary software, facilities, and support.

Chapter 21
Next-Generation Enterprise Modeling

Bas van Gils and Hend*erik* A. Proper

Abstract In the Western world, digital has become the new normal, both in our daily lives and at our work. Additionally, Western countries have seen a transition from a goods-oriented economy to a services-oriented economy. Whereas, in the recent past, it was already the case that change was *the only constant*, these intertwined, and mutually amplifying, trends even further increase the pace of change. As a result, enterprises are confronted with a need to transform (continuously) accordingly.

During any enterprise transformation, *coordination* among the key stakeholders and the projects that drive the transformations is essential. A shared understanding, agreement, and commitment are needed on (1) what the overall mission/vision of the enterprise is, (2) the current affairs of the enterprise and any ongoing changes, (3) the current affairs of the context of the enterprise, and (4) what (given the latter) the ideal future affairs of the enterprise are.

Models, and ultimately enterprise (architecture) modeling languages and framework, are generally considered as an effective way to support such (informed) coordination. In the past, different frameworks and languages have been developed to this end, including the ArchiMate language. The latter has evolved to become a widely accepted industry standard.

The objective of this chapter is threefold: (1) we intend to illustrate some of the key challenges which the digital transformation, and the two intertwined trends that drive it, puts on enterprise (architecture) modeling languages, (2) assess to what extent ArchiMate meets these challenges, and (3) draft the outline of a next-generation enterprise (architecture) modeling language (framework) that may be more suited to meet the challenges of these trends.

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_21

21.1 Introduction

In the Western world, digital has become the new normal, both in our daily lives and at our work. Computers continue to shrink in size, while their computing powers increase. Furthermore, all these computers have become increasingly interconnected (e.g., the Internet of Things Weber & Weber, 2010). Even the smallest light bulbs have become connected.

It seems as if every aspect of our lives is being impacted on by this trend. Letters are all but replaced by email, and books are digitized, while we track our health through digital/wearable technology (leading to the so-called quantified self Swan, 2012). With the increasing popularity of dating sites, it seems that even our love life is increasingly becoming digital. The same holds for organizations. Whereas IT originally was a mere supportive tool for administrative purposes, it is safe to say that nowadays IT has become an integral part of an organization's primary processes. If the people working in an organization go on strike, then this is likely to lead the organization to come to a *grinding* halt. However, when IT systems fail, most organizations will come to an *abrupt* halt.

According to a recent publication (Brown, 2017), we should even prepare for new forms of diversity in the workforce, where humans should learn to collaborate better with non-humans (e.g., agents, robots, etc.). Human actors, and digital actors, will increasingly work close together.

From a management perspective, this means that considering only the alignment between *business* and *IT* (Henderson & Venkatraman, 1993) is no longer sufficient. The difference between business and IT is increasingly fading; they have been "fused" into one. This has given rise to a wide range of management approaches that are considered to be more "holistic" and consider all aspects of the enterprise; this includes, e.g., information systems architecture (Scheer, 1992; Zachman, 1987) and enterprise architecture (EA) approaches (Pereira & Sousa, 2005; Op 't Land et al., 2008a; Dietz, 2008; van Gils & van Dijk, 2014). Companies, such as Amazon, Airbnb, Uber, Netflix, Spotify, Bitcoin, etc., illustrate how IT and business have indeed become fused. The CEO of a major bank, such as the ING Bank, can even be quoted as stating "We want to be a tech company with a banking license" (Hamers, 2017).

In parallel, Western countries have seen a transition from a goods-oriented economy to a services-oriented economy. Marketing sciences (Vargo & Lusch, 2008; Grönroos & Ravald, 2011; Lusch & Nambisan, 2015; Vargo & Lusch, 2016) suggest that the notion of economic exchange, core to the economy, has shifted from following a goods-dominant logic to a service-dominant logic. While the former focuses on tangible resources to produce goods and embeds value in the transactions of goods, the latter concentrates on intangible resources and the creation of value in relation with customers. It should be noted that a goods-dominant logic is not only adhered to when selling goods. For example, when buying a train ticket, one (might think to) buy a service to get from A to B. At the same time, numerous travelers have experienced that a train ticket is not a guarantee to get from *A* to *B* at all, let alone on time to make it to an important face-to-face meeting.

Service dominance puts the continuous *value co-creation* between providers and consumers at the core. For instance, in the airline industry, jet turbine manufacturers used to follow a classical goods-dominant logic by selling turbines to airlines. However, since airlines are not interested in *owning* turbines, but rather in the realization of *airtime*, manufacturers nowadays sell airtime to airlines instead of jet turbines. As a result, *value co-creation* is shaping up as a key design concern for modern-day enterprises.

Both of these trends are highly intertwined while also amplifying each other. The digital transformation enables new ways of doing business that also enables more value co-creation, resulting in the development of a plethora of new *digital services*. Conversely, the desire of enterprises to co-create value results in a need for more integrated IT solutions, to, e.g., better understand the precise needs of customers and better integrate them in the design/delivery process (Grönroos & Ravald, 2011).

Whereas, in the recent past, it was already the case that change was *the only constant*, the combination of these trends even further increases the pace of change. Ever since the Industrial Revolution, change has often been driven by the introduction of new technology. It seems that the organization that is best at levering technology wins in the marketplace—meaning that keeping up (or even ahead) of developments has become a crucial capability for modern organizations. The plethora of changes that the digital transformation has brought about, and the many more that we are not even aware of yet or have not even been thought of yet, provides organizations with deep and fundamental challenges. How to excel as an organization, while everything is changing constantly? There are hardly any securities left; traditional business models are continuously challenged by digitally inspired and empowered startups.

We consider the trends of *business-IT fusion* and the *shift to value co-creation*, as being the key challenges to enterprises (be they companies, governmental agencies, or organizations) which aim to thrive (or at least survive) in the digital transformation of society.

There are no simple answers to these challenges: a truly wicked problem (Camillus, 2008). We observe how approaches for *digital transformation* increasingly gain popularity, in order to manage the complexity that arises around digitization and the increased speed of change (Gouillart & Kelly, 1995; Rouse & Baba, 2006; Berman, 2012; Westerman et al., 2014). In line with the definitions provided in Chap. 1, we consider transformation to be "the coordinated effort to change the architecture of an enterprise" and digital transformation to be "a transformation of the enterprise with a major impact on its digital resources."

Key in the latter definition is the phrase *coordinated effort*: the actors that make up the enterprise coordinate their efforts not only to fulfil the goals of the enterprise but also coordinate their efforts in ongoing, deliberate change initiatives. During an enterprise transformation, *coordination* (Proper et al., 2018c) among the key stakeholders and the projects that drive the transformations is indeed essential. A shared understanding, agreement, and commitment are needed on (1) what the overall mission/vision of the enterprise is, (2) the current affairs of the enterprise and any ongoing changes, (3) the current affairs of the context of the enterprise, and (4) what (given the latter) the ideal future affairs of the enterprise are. Borrowing the terminology from architecture frameworks such as TOGAF (The Open Group, 2011), this refers to the development of a shared vision, a baseline architecture, and a target architecture, respectively.

Given the speed of change, it remains to be seen whether a true "target architecture" can be developed in light of the fact that it is hard to make predictions, especially about the future. As such, it may be wiser to depart from using the term "target architecture" and refer to it in more open terms, such as "directional architecture," to clarify that it expresses a desired direction of development, rather than a specific target. This would also require a departure from the traditional style of defining a target architecture in terms of a rather "instructive" style in terms of typical "boxes and lines" diagrams toward a more "directional"/"regulative" approach using, e.g., (normative) architecture principles (Greefhorst & Proper, 2011a).

Models, and ultimately enterprise (architecture) modeling languages and framework, are generally considered as an effective way to support such (informed) coordination. Many languages and frameworks have indeed been suggested as a way to create and capture a shared understanding of the desired future affairs. Examples include DEMO (van Reijswoud et al., 1999; Dietz & Hoogervorst, 2007), BPMN (Freund & Rücker, 2012), UML (Object Management Group, 2010), ArchiMate (Band et al., 2016; Lankhorst et al., 2017), 4EM (Sandkuhl et al., 2014), and MERODE (Snoeck, 2014). The latter approaches are applicable in the context of capturing an enterprise's current affairs in terms of its baseline architecture. However, as argued above, for an (open!) future-oriented "directional architecture," these "boxes and lines"-based approaches may have to be complemented with a "directional"/"regulative" approach using, e.g., (normative) architecture principles (Greefhorst & Proper, 2011a).

It appears that ArchiMate is rapidly becoming the industry standard for enterprise architecture modeling[1] and has, as such, a key role to play in the coordination of (Proper et al., 2018c) enterprise transformations. Based on our experience from research and practical work in the field, however, we hypothesize that there are serious challenges with the existing ArchiMate language in light of the needs to support digital transformation efforts (van Gils & Proper, 2018; Proper et al., 2020; Proper, 2020; Proper & van Gils, 2019). As we will see below, these challenges are not only limited to the "instructive" vs. "directional" issue.

The objective of this chapter is therefore threefold: (1) we intend to illustrate some of the challenges that the digital transformation puts on enterprise architecture modeling languages, (2) assess to what extent ArchiMate meets these challenges,

[1] The support for this claim lies in the steady growth of the number of certified professionals http://archimate-cert.opengroup.org/certified-individuals as well as the popularity of the ArchiMate topic on Google trends https://trends.google.com/trends/explore?date=all&q=archimate.

and (3) draft the outline of a next-generation architecture modeling language (infrastructure) that may be more suited to meet the challenges of the digital transformation. In line with this, the remainder of this chapter is structured as follows. We start, in Sect. 21.2, by more closely investigating the challenges that the digital transformation may have on enterprise architecture and the modeling languages used. Based on this, we will then, in Sect. 21.3, present a critical reflection of the suitability of the ArchiMate language in light of these findings. This is followed, in Sect. 21.4, by an outline of a possible "digital transformation-ready" next-generation architecture modeling language (infrastructure). We end this chapter, in Sect. 21.5, with conclusions and directions for future research.

Throughout this chapter, we will use small examples to illustrate key points. Most of these examples derive from real-world projects that have been conducted over the last few years.

21.2 Challenges for Enterprise Modeling

The aim of this section is to identify some of the key challenges on enterprise (architecture) modeling in the context of digital transformation and the increasing focus on value co-creation. The resulting challenges will be used, in Sect. 21.3, as a base to critically reflect on the extent to which ArchiMate already meets the challenges brought forward by these fundamental trends, as well as reflect on possible modifications to ArchiMate to make it better meet the challenges (see Sect. 21.4).

We have grouped the challenges in three classes. First, we discuss challenges pertaining to the expressiveness of the modeling language used in light of the digital transformation and value co-creation. The digital transformation and value co-creation trends push for further specialization and domain specificity of modeling languages. Therefore, the second class of challenges zooms in on the need to be able to manage the resulting spectrum of modeling concepts. The final class of challenges concerns the earlier made observation that the digital transformation fuels the speed of change in organizations and their enterprises.

21.2.1 Expressiveness of the Modeling Language

21.2.1.1 Objects Can Be Operand and Operant

Objects (including humans) in the world around us can play different roles. Sometimes, they play an active role, in the sense that they become the operant actor, which (co-)enacts a certain activity. They may even become the actor bearing the social responsibility for the enactment of such an activity (Dietz et al., 2013). Objects may also play a passive role, in which case they can actually be the operand/subject of an activity. Key is that it is natural for the same objects to play different roles in the course of time or even in parallel.

In traditional views on enterprise architecture, it was more or less assumed that objects were either passive (operand) or active (operant) for their entire life. One would certainly not mix these types of roles. This simplification might have indeed worked in former times. However, in the context of the digital transformation, this simplification becomes increasingly difficult to uphold.

Objects, in particular the digital ones, are created and manipulated by other (human and/or digital) objects. In the digital world, objects can be both operand and operant, even at the same time. An enterprise architecture modeling language used in digital transformations should therefore be able support this plurality of the roles played by objects:

Challenge 1 *Objects should be allowed to play operand and operant roles.*

21.2.1.2 Information Versus Reality

Digital transformations also result in an increased reliance on the quality of (digitally represented) information in terms of the correctness at which it represents the world around us. As a result, it becomes increasingly important to remain aware of, and thus explicitly capture, the distinction between elements in the real world and the information that *stands model* for those real-world elements.

Enterprise (architecture) modeling languages should, therefore, also clearly reflect such a distinction: for example, in terms of a clear distinction between *business objects* as they exist in the real world and *business information objects* that represent information *about* the former objects. An example of an architecture framework which already supports such a distinction is the Integrated Architecture Framework (Wout et al., 2010). This results in the following challenge for enterprise modeling languages:

Challenge 2 *Clear separation between objects that represent"things" in the real world and objects representing information about the real world.*

21.2.1.3 Natural Duality of Human and Digital Actors

As also discussed in the introduction, according to a recent publication (Brown, 2017), a consequence of the digital transformation is that we should prepare for new forms of diversity in the workforce, where humans should learn to collaborate closely with digital actors (e.g., agents, robots, etc.). In line with (Dietz et al., 2013), we take the position that the social responsibility of activities should remain with human and/or organizational entities. We are, for example, not (yet) expecting that robots can be taken to court, to account for their actions, and possibly be punished when breaking societal rules. Underlying this is the well-known question: *When an autonomous car causes an accident, who is responsible?*

Modern-day enterprise modeling languages should, therefore, be able to deal more naturally with the duality of human and digital actors while making explicitly clear where the ultimate social responsibility and accountability of the actions by these actors lie:

Challenge 3 *Ability to deal naturally with the duality of human and digital actors.*

21.2.1.4 Identification Management

A key aspect in traditional (conceptual) data modeling is the notion of *unique identification*: in other words, the ability to specify how objects in the real world can be distinguished from one another.

The ability to uniquely identify the (passive and/or active, operand/operant) objects around is indeed quite convenient, even though not all applications will need it. Depending on the application context, we may need unique identification. In some cases, it might even be illegal, e.g., due to privacy considerations, to have such a unique identification. For example: Is there a need to uniquely identify all water molecules in the stream of water coming from a well with mineral water? Probably not. Is there a need to identify each individual bottle of water filled with water from this well? Probably yes, as well as the date when it was filled. Is there a need to identify each individual traveler on a public transport system? Would probably be useful for optimization purposes, as well as monitoring possible "terrorist" activities. Is it allowed/desirable from a privacy perspective? Probably not.

At the same time, even when a unique identification mechanism is available and is allowed to be used, there may be limits regarding its completeness and uniqueness. In a business network involving multiple partners, one may have to use multiple, partially overlapping, identification mechanisms. Even more, one may not have control over the creation of objects, which may (accidentally or maliciously) end up having the same properties as used in the identification.

For enterprise modeling languages, this makes it important to be able to specify if objects can, should, and/or are allowed to be uniquely identified and, if so, to what extent this unique identification can indeed be assumed to cover the entire (possible) population of such objects:

Challenge 4 *Ability to specify if objects can, should, and/or are allowed to be uniquely identified.*

21.2.1.5 Optional Modalities on Relationships

Most enterprise modeling languages do not allow for detailed modalities (mandatory, optional, one-to-one, one-to-many, etc.) on relationships. In general, this has been a deliberate choice by the language designers. In practice, however, this decision is challenged. It has been debated extensively—for example, in the

LinkedIn group for ArchiMate as well as during training and coaching sessions—how useful it would be to be able to specify modalities, in particular, in the context of privacy and security, two concerns that become even more important in digital transformation(s).

One may argue that such modality rules are "too detailed" to be included at an architecture level. At the same time, there are many cases where there is a need to specify (even at an architecture level) the rules governing relationships in more detail. A typical example would be the four-eyes principle, where two roles must be fulfilled when performing a certain task. It is expected that such modeling constructs will be needed frequently, in the context of privacy and security. We suggest that, although one should not categorically require architecture models to use modalities on relationships, this should be addable when needed:

Challenge 5 *Ability to specify modalities on relationships.*

21.2.1.6 Orientation Toward Value Co-creation

Western countries have witnessed a transition from a goods-oriented economy to a services-oriented economy. Digital transformation triggers the development of a plethora of new *digital services*, even further boosting the dominance of services in Western economies.

Several studies (Vargo & Lusch, 2004; Lusch & Vargo, 2006; Vargo & Lusch, 2008; Maglio et al., 2009; Grönroos & Ravald, 2011) observe a fundamental paradigm shift from, what they call, a goods-dominant logic to a service-dominant logic. While the former focuses on the production of goods, the latter concentrates on the delivery of services using resources and/or goods in doing so. These studies motivate this shift by observing that it is ultimately the customer who attributes value to a good or a service. Goods and services, "at rest," only have a potential value to a customer. The actual value is experienced when the resources/goods are actually *used* by the customer to some purpose.

In parallel to the shift from a goods-dominant logic to a service-dominant logic, one can observe a growing awareness that a conventional enterprise-centric (inside-out) view of value creation is now being challenged by a newer customer-centric (outside-in) view of value creation (Prahalad & Ramaswamy, 2000; Priem, 2007; Lepak et al., 2007; Priem et al., 2013). This leads to the perspective that value results by way of a process of *co*-creation between producer and consumer, involving the integration of their resources (Vargo & Lusch, 2008).

To achieve strategic advantage, service-providing enterprises must be able to co-create value for their customers, at a higher level of quality than the competition does (Bettencourt et al., 2014). This also entails a need for enterprises to broaden the scope of their enterprise architecture, more specifically, from a focus on the design of efficient, reliable, and flexible (IT-supported) business processes to a broadened one, with a more prominent place for the design of value co-creation with partners and customers.

The *digital transformation* not only brings about a new wave of digital services, but it also acts as an enabler that allows providers of goods and service to better optimize the co-creation of value with their customers: for example, by being able to (1) more swiftly create, and manage, *on-the-fly* business processes and (2) tune/customize their products and services to the needs of specific users (in their context of use) and (3) based on detailed (digital) profiles of the needs, preferences, and habits of the users.

As a consequence, enterprise architecture modeling languages need to include constructs to explicitly express the (potential) value(s) of products and services to customers, in particular in terms of *value in use* and *resource integration*, and how this results in *value co-creation* between providers and consumers of services.

Challenge 6 *Ability to capture (potential) value(s) of products and services and how this results in value co-creation between providers and consumers of services by way of resource integration.*

21.2.1.7 Implementation and Design Choice Awareness

Given the speed of technological developments that drive the digital transformation, it is increasingly important for organizations to be aware of the essential design choices shaping the essence of their business[2] activities, as well as choices with regard to their implementation by means of different platforms and technologies. The latter includes choices such as the use of (business process) outsourcing, software platforms, hardware platforms, cloud computing, division of labor between human and computer-based actors (also see Challenge 3), etc.

For enterprise (architecture) modeling languages, this means that one should be able to express the design of the enterprise (including its use of information technology) at different levels of specificity with regard to implementation decisions, as well as enable the capturing of the associated design decisions and their motivation (Plataniotis et al., 2015b, 2014b).

Challenge 7 *Express the design of the enterprise at different levels of specificity with regard to implementation decisions.*

Challenge 8 *Capture design decisions and their motivation.*

[2] When using the word "business," we do so in the sense of "a particular field of endeavour" (Meriam–Webster, 2003); i.e., we are specifically not only referring to "commercial businesses."

21.2.2 Managing the Spectrum of Modeling Concepts

21.2.2.1 Managing the Set of Modeling Concepts

An enterprise (architecture) modeling language typically features a rich set of modeling concepts. As a natural consequence of the use of such a language, and as a corollary to the law of entropy, there is a tendency to continue adding concepts to modeling languages (Bjeković et al., 2014), in particular when such a language has the status of being a standard.

Digital transformation, due to its deep impact and multifacetedness, is likely to further fuel the entropic forces, likely leading to a further increase in the number of modeling concepts. Some of the challenges listed above actually also point toward a desire to extend existing modeling languages. Next to that, specific concerns, such as security, privacy, value co-creation, etc., are likely to play a stronger role in digital transformation and thus also trigger a need for dedicated modeling concepts (Bjeković et al., 2014).

At the same time, an ever-increasing set of modeling concepts will lead to a modeling language that will be hard to learn (Moody, 2009; Krogstie et al., 1995) while also endangering the overall consistency of the set of modeling concepts.

This leads to the following challenge on enterprise (architecture) modeling language (frameworks):

Challenge 9 *A way to manage the set of modeling concepts, balancing the needs of domain, and purpose, specificity, the need for standardization, and comprehensibility of the modeling language.*

21.2.2.2 Consistent Abstraction Layer Structures

Enterprise architecture modeling languages typically involve different abstraction layers. Examples include the business, application, and technology layer as used in ArchiMate (Lankhorst et al., 2017); the essential and implementation layer as suggested by Enterprise Ontology (Dietz & Hoogervorst, 2007); the function and construction perspective as suggested by the same; the business, information systems, and technology layer from TOGAF (The Open Group, 2011); the business, information, information systems, and technology infrastructure columns from IAF (Wout et al., 2010); as well as the conceptual, logical, and physical layers of the same.

In line with the earlier discussion on implementation and design choice awareness (Challenges 7 and 8), using such abstraction layers for digital transformations is indeed wise. At the same time, we observe in practice (both in using such frameworks and teaching about them) that confusion about the precise scoping of the used abstractions exists. In this regard, one can even distinguish changes in the interpretation of the business, application, and technology layer from ArchiMate as intended originally (Lankhorst et al., 2017), where the technology layer was purely intended as the (IT) technological *infrastructure*, to the current interpretation, where it has evolved to include the entire (IT) technological implementation (Band et al., 2016). As we will discuss in Sect. 21.3, this also leads to further challenges regarding relationships between layers in the case of ArchiMate.

In general, one could say that abstraction layers (even in multiple dimensions, as suggested by the IAF (Wout et al., 2010) and Zachman (Zachman, 1987) frameworks) result from the *design philosophy* underlying the specific framework. In this chapter, we do not aim to take a specific position with regard the question of which *design philosophy* would be best. However, we do argue, in particular when considering the challenges of digital transformations, that it is important that the layer structure must use clear and consistent abstractions.

For enterprise (architecture) modeling language (frameworks), this leads to the following challenge:

Challenge 10 *Provide a structure that allows to consistently use abstractions across relevant aspects of the enterprise.*

21.2.2.3 Grounding Modeling

Enterprise (architecture) models play an increasingly important role. When developing/evolving an enterprise, models are used to capture the current affairs, as well as articulate different possible future affairs. Even more, nowadays, it is quite common that models are even part of the "running system," in the sense that they are an artifact that drives/guides day-to-day activities. This includes workflow models, business rule sets, etc.

This makes it important that enterprise models also capture their meaning[3] in a way that is understandable to the model's audience. We therefore posit that a conceptual model should be grounded in the terminology as it is actually *used* (naturally) by the people involved in/with the modeled domain. We see this as a key enabler for the transferability of models across time and among people, in particular in situations where the model needs to act as a *boundary object* (Abraham et al., 2013a).

Most existing enterprise modeling languages (e.g., process models, goal models, actor models, value models, architectural models, etc.) only offer a "boxes and lines"-based representation that only provide a limited linkage to the (natural) language as used by the model's audience. In general, the only link in this regard are the names used to label the "boxes." Relationships are replaced by generic graphical representations in terms of arrows and lines capturing relations such as "assigned to," "part of," "realizes," "aggregates," and "triggers."

[3] In principle, we would prefer to use the word "semantics" here. However, since the word "semantics," in our computer science-oriented community, tends to be equated to only mean "formal semantics," we will use the word *meaning*.

While these abstract, and more compact, notations of purpose/domain-specific modeling languages enable a more compact representation of models, they offer no means to provide a "drill down" to an underlying grounding in terms of, e.g., well-verbalized fact types that capture, and honor, the original natural (language) nuances (Hoppenbrouwers et al., 2019). They leave no room for situation-specific nuance or more explicit capturing of the meaning of the models in a way that is understandable to the model's audience (beyond engineers). The challenge therefore is:

Challenge 11 *How to ground enterprise models in terms of natural language like verbalizations, without losing the advantages of having compact notations (as well).*

21.2.3 Enabling a Regulative Perspective

As mentioned in the introduction of this chapter, an often used idiom is that *change is the only constant*, while the digital transformation results in a further increase in the *rate* of this "constant change." In light of such rapid changes, the notion of the traditional baseline architecture has its difficulties.

As a result of these rapid changes, the enterprise is in a constant motion, which means that the baseline is not simply a "state," but rather a "vector" (Proper & Lankhorst, 2014). Hence, it is better to speak about capturing "current affairs," which includes past, and present, change trends, of the enterprise and its environment. As a consequence, the traditional concept of a "target architecture" needs to be reconsidered as well. Of course, in terms of TOGAF and ArchiMate, this concept has been extended toward a multi-stage version in terms of "plateaus" toward the future. Nevertheless, it remains to be seen how specific such plateaus/target architectures can be developed in light of the fact that it is hard to make predications, especially about the future.

As such, it may be wiser to depart from using the term "target" and refer to it in more open terms, such as "directional," to clarify that it expresses a desired direction of development, rather than a specific target. This would also require a departure from the traditional style of defining a target architecture (or plateaus) in terms of a rather "instructive" style in terms of typical "boxes and lines" diagrams toward a more "directional/"regulative" approach using, e.g., (normative) architecture principles (Greefhorst & Proper, 2011a).

When considering the motivation extension of the latest ArchiMate (Band et al., 2016) version, and the increasing awareness of the role of architecture principles, it seems sensible to identify three levels of enterprise architecture modeling:

Desires-oriented dealing with goals of stakeholders and their ensuing requirements. Model artifacts from this perspective should be owned (content-wise) by the stakeholders and should be formatted in terms of what the stakeholders want to do and achieve.

Constraints-oriented dealing with (normative) architecture principles, regulations, constraints, etc., limiting the design space. Model artifacts from this perspective should be owned by both stakeholders and architects/designers and form a translation from the stakeholders' desires to consequences/constraints toward the actual design, without making concrete/specific design decisions yet.

Construction-oriented dealing with specific "instructions" on how (parts of) the enterprise should actually be constructed (and implemented). This involves the typical boxes and lines diagrams. Ownership lies with the architects/designers, and design decisions should of course comply to what has been stated from the constraints- and desires-oriented perspectives.

The resulting challenge for modeling languages is:

Challenge 12 *How to balance a desires-, a constraints-, and a construction-oriented perspectives on an enterprise, in light of constant change.*

21.3 ArchiMate's Readiness for the New Enterprise Modeling Challenges

In this section, we start by providing a high-level introduction to the current version of the ArchiMate language, including its development history.[4] We then continue with a discussion to what extent the current version of ArchiMate meets the challenges of digital transformations, as identified in Sect. 21.2. This provides the context for the discussion in the next section, where we propose improvements of the language.

21.3.1 The Development of the ArchiMate Language

In line with (Hoppenbrouwers, 2000, 2003; Hoppenbrouwers et al., 2005b; Frank, 2013, 2011), we argue that a modeling language, and designed languages in general, should reflect the actual (intended) use of the language.

A purposely developed language, such as an enterprise architecture modeling language, is fundamentally an artifact in the design science research (Hevner et al., 2004; van Aken, 2004; Peffers et al., 2007) sense. A design science process typically follows an (iterative) design process in terms of requirements elicitation, design, and development, followed by some form of testing/evaluation, while also allowing for

[4] At the time of writing, ArchiMate 3.0.1 is available online.

possible iterations. The process for the development of domain-specific modeling languages as suggested by, e.g., (Frank, 2013) follows a similar pattern.

In line with this, it would be appropriate to use the design science research process as suggested in, e.g., (Peffers et al., 2007) in the development of a language such as ArchiMate. Even though at the time of the development of the initial versions of the ArchiMate language (early 2000s), design science research had not yet fully emerged (Hevner et al., 2004), the development of ArchiMate did follow a basic design process. ArchiMate's development started with the establishment of a set of initial requirements (Bosma et al., 2002; Jonkers et al., 2003). Using further input from enterprise architects from industrial partners involved in the research project, the architecture of the ArchiMate language was then developed (Lankhorst et al., 2010), and the final design of (the initial version of) the ArchiMate language was created.

Since then, the ArchiMate language has gone through several iterations (Lankhorst et al., 2017; Band et al., 2016). Based on real-world use of the language, several refinements and improvements were made. In addition, the tighter integration into TOGAF (The Open Group, 2011) also resulted in additional extensions. As a result, the language has also grown considerably in terms of the included concepts.

21.3.2 Overview of the ArchiMate Language

ArchiMate is a dedicated language for representing (enterprise) architecture models that was originally developed by a consortium of organizations in the Netherlands after which it was adopted by The Open Group as an (open) standard (Lankhorst et al., 2017; Band et al., 2016). Its adoption has grown rapidly, both in terms of the *users* of the language and the *vendors* that deliver software solutions based on this language. A full discussion of ArchiMate (Lankhorst et al., 2017) is beyond the scope of this chapter. However, for purposes of our analysis, we will present a rough outline of the structure of the language.

The current version of the language supports five layers (strategy, business, application, technology, and physical) and four aspects (active structure, passive structure, behavior, and motivation). The core of the framework—and focus of this discussion—consists of the business/application/technology layer, and all aspects save the motivation aspect. The rationale for leaving out motivation and implementation aspects lies in the fact that these are crucial for the architecture process, but are not used to describe the actual architecture of the enterprise. The layers in the core have the same generic meta-model which is shown in Fig. 21.1. The later meta-model is also contained in the specification of the ArchiMate standard (Band et al., 2016), albeit with some additional details.

Services are used as a decoupling mechanism. They are used to specify what an active structure element exposes to its environment and hide the complexity of how the services are realized. Services can be used both within a layer (e.g., a department

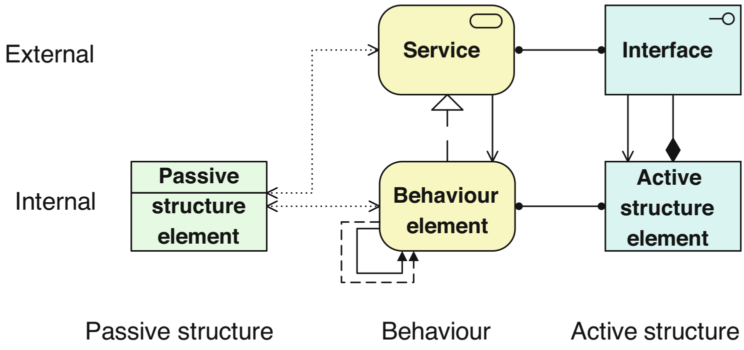


Fig. 21.1 Generic ArchiMate meta-model, adapted from Lankhorst et al. (2017); ©2017 Springer-Verlag Berlin Heidelberg; reprinted with permission

offering services to another or an application offering services to another) and across layers (e.g., which processes are served by an application service).

A second abstraction mechanism in the language is the *specialization relation*, which is to be interpreted as "is a kind off." Some languages, such as ORM and UML, distinguish between (a) specialization, (b) generalization, and (c) type/instance relations (Halpin, 2001; ter Hofstede & van der Weide, 1993; Hay, 2011; Fowler, 2004). In ArchiMate, these are all captured by the same specialization relation. Using this relation, it is possible to relate generic architecture constructs (e.g., a process pattern) into more specific manifestations (e.g., distinguishing between the regular manifestation of the process or the manifestation that is followed during times of crisis).

An abstraction mechanism that was introduced in version 3 of the ArchiMate language is the use of *grouping*. Previously, the grouping was a visual construct only, which was intended to show on a view which concepts "belong together" for some reason. Since ArchiMate version 3, the intended meaning is more rigorous: the grouping is said to aggregate the concepts that are in it and thus functions as a semantic whole. Groupings may be related to other concepts (including other groupings). This makes it particularly well suited to use the grouping as a form of *building blocks* along the lines of the TOGAF standard (e.g., The Open Group, 2011, Chapter 37).

The last mechanism that is relevant to our discussion here is the notion of *cross-layer dependencies* (Band et al., 2016, Chapter 12). The general idea is that elements from one layer can be connected to elements of other layers using the *serves* (previously: used-by) relation or the *realization* relation. Through this mechanism, we can specify, for example, that a business process is realized by an application process. Along the same lines, it allows us to specify that a group of elements (i.e., a building block) is realized by another group of elements (another building block).

Putting this all together leads to the example of Fig. 21.2 that illustrates the concepts explained in this section. Starting at the top left, we see an architecture

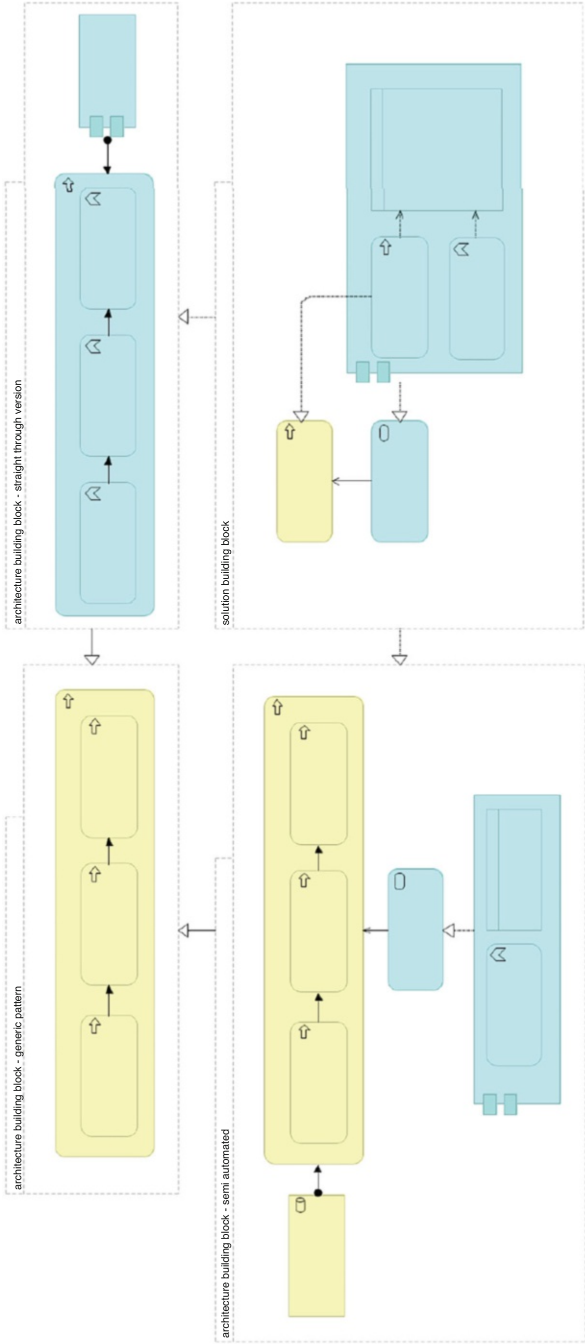


Fig. 21.2 Abstraction mechanisms in ArchiMate, version 3

building block that specifies a process pattern of three steps. Using the specialization relation, we see two more specific building blocks at the bottom left (semi-automated) and top right (straight through). Using the realization relation, we see that these two together are implemented through a solution building block that ties objects together.

21.3.3 Analysis in Light of the Identified Challenges

In this section, we briefly touch upon each of the identified challenges before presenting a short reflection on the current "digital transformation readiness" of ArchiMate.

Challenge 1 *Objects should be allowed to play operand and operant roles.*

The current ArchiMate language does not deal with this well due to the strict distinction between *active* and *passive* structure elements. This challenge lies at the heart of the ArchiMate language.

Challenge 2 *Clear separation between objects that represent "things" in the real world and objects representing information about the real world.*

This challenge refers to the passive structure elements in ArchiMate. Currently, there is no clear distinction between the two types of objects, other than the observation that data objects/artifacts presumably are about the bits and bytes that represent information. The ArchiMate specification does suggest that the *business object* concept can be specialized but in the default language this has not been done. What the specification does not mention is that additional relations may also be required in order to present that informational business object A *is about* real-world business object B.

Challenge 3 *Ability to deal naturally with the duality of human and digital actors.*

Here, the ArchiMate language, through its layering, does provide a fair attempt at tackling this challenge since there are different concepts for, e.g., actor, information system, and node. Some interesting challenges remain, however. First of all, only (business) actors can be assigned a role in behavior; other structure elements cannot. A second mismatch lies in the fact that *collaborations* in ArchiMate can only be composed of structure elements from the same layer. This prevents us from specifying that a human actor and computer actor collaborate to achieve a certain task.

Challenge 4 *Ability to specify if objects can, should, and/or are allowed to be uniquely identified.*

In ArchiMate, concepts are essentially "types," representing the "instances" in the real world. The ArchiMate concepts have a name to tell one apart from the other. There is no mechanism to specify how the "instances" should be told apart.

Challenge 5 *Ability to specify modalities on relationships.*

For this challenge, we can be short: ArchiMate has no support for this. Objects are either related, or they are not.

Challenge 6 *Ability to capture (potential) value(s) of products and services and how this results in value co-creation between providers and consumers of services by way of resource integration.*

Evaluation of the current ArchiMate language against this challenge is somewhat tricky. This is because the language does have the *value* concept, and it seems possible to model value *co*-creation by using the collaboration/interaction concepts. However, as discussed in, e.g., (Razo-Zapata et al., 2017, 2018), representing value co-creation (Lusch & Vargo, 2006; Lusch & Nambisan, 2015), scenarios require more dedicated modeling constructs.

Challenge 7 *Express the design of the enterprise at different levels of specificity with regard to implementation decisions.*

Challenge 8 *Provide a structure that allows to consistently use abstractions across relevant aspects of the enterprise.*

These challenges are closely related. The latest version of ArchiMate does indeed provide some rudimentary support to tackle these challenges through the *grouping* mechanism. It is now possible to express the fact that one group of concepts (together) realizes another group of concepts. This allows the modeler to work from a big picture level to a more detailed level, as well as from a functional level to a more construction-oriented level.

Challenge 9 *Capture design decisions and their motivation.*

There is limited support in ArchiMate to address this challenge. We would argue that using (a specialization of) the requirement concept could potentially work, but is far from elegant. As an example of a more elaborate approach to the motivation of design decisions, consider the work reported in (Plataniotis et al., 2014a, 2015a).

Challenge 10 *A way to manage the set of modeling concepts, balancing the needs of domain, and purpose, specificity, the need for standardization, and comprehensibility of the modeling language.*

Potentially, this challenge is addressed partially by means of the *extension mechanisms* to tailor the language to local needs while keeping the core of the language compact. This can be done by specializing existing concepts or by adding properties to existing concepts. While it is good that the language indeed supports this, being able to reuse extensions across toolsets of different vendors is not straightforward. Even more, the extension mechanism is not really positioned as a key feature in the standard either.

In addition, recent extensions of the language have been captured as so-called extensions, such as the *motivation* extension and the *implementation and migration* extension.

Challenge 11 *How to ground enterprise models in terms of natural language like verbalizations, without losing the advantages of having compact notations (as well).*

ArchiMate has no support for this, neither in the language nor the modeling process. Even more, there is no predefined modeling procedure such as ORM's CSDP (Halpin & Morgan, 2008), leaving (in particular novice modelers) to guess how to master ArchiMate's elaborate set of modeling concepts (Proper et al., 2018b).

Challenge 12 *How to balance a desires-, a constraints-, and a construction-oriented perspectives on an enterprise, in light of constant change.*

Support for this challenge is limited. ArchiMate does have the ability to model different plateaus—which relates to different points in time—and it allows the modeler to link concepts to motivational elements of key stakeholders. Full support for balancing the different perspectives, however, is lacking.

21.3.4 Reflection

Building a modeling language that supports modelers to consistently solve challenges, and solve them well, is a difficult task indeed. After listing modeling challenges and evaluating the current version of ArchiMate against these challenges, we conclude that—even though ArchiMate has been around for a while and has a strong conceptual framework—its support for the challenges of digital transformation is fair at best. At first glance, it appears that several of the constructs in the language need reconsideration in order to meet the listed challenges. How this could play out is the topic of the next section.

21.4 Next-Generation Architecture Modeling Language

A full (re)design of the ArchiMate language is certainly beyond the scope of this chapter. Instead, we provide (motivated) recommendations that could overcome the challenges as discussed above.

21.4.1 Modular Language Design

modular As discussed in Sect. 21.3.1, the set of modeling constructs within the ArchiMate language has grown considerably. Furthermore, as discussed in Sect. 21.3.3, the use of ArchiMate's *extension mechanism* indeed provides a good starting point to better manage the resulting set of concepts. The positioning of recent additions to the language as *extensions*, such as the *motivation* extension and the *implementation and migration* extension, indeed underlines this.

In general, we suggest that modeling language standards should focus primarily on providing a generic core of well-defined, and possibly even formalized (ter Hofstede & Proper, 1998), modeling concepts. On top of this core, one could then define refinement mechanisms that can be used to extend/tailor the core to the needs at hand. This may involve both specializations of the core concepts and the introduction of different abstraction layers.

In addition, a library of (meta-model) *modules* can be defined, which could potentially even be (re)used across different language cores. For example, a generic *motivation* module could be shared between ArchiMate, DEMO (Dietz, 2006), and BPMN (OMG, 2011). At the same time, a modular approach would also enable more flexibility in terms of, e.g., the layering of abstractions. For instance, ArchiMate (and TOGAF) have a "hard-wired" layering of the so-called business-to-IT stack involving *business*, *application*, and *technology*. Other frameworks, such as Capgemini's IAF (Wout et al., 2010), have a more refined layering "hard-wired" into their structure, involving *business*, *(business) information (systems)*, *(computerized) information systems*, and *technology*.

When looking at the original architecture of the ArchiMate language as reported in (Lankhorst et al., 2010), there are indeed ample opportunities for further modularization of the ArchiMate language. For example, following (Lankhorst et al., 2010), and as also confirmed by (Band et al., 2016), the core of the language is formed by five key generic "active systems" modeling concepts: *objects*, *service*, *internal behavior*, *interface*, and *internal structure*. All other concepts are explicitly derived from these in terms of specializations (Lankhorst et al., 2010).

We argue that this specialization hierarchy has been left too implicit for far too long and that an explicit re-factoring of the current ArchiMate language based on this hierarchy is long overdue, more specifically, using language construction mechanisms such as:

Meta-model modules: that allows for the expression of specific language functionalities, such as *motivation* and *migration planning*. Each of such modules should include the identification of an *interface* by which it can be connected to other language modules.

For example, a *motivation* module could feature a generic *design element* as a placeholder for the elements of design for which a motivation needs to be provided. The motivation module can then be used to motivate designs in different languages.

Layering mechanisms: involving a set of *meta-model modules* used to connect multiple layers. For instance, ArchiMate's business/application/technology layers are typically connected by means of services-calls and realizations-relations. Making these into more explicit modules allows users to adapt the layering to the needs of their organization. For instance, as mentioned above, IAF (Wout et al., 2010) suggests a more refined layering in terms of business/information/(computerized) information systems/technology.

Concept specialization: in terms of, e.g., the existing extension mechanism.

21.4.2 Grounding Enterprise Modeling

Challenge 5 suggests that enterprise models should (unless they only serve a temporary "throw away" purpose) include a precise definition of the meaning[5] of the concepts used in the model. We see this as a key enabler for the transferability of models across time and among people (Proper et al., 2004; Hoppenbrouwers et al., 2005a), in particular in situations where the model needs to act as a *boundary object* (Abraham et al., 2013a).

In line with this, we posit that, to ensure that a model is understandable to its audience, it should be grounded on an (underlying) fact-based model involving verbalizations using the terminology as it is actually *used* (naturally) by the people involved in/with the modeled domain (Hoppenbrouwers et al., 2019).

As exemplified in (van Bommel et al., 2007a,b; Tulinayo et al., 2013; Proper et al., 2018b), fact-based models can be used to ground enterprise models that are expressed in languages, such as ArchiMate, DEMO (Dietz, 2006), system dynamics (Rouwette & Vennix, 2006), and BPMN (OMG, 2011), and architecture principles (Greefhorst & Proper, 2011a), in terms of underlying fact models. In doing so, the basic idea is to:

1. Consider an enterprise (be it an existing one or an imagined future one), including all its aspects (in particular, the "business-to-IT stack"), as an active system (of systems)
2. describe the structures and behavior of this active system in terms of (observable) facts

The latter is fully aligned to ArchiMate's roots on natural language structures involving agens (*active structure*), patiens (*passive structure*), and verb (*behavior*).

When indeed observing an active system in terms of fact (types), one essentially creates the (structure of a) fact-based logbook of what "happens" in the active system (van Bommel et al., 1996).

Even though we strongly suggest to remain close to the terminology as it is actually *used* (naturally) by the people involved in/with the modeled domain, we do see the potential benefits of providing guidance in structuring/refining this terminology based on, e.g., foundational ontologies (Guizzardi, 2006).

Grounding ArchiMate on fact-based models would also lead to a natural way to deal with Challenge 1, i.e., the challenge that objects should be allowed to play operand and operant roles. Indeed, when observing objects and expressing their engagements in activities in terms of fact types, one can easily observe object to play different roles in different facts (types), mixing between *passive structure/active structure/behavior* roles. For example, a computer may be a passive element in the

[5] In principle, we would prefer to use the word "semantics" here. However, since the word "semantics," in our computer science-oriented community, tends to be equated to only mean "formal semantics," we will use the word *meaning*.

context of it being manufactured, but it may be an active element in a context where it processes key processes at some company.

Based on (Proper et al., 2018b), the suggested solution would be to treat the ArchiMate concepts as *roles* which an object may enact. This allows for a natural way for an object, say the computer in the above example, to enact both the role of an active and a passive element in the same ArchiMate model.

21.4.3 Adding More Semantic Precision

Both Challenge 4 and Challenge 5 require the ability to specify more semantic specificity regarding objects and relations. Such properties, e.g., identification mechanisms and cardinality constraints, have always been part of modeling languages such as ER (Chen, 1976), ORM (Halpin & Morgan, 2008), and UML (OMG, 2007). As such, it would be logical to "import" such mechanisms from these existing languages into ArchiMate.

Needless to say, it is not required for architects to specify such constraints in all situations. The key is to provide the ability to do so when required.

As an example, consider the situation as shown in Fig. 21.3. This is not a full example, yet it illustrates the main line of thinking. The setting is risk management and (quality) control in a production process. Suppose that we have a manual production process that is tightly controlled with production guidelines, metrics, controls, etc. In such a situation, there is a need to be able to represent the fact that:

- Supervisor roles must be executed by a human actor who may, in some other setting, also perform other tasks. The supervisor role may be played in one or more processes. However, a production process must have one and only one supervisor.
- Production roles must be fulfilled by a human actor also. The production role may be played in many processes. Even more, a production process may have more than one production role.
- We want to avoid that the supervisor role is played by the same human actor as the production role.

In Fig. 21.3, we have chosen to use the UML-style notation of adding cardinality at the association ends.

21.4.4 Abstraction Layers

Challenges 2, 4, and 10 are essentially all concerned with different ways to "separate concerns." As argued in Sect. 21.2.2, it is important to ensure a clear and consistent

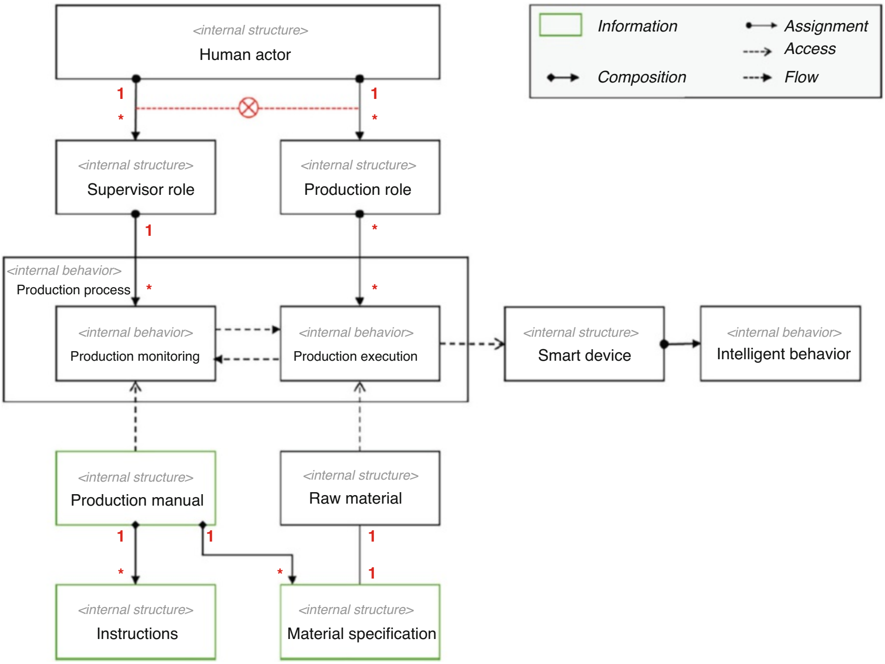


Fig. 21.3 Example of the use of with modality and cardinality constraints

structure of abstraction layers. In this chapter, we do not aim to take a specific position as regards the question of which *design philosophy* would be best.

When looking "across" different frameworks (ArchiMate Lankhorst et al., 2017, Enterprise Ontology Dietz & Hoogervorst, 2007, TOGAF The Open Group, 2011, IAF Wout et al., 2010, and Zachman, 1987), we posit that the following key constructs for the creation of abstractions (in different dimensions) are used:

Function-construction—This involves a distinction between:

1. Function refers to how a system is intended to function in light of what users, clients, and other stakeholders might deem useful.
2. Construction refers to how a system actually functions/is constructed to realize the provided functions.

Note that there may be good reasons for a constructed system to deviate from how it was specified from a functional perspective. For example, it may be more cost-effective to purchase a system that provides additional functionality—which was not originally specified—than to construct a system in line with what was specified.

Informational functioning—This dimension concerns different levels (of *aspect* systems (of systems)) that describe different levels of *functioning* of an enterprise in terms of informational support, leading to a *business*, an *informational*, and a *documental* level.

Infrastructure usage—This concerns the fact that one system (of systems) can *use* the functions of another system (of systems), where the actual construction of the latter is of no interest to the (designers) of the former, except to the extent of defining service-level agreements.

Implementation abstraction—This concerns the gradual/stepwise introduction of details of the socio-technical implementation. For example, in IAF (Wout et al., 2010), this corresponds to the distinction between a conceptual, logical, and physical level, while in TOGAF (The Open Group, 2011), this corresponds to the level of architectural and logical building blocks.

Making a clear implementation abstraction also provides a natural way to deal with Challenge 3 pertaining to deal with the duality of human and digital actors. At the highest level of implementation abstraction, one would need to describe the workings of the enterprise independent of the question if it will be implemented by means of human actors or computerized actors. The immediate next level of implementation abstraction might then to make choices with regard to human/computerized actors explicitly, even allowing for mixed scenarios, while, e.g., also identifying which actors are ultimately responsible and accountable.

Each of the above-discussed abstraction mechanisms has a potential added value, also in the context of digital transformation. It is important to note that these abstraction mechanisms should not be thought of as a set of orthogonal dimensions. On the contrary, the *function-construction* mechanism and *information functioning* or *function-construction* and *infrastructural usage* can easily be mixed. We also do not want to suggest to "prescribe" a specific set of dimensions. We do, however, argue that an enterprise modeling language (framework) should ensure a consistent use of the above mechanisms within one dimension.

As discussed in Sect. 21.3, ArchiMate seems to have been mixing some of these dimensions in an inconsistent way.

21.4.5 Value Co-creation

The increasing focus on value co-creation, resulting from the shift from a goods-dominant logic to a service-dominant logic (Vargo & Lusch, 2008; Grönroos & Ravald, 2011; Lusch & Nambisan, 2015; Vargo & Lusch, 2016), results in Challenge 6, i.e., how to capture (potential) value(s) of products and services and how this results in value co-creation between providers and consumers of services by way of resource integration.

ArchiMate already provides *value* concept, and it seems possible to model value *co*-creation by using the collaboration/interaction concepts. However, as mentioned before, value, or even a value stream, is not the same as value *co-creation*. How to best express this is still largely an open question. Some initial work/suggestions have been presented in Razo-Zapata et al., 2016; Feltus & Proper, 2017a, 2017b; Razo-Zapata et al., 2017).

The very nature of value co-creation also requires a shift from (only) architecting the "internals" in an enterprise to co-architecting the collaboration (including needed inter-organizational IT platforms) between multiple partners in the co-creation network (Chew, 2016).

In further elaborating the set of needed concepts for value co-creation, our recommendation (Proper et al., 2018a) is to (1) use the provider/customer roles as identified in (Grönroos & Voima, 2013), specialized to more specific co-creation activities taking place within the *provider sphere*, the *joint sphere*, or the *customer sphere*, as a reference model, while (2) using the foundational premises as articulated in (Vargo & Lusch, 2016) as design/architecture principles (Greefhorst & Proper, 2011a) that will guide the design of service systems for value co-creation, and (3) apply this in the context of real-world cases, to gain insight into the actually needed modeling concepts.

21.4.6 Capturing Design Motivations

The current version of ArchiMate does provide a motivation extension. However, as discussed in the previous section, it does not meet Challenge 8 in a satisfactory way. Separate from the fact that, as suggested above, it would be good if such an extension could be shared between, e.g., ArchiMate, 4EM (Sandkuhl et al., 2014), and BPMN (OMG, 2011), the actual level at which design decisions remain rather crude.

The work as reported in, e.g., Plataniotis et al. (2014a, 2015a) provides suggestions on how to remedy this. This includes the ability to, e.g., capture trade-offs between design alternatives and the actual decision-making process and the criteria used to make decisions (including the identification of compensatory and/or non-compensatory (Rothrock & Yin, 2008) criteria).

21.4.7 Managing Constant Change

As discussed in Sect. 21.2, the digital transformation requires enterprises to change constantly. This makes it less realistic to capture an enterprise's *current affairs* and/or *desired affairs* in terms of traditional notions such as "baseline" architecture and "target" architecture or even *plateaus/transition* architectures. Even though we observe some ingredients toward solutions for this challenge, we would argue that more research is certainly needed.

In an ideal world, the description of the *current affairs* would be maintained continuously, preferably in an automated way (Proper, 2014). Approaches such as process mining (van der Aalst, 2011) and enterprise cartography (Tribolet et al., 2014b) indeed provide good starting points.

Architectures capturing the *desired affairs* also tend to be specified using a rather "instructive" of typical "boxes and lines" diagrams. This does not really invite architects to reflect on what the more *endurable* elements and assumptions and what the less stable elements and assumptions are. This has also triggered the development of the concept of multi-speed enterprise (IT) architectures (Abraham et al., 2012). It also resulted in a stronger positioning of, e.g., (normative) architecture principles (Greefhorst & Proper, 2011a) as a way to complement the "instructive" style (the "boxes and lines" diagrams) by a more "directional"/"regulative" perspective.

21.4.8 Consequences for the Meta-Modal

Modeling is a key aspect of how we, as humans, attempt to get to grips with reality. Whether models are—as much as possible—an accurate representation of what happens/should happen in the real world or whether they are "merely a hypothesis" of what we believe to be true about the real world does not change the fact that the modeling *language* must be precise enough to express what we want. In the previous section, we explored limitations of the current predominant architecture modeling language: ArchiMate. In this chapter, we explored how some of these limitations can be alleviated. This is deliberately positioned as an *exploration*.

The main contribution of our approach is twofold. First of all, we believe that the meta-model for representing all aspects of the digital enterprise should be (a) greatly simplified and (b) made more flexible. The biggest change is to remove the active/passive structure dichotomy, which provides a more natural way of expressing the state of affairs in the real world. Another is only apply layering on the structure side, which avoids duplication of modeling decision. Even more, when combined with a decision to have less predefined (structure) concepts, this allows users of the language to adapt the language more to their individual needs while still retaining the integrity rules of the overall framework. Last but not least, adding the notion of constraints and modalities will give modelers the option to add more precision to their models where needed.

It is a well-known fact that the proof of the pudding is in the eating: it would make sense to use our approach in practice to see if, indeed, it lives up to its promise.

21.5 Conclusion and Further Research

In this chapter, we presented key challenges which the digital transformation puts on enterprise (architecture) modeling languages. These challenges are based on practical experiences and insights from the field of enterprise architecture.

We then assessed the extent to which the current version of ArchiMate meets these challenges. The conclusion was that ArchiMate does not yet fully cover all of the identified challenges. This can be explained by the fact that ArchiMate was developed at a time when the digital transformation was not yet that dominant.

We then provided suggestions on how to possibly improve ArchiMate to better meet the challenges of digital transformations. In further research, we intend to further elaborate these suggestions, in particular with the aim of finding strategies that work in real-world practice.

Chapter 22
Conclusion

Bas van Gils, Kazem Haki, and Henderik A. Proper

This conclusion will briefly reflect on the contributions of these chapters in this part.

In the first chapter of this part, we explored the application of cartography (the science of making maps) in light of enterprises. Enterprise cartography is defined to be the process of abstracting, collecting, structuring, and representing architecture artifacts (the observable elements in the enterprise) and their relations from observations of enterprise reality. The latter point ties in with the notion of *grounded modeling* as discussed in the last chapter of this part. One of the key notions in this chapter is the evolution of maps/model from AS-WAS, to AS-IS, to TO-BE.

The notion of an *emerging AS-IS* model is introduced. With increasing speeds of change, it is more and more important to know where you are on the map exactly, and therefore, this emerging AS-IS is so important to the success of digital transformation initiatives. The approach is defined through clear definitions of key concepts (e.g., architecture statement, architecture map, transformation initiative) and five core principles (e.g., "all enterprise artifacts have a 5-stage life cycle"). In the evaluation of the EC approach, based on cases, the authors conclude that maps help with achieving "organizational self-awareness" which, in turn, is key for digital transformation.

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_22

In the chapter of this part, the thread of AS-WAS, AS-IS, and TO-BE is picked up and explored further. This chapter is based on the observation that models can only be of use if they accurately represent the real world. Given that the real world evolves from AS-WAS to TO-BE, it makes sense that models should co-evolve.

The evolution of models is a topic that has been researched extensively. For example, in (Proper, 1994), a theory was presented to capture the evolution of conceptual models in light of evolving application domains. Chapter 20 goes a step further and explores a conceptual model of model evaluation at the architecture level. The study is based on a design science approach and results in a conceptual model with eight concepts (e.g., *enterprise architecture description element*, *enterprise architecture description*, *lifecycle*, *change*, etc.) and ten relations to connect them. The model was implemented in a software tool and tested against the Moody and Shanks criteria (completeness, simplicity, flexibility, integration, understandability, and implementability). Despite some limitations, the conceptual model and approach seems promising in light of its stated objectives to understand model evolution at the architecture level.

Chapter 21 evaluated the ArchiMate modeling language which emerges as the de facto standard for enterprise architecture modeling in the field. Based on a short survey of available literature and the practical experience of one of the authors in the field, several challenges have been identified for architecture modeling in light of digital transformation initiatives. The analysis focused on eight challenges with regard to the expressiveness of the modeling language itself and four challenges around managing the spectrum of modeling concepts. In our view, the development of the next version of ArchiMate should address these challenges. The next evolution step hinges on (a) modular language design, (b) grounded enterprise modeling, (c) adding more semantic precision, (d) fixing/redesigning abstraction mechanisms, (e) more explicit support for value co-creation, (f) capturing design decisions, and (h) managing constant change—from AS-WAS, AS-IS, to TO-BE. We have illustrated how this could play out with examples.

Evaluating and synthesizing the findings of these three chapters, we conclude that architecture modeling continues to play an important role in digital transformation initiatives, but in order to stay relevant and effective, evolution is key: not only for the models themselves but also for the meta-model and frameworks behind them.

Part V
Epilogue

Chapter 23
Final Conclusions and Outlook

Hend*erik* A. Proper, Bas van Gils, and Kazem Haki

In this final chapter, we reflect on the contributions of this book as a whole while also identifying challenges for future research.

23.1 Summary

As discussed in Chap. 1, the creation of this book was triggered by three, mutually amplifying, trends that drive enterprises to change: the transition to the digital age, the emergence of service ecosystems, and the growing role of data as a key underlying resource. As a result of these intertwined, and mutually amplifying, trends, enterprises are more than ever confronted with a need to transform while becoming increasingly *service-focused*, *digitally powered*, and *data-fueled*.

To meet these challenges, there is no one-size-fits-all approach. Therefore, the aim of this book was to explore different relevant aspects in more detail while at the same time also providing concrete suggestions for enterprises to meet the resulting challenges. In line with this, the contributions brought together in this book covered four key perspectives:

H. A. Proper (✉)
Institute of Information Systems Engineering, TU Wien, Vienna, Austria
e-mail: henderik.proper@tuwien.ac.at

B. van Gils
Strategy Alliance, Amersfoort, the Netherlands
e-mail: bas.vangils@strategy-alliance.com

K. Haki
Campus Battelle, Bâtiment B, Haute Ecole de Gestion de Genéve, Geneva, Switzerland
e-mail: kazem.haki@hesge.ch

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5_23

1. Part I—involving *experience reports* on how enterprises deal with these trends in practice
2. Part II—concerned with the need for a new *design logic*
3. Part III—addressing the need for *architectural coordination* of the needed transformations
4. Part IV—discussing consequences for *enterprise modeling*

23.2 Fundamental Challenges

The contributions in this book also point toward more fundamental challenges underlying digital transformation and enterprise transformation in general. Below, we discuss three of the main challenges we see.

Each of these challenges requires further research: both to understand the underlying mechanisms and to formulate strategies and approaches to meet these challenges.

23.2.1 A Multi-speed and People-Driven Approach to Change

As different contributions in this book already underlined, "the only constant is change." At the same time, these changes come at different speeds, thus implying a need to also follow a multi-speed approach in dealing with change: for instance, involving at least a fast cycle (dealing with the immediate threats and opportunities at hand) and a slow cycle (steadily directing the enterprise toward a desired future). However, as needed, additional levels may be identified.

From an enterprise architectural point of view, these different speeds of accommodating change result in a gradual shift from a descriptive style of architecting for those elements/aspects of the enterprise that can be assumed to be stable for a longer time toward a more prescriptive style of architecting for those elements/aspects that are likely to involve a high level of dynamics.

As also observed in different contributions in this part, digital transformation is not only about *technology* but also about the *people* driving/realizing changes. The challenge then becomes how to, on the one hand, engage/mobilize the right people to drive change while, on the other hand, remaining in control of many (human-driven) bottom-up changes as they occur across an enterprise.

23.2.2 Building Capabilities for Change

Several contributions in this book highlighted the need to better organize an enterprise's capabilities for change. Capability is a contraction of capacity and ability. Understanding and influencing capability of an enterprise can be dubbed *capability-based planning*, a notion that stems from the military domain (we need the capability to wage war on two fronts and win).

Research on strategic management (Winter, 2003; Teece, 2007; Pavlou & El Sawy, 2010) argues that, in dealing with turbulent environments, enterprises need three types of capabilities, namely, *operational*, *dynamic*, and *improvisational*. Operational capabilities are organizational routines and processes that are developed over time through learning and provide organizations with the capacity to undertake activities in a reliable manner. Dynamic capabilities are forward-looking capabilities by which organizations extend, modify, or reconfigure existing operational capabilities into new ones in response to disruptive technological shifts and innovations. Improvisational capabilities are second-order dynamic capabilities by which organizations spontaneously reconfigure existing resources into new ones to address urgent and unpredictable environmental situations.

For enterprises to be successful in managing change, they will need to find a balance between the different capabilities while also managing the needed differentiation in speed as discussed in the previous subsection.

23.2.3 Modeling Capability as a Foundational Capability

In dealing with the many levels and speeds of change that confront enterprises, it will become increasingly important for enterprises to be aware of all relevant activities and activities inside, and outside, the organizational boundaries. Even more, the different actors involved in/impacted by these changes need (1) to have insight into the existing structures and operations of an enterprise; (2) to be able to express, assess, and evaluate different design options for their future; and (3) to have instructions on how to make the necessary changes to these structures and operations and (4) how to operate in the future.

These needs were also touched upon in the different contributions in this book. Chapter 11 explicitly introduced the notion of organizational self-awareness to stress the need for the actors involved in an enterprise to be aware of the current operations of an enterprise, as well as its future. As argued in, e.g., Magalhães and Proper (2017), Proper and Bjeković (2019), Proper (2021), (enterprise) models should also be understood from a broader perspective than mere "boxes and lines" diagrams. More specifically, enterprise models potentially capture important enterprise knowledge Lillehagen and Krogstie (2010). This can, e.g., pertain to knowledge in relation to the well-known interrogatives (*why*, *who*, *whose*, *when*, *how*, *with*), be positioned in time (*as-was*, *as-is*, *as-planned*, *to-be*, etc.), be nuanced in terms of modalities (*must*, *ought*, *desired*, etc.), take a prescriptive or a descriptive perspective, etc.

As such, next to, e.g., an enterprise's operational capabilities and dynamic capability, its *modeling capabilities* will become an increasingly important foundational capability of enterprises. The challenge will be to further improve these *modeling capabilities* by means of tools, modeling languages, and associated processes while balancing the return on modeling effort (RoME Op 't Land et al., 2008b; Guizzardi & Proper, 2021).

Bibliography

Åström, K. J. & Murray, R. M. (2008). *Feedback systems: An introduction for scientists and engineers*. Princeton University Press. ISBN: 978-0691135762

van der Aalst, W. M. P. (2011). *Process mining: Discovery, conformance and enhancement of business processes*. Springer. ISBN: 978-3642193446.

van der Aalst, W. M. P. (2014). Data scientist: The engineer of the future. In K. Mertins et al. (Eds.), *Enterprise Interoperability VI, Proceedings of the I-ESA Conferences 7*. Springer.

van der Aalst, W. M. P. et al. (2012). Process mining manifest. In *Business Process Management Workshops*, volume 99 of *Lecture Notes in Business Information Processing* (pp. 169–194).

Aarikka-Stenroos, L., & Jaakkola, E. (2012). Value co-creation in knowledge intensive business services: A dyadic perspective on the joint problem solving process. *Industrial Marketing Management, 41*(1), 15–26. https://doi.org/10.1016/j.indmarman.2011.11.008

Abraham, R., Aier, S., & Winter, R. (2012). Two speeds of EAM—a dynamic capabilities perspective. In [Aier, S., Ekstedt, M., Matthes, F., Proper, H. A., & Sanz, J. (Eds.) (2012). *Trends in Enterprise Architecture Research and Practice-Driven Research on Enterprise Transformation—7th Workshop, TEAR 2012, and 5th Working Conference, PRET 2012, Held at The Open Group Conference 2012, Barcelona, Spain, October 23-24, 2012. Proceedings*, volume 131 of *Lecture Notes in Business Information Processing*. Springer. ISBN: 978-3-642-34162-5. http://doi:10.1007/978-3-642-34163-2] (pp. 111–128). ISBN: 978-3-642-34162-5. https://doi.org/10.1007/978-3-642-34163-2

Abraham, R., Niemietz, H., de Kinderen, S., & Aier, S. (2013a). Can boundary objects mitigate communication defects in enterprise transformation? Findings from expert interviews. In R. Jung & M. Reichert (Eds.), *Proceedings of the 5th International Workshop on Enterprise Modelling and Information Systems Architectures, EMISA 2013, St. Gallen, Switzerland, September 5–6, 2013*, volume 222 of *Lecture Notes in Informatics* (pp. 27–40). Gesellschaft für Informatik. ISBN: 978-3-88579-616-9. https://dl.gi.de/handle/20.500.12116/17238

Abraham, R., Tribolet, J. M., & Winter, R. (2013b). Transformation of multi-level systems—theoretical grounding and consequences for enterprise architecture management. In H. A. Proper, D. Aveiro, & K. Gaaloul (Eds.), *Advances in Enterprise Engineering VII—Third Enterprise Engineering Working Conference, EEWC 2013, Luxembourg, May 13-14, 2013. Proceedings*, volume 146 of *Lecture Notes in Business Information Processing* (pp. 73–87). Springer. ISBN: 978-3-642-38116-4 https://doi.org/10.1007/978-3-642-38117-1_6

Abraham, R., Aier, S., & Winter, R. (2014). Fail early, fail often: Towards coherent feedback loops in design science research evaluation. In M. D. Mayers & D. W. Straub (Eds.), *Proceedings of the International Conference on Information Systems—Building a Better World Through*

H. A. Proper et al. (eds.), *Digital Enterprises*, The Enterprise Engineering Series,
https://doi.org/10.1007/978-3-031-30214-5

Information Systems, ICIS 2014, Auckland, New Zealand, December 14–17, 2014. Association for Information Systems. http://aisel.aisnet.org/icis2014/proceedings/ISDesign/3

Ackoff, R., & Emery, F. E. (1972). *On purposeful systems*. Tavistock Publications.

AFME (2015). *Post Trade Explained: The role of post-trade services in the financial sector*. Report, The Association for Financial Markets in Europe. http://www.afme.eu

Agrawal, R., Gunopulos, D., & Leymann, F. (1998). *Mining process models from workflow logs*. Springer.

Aier, S. (2013). Understanding the role of organizational culture for design and success of enterprise architecture management. In R. Alt & B. Franczyk (Eds.), *Proceedings of the 11th International Conference on Wirtschaftsinformatik (WI 2013), Leipzig, Germany* (pp. 879–894).

Aier, S., & Winter, R. (2010). Fundamental patterns for enterprise integration services. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET), 1*(1), 33–49. http://doi:10.4018/jssmet.2010010103

Aier, S., Ekstedt, M., Matthes, F., Proper, H. A., & Sanz, J. (Eds.) (2012). *Trends in Enterprise Architecture Research and Practice-Driven Research on Enterprise Transformation—7th Workshop, TEAR 2012, and 5th Working Conference, PRET 2012, Held at The Open Group Conference 2012, Barcelona, Spain, October 23-24, 2012. Proceedings*, volume 131 of *Lecture Notes in Business Information Processing*. Springer. ISBN: 978-3-642-34162-5. http://doi:10.1007/978-3-642-34163-2

Akaka, M. A. & Vargo, S. L. (2014). Technology as an operant resource in service (eco)systems. *Information Systems and E-Business Management, 12*(3), 367–384. http://doi:10.1007/s10257-013-0220-5

Akaka, M. A., Vargo, S. L., & Lusch, R. F. (2012). An exploration of networks in value cocreation: A service-ecosystems view. *Review of Marketing Research, (9)*, 13–50. http://doi:10.1108/S1548-6435(2012)0000009006

van Aken, J. E. (2004). Management research based on the paradigm of the design sciences: The quest for field-tested and grounded technological rules. *Journal of Management Studies, 41*(2), 219–246.

Akerlof, G. A., & Kranton, R. E. (2012). *Identity Economics: How our identities shape our work, wages, and well-being*. Princeton University Press.

Albert, S., & Whetten, D. A. (1985). Organizational identity. In L. L. Cummings & B. M. Staw (Eds.), *Research in organizational behavior* (number 8, pp. 263–295). JAI Press.

Alter, S. (2008). Defining information systems as work systems: Implications for the is field. *European Journal of Information Systems, 17*(5), 448–469.

Alves, H., Fernandes, C., & Raposo, M. (2016). Value co-creation: Concept and contexts of application and study. *Journal of Business Research, 69*(5), 1626–1633. http://doi:10.1016/j.jbusres.2015.10.029

Alvesson, M., & Empson, L. (2008). The construction of organizational identity: Comparative case studies of consulting firms. *Scandinavian Journal of Management, 24*(1), 1–16.

AMA. (2017). Agẽncia Para Moderniçăo Administativa. https://www.ama.gov.pt

Anderson, D., & Ackerman Anderson, L. S. (2001). *Beyond change management: Advanced strategies for today's transformational leaders*. Jossey-Bass/Pfeiffer.

Anthony, S., Viguerie, S., & Waldeck, A. (2016). *Corporate longevity: Turbulence ahead for large organizations*. Executive Briefing Spring. Innosight.

Argote, L., & Kane, A. A. (2009). Superordinate Identity and Knowledge Creation and Transfer in Organizations. In N. Foss & S. Michailova (Eds.), *Knowledge governance: Processes and perspectives.* Oxford University Press.

Arrow, K. (1974). *The limits of organization*. W.W. Norton.

Ashby, W. R. (1956). *An introduction to cybernetics*. Chapman & Hall. ISBN: 0-412-05670-4.

Ashforth, B. E., Harrison, S. H., & Corley, K. G. (2008). Identification in Organizations: An examination of four fundamental questions. *Journal of Management, 34*(3), 325–374.

ASX. (2016). *ASX's replacement of CHESS for equity post-trade services: Business requirements*. Consultation paper, ASX Limited, Sydney.

Aveiro, D., Silva, A. R., & Tribolet, J. (2010). Towards a GOD-theory for organizational engineering: Continuously modeling the continuous (re)generation, operation and deletion of the enterprise. In *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC), Sierre, Switzerland, March 22–26*.

Axelos. (2015). *ITIL foundation handbook*. The Stationery Office. ISBN: 978-0113314690

AZoRobotics. (2018). Using artificial intelligence and autonomous robotics for rapid exploration of deep-sea ecosystems. https://www.azorobotics.com/News.aspx?newsID=10037

Bénaben, F., Touzi, J., Rajsiri, V., Truptil, S., Lorré, J. P., & Pingaud, H. (2008). Mediation information system design in a collaborative SOA context through a MDD approach. In *Proceedings of MDISIS* (pp. 1–17).

Band, I., Ellefsen, T., Estrem, B., Iacob, M.-E., Jonkers, H., Lankhorst, M. M., Nilsen, D., Proper, H. A., Quartel, D. A. C., & Thorn, S. (2016). *ArchiMate 3.0 specification*. The Open Group. ISBN: 978-9401800471

Barile, S., Lusch, R., Reynoso, J., Saviano, M., & Spohrer, J. (2016). Systems, networks, and ecosystems in service research. *Journal of Service Management, 27*(4), 652–674. http://doi:10.1108/JOSM-09-2015-0268

Barley, S. R., & Kunda, G. (1992). Design and devotion: Surges of rational and normative ideologies of control in managerial discourse. *Administrative Science Quarterly, 37*, 363–399.

Barrett, R. (2006). *Building a values-driven organization: A whole system approach*. Butterworth-Heinemann.

Baskerville, R. L., & Myers, M. D. (2009). Fashion waves in information systems research and practice. *MIS Quarterly, 33*(4), 647–662. https://doi.org/10.2307/20650319

Baskerville, R., & Pries-Heje, J. (2010). Explanatory design theory. *Business & Information Systems Engineering, 2*(5), 271–282. https://doi.org/10.1007/s12599-010-0118-4

Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Kranenburg, R. v., Lange, S., & Meissner, S. (Eds.) (2013). *Enabling Things to Talk—Designing IoT solutions with the IoT Architectural Reference Model*. https://doi.org/10.1007/978-3-642-40403-0_8

Bauer, M., Boussard, M., Bui, N., de Loof, J., Magerkurth, C., Meissner, S., Nettsträter, A., Stefa, J., Thoma, M., & Walewski, J. W. (2013a). IoT reference architecture. In [Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Kranenburg, R. v., Lange, S., & Meissner, S. (Eds.) (2013). *Enabling Things to Talk—Designing IoT solutions with the IoT Architectural Reference Model*. https://doi.org/10.1007/978-3-642-40403-0_8].

Bauer, M., Bui, N., de Loof, J., Magerkurth, C., Nettsträter, A., Stefa, J., & Walewski, J. W. (2013b). IoT reference model. In [Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Kranenburg, R. v., Lange, S., & Meissner, S. (Eds.) (2013). *Enabling Things to Talk—Designing IoT solutions with the IoT Architectural Reference Model*. https://doi.org/10.1007/978-3-642-40403-0_8]

Beck, D. E., & Cowan, C. (2005). *Spiral dynamics: Mastering values, leadership, and change*. Wiley-Blackwell.

Becker, H. (2014). *Social impact assessment: Method and experience in Europe, North America and the developing world*. Routledge. ISBN: 9781315072432. https://doi.org/10.4324/9781315072432

Becker, M., & Klingner, S. (2013). A metamodel for component-based service modeling. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET), 4*(2), 60–77. https://doi.org/10.4018/jssmet.2013040105

Beirão, G., Patrício, L., & Fisk, R. (2017). Value cocreation in service ecosystems: Investigating health care at the micro, meso, and macro levels. *Journal of Service Management, 28*, 227–249. https://doi.org/10.1108/JOSM-11-2015-0357

Bennaceur, A., Tun, T. T., Bandara, A. K., Yu, Y., & Nuseibeh, B. (2018). Feature-driven mediator synthesis: Supporting collaborative security in the Internet of Things. *ACM Transactions on Cyber-Physical Systems, 2*(3), Paper nr. 21. https://doi.org/10.1145/3134843

Berghout, E., Nijland, M., & Powell, P. (2011). Management of lifecycle costs and benefits: Lessons from information systems practice. *Computers in Industry, 62*(7), 755–764.

Berio, G., & Vernadat, F. (2001). Enterprise modelling with CIMOSA: Functional and organizational aspects. *Production Planning & Control, 12*(2), 128–136. https://doi.org/10.1080/09537280150501239

Berman, S. J. (2012). Digital transformation: Opportunities to create new business models. *Strategy & Leadership, 40*(2), 16–24.

Bertalanffy, L. V. (1979). *General system theory: Foundations, development, applications*. Braziller. ISBN: 0-807-60452-6

Besson, P., & Rowe, F. (2012). Strategizing information systems-enabled organizational transformation: A transdisciplinary review and new directions. *Journal of Strategic Information Systems, 21*(2), 103–124. https://doi.org/10.1016/j.jsis.2012.05.001

Bettencourt, A. A., Fusch, R. F., & Vargo, S. L. (2014). A service lens on value creation: Marketing's role in achieving strategic advantage. *Californian Management Review, 57*(1), 44–66.

Biggart, N. W., & Guillén, M. F. (1999). Developing difference: Social organization and the rise of the auto industries in South Korea, Taiwan, and Argentina. *American Sociological Review, 64*, 722–747.

Binns, R., Lyngs, U., Van Kleek, M., Zhao, J., Libert, T., & Shadbolt, N. (2018). Third party tracking in the mobile ecosystem. arXiv:1804.03603. https://arxiv.org/pdf/1804.03603

Birch, D., Brown, R., & Parulava, S. (2016). Towards ambient accountability in financial services: Shared ledgers, translucent transactions and the technological legacy of the great financial crisis. *Journal of Payments Strategy & Systems, 10*(2), 118–131.

Bitner, M., Brown, S. W., & Meuter, M. L. (2000). Technology infusion in service encounters. *Journal of the Academy of Marketing Science, 28*(1), 138–149.

Bjeković, M., Proper, H. A., & Sottet, J.-S. (2014). Enterprise modelling languages—just enough standardisation? In B. Shishkov (Ed.), *Business Modeling and Software Design—Third International Symposium, BMSD 2013, Noordwijkerhout, the Netherlands, July 8-10, 2013, Revised Selected Papers*, volume 173 of Lecture Notes in Business Information Processing (pp. 1–23). Springer. ISBN: 978-3-319-06670-7. https://doi.org/10.1007/978-3-319-06671-4_1

Black, J. (2008). Forms and paradoxes of principles-based regulation. *Capital Markets Law Journal, 3*(4), 425–457.

Black, J., Hopper, M., & Band, C. (2007). Making a success of principle-based regulation. *Law and Financial Markets Review, 1*(3), 191–206.

Blaschke, M. (2018). Socio-technical complexity in digital platforms: The revelatory case of Helix Nebula—the science cloud. In N. Urbach & M. Röglinger (Eds.), *Digitalization cases*. Springer. https://doi.org/10.1007/978-3-319-95273-4_9

Blaschke, M., Haki, M., Aier, S., & Winter, R. (2018). Value co-creation ontology—a service-dominant logic perspective. In P. Drews, B. Funk, P. Niemeyer, & L. Xie (Eds.), *Proceedings of the Multikonferenz Wirtschaftsinformatik 2018 (MKWI 2018), Lüneburg, Germany*.

Blaschke, M., Haki, K., Aier, S., & Winter, R. (2019a). Taxonomy of digital platforms: A platform architecture perspective. In *Proceedings of the 14th International Conference on Wirtschaftsinformatik (WI2019), Siegen, Deutschland*. Association for Information Systems.

Blaschke, M., Riss, U., Haki, K., & Aier, S. (2019b). Design principles for digital value co-creation networks—a service-dominant logic perspective. *Electronic Markets, 29*, 443—472. https://doi.org/10.1007/s12525-019-00356-9

Boland, R. J., Collopy, F., Lyytinen, K., & Yoo, Y. (2008). Managing as designing: Lessons for organization leaders from the design practice of Frank O. Gehry. *Design Issues, 24*(1), 10–25.

van Bommel, P., Frederiks, P. J. M., & van der Weide, T. P. (1996). Object–oriented modeling based on logbooks. *The Computer Journal, 39*(9), 793–799.

van Bommel, P., Buitenhuis, P. G., Hoppenbrouwers, S. J. B. A., & Proper, H. A. (2007a). Architecture principles—a regulative perspective on enterprise architecture. In M. Reichert, S.

Strecker, & K. Turowski (Eds.), *Enterprise Modelling and Information Systems Architectures—Concepts and Applications, Proceedings of the 2nd International Workshop on Enterprise Modelling and Information Systems Architectures (EMISA'07), St. Goar, Germany, October 8–9, 2007*, number 119 in Lecture Notes in Informatics (pp. 47–60). Gesellschaft für Informatik. ISBN: 978-3-88579-213-0. https://tinyurl.com/y4ddva3r

van Bommel, P., Hoppenbrouwers, S. J. B. A., Proper, H. A., & van der Weide, T. P. (2007b). On the use of object-role modeling for modeling active domains. In K. Siau (Ed.), *Research issues in system analysis and design, databases and software development* (pp. 123–145). IGI Publishing. ISBN: 978-1-599-04927-4

Bonchek, M., & France, M. (2015). The best digital strategists don't think in terms of either/or. *Harvard Business Review, 6*, 1–5.

Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The unified modeling language user guide*. Addison–Wesley Publishing Company.

Borgers, M. A. C. (2021). *Are architecture principles effective? A study in Dutch governmental organizations.* Ph.D. Thesis, Maastricht University. ISBN: 978-94-6423-297-4

Borgers, M. A. C., & Harmsen, A. F. (2016). Case report of identifying and measuring it architecture principles in the Dutch tax agency. In E. Kornyshova, G. Poels, & C. Huemer (Eds.), *18th IEEE Conference on Business Informatics, CBI 2016, 29th August—1st September 2016, Paris, France, Volume 2—Short and Workshop Papers*: IEEE Computer Society. https://doi.org/10.1109/CBI.2016.57

Bosma, H., ter Doest, H. W. L., & Vos, M. (2002). *Requirements*. Technical Report ArchiMate Deliverabe D4.1, TI/RS/2002/112, Telematica Instituut.

Boucharas, V., van Steenbergen, M., Jansen, S., & Brinkkemper, S. (2010). The contribution of enterprise architecture to the achievement of organizational goals: A review of the evidence. In H. A. Proper, M. M. Lankhorst, M. Schönherr, J. Barjis, & S. Overbeek (Eds.), *Proceedings of the 5th International Workshop on Trends in Enterprise Architecture Research*, number 70 in Lecture Notes in Business Information Processing—Advances in Enterprise Engineering I (pp. 1–15). Springer.

Breidbach, C. F., & Maglio, P. P. (2016). Technology-enabled value co-creation: An empirical analysis of actors, resources, and practices. *Industrial Marketing Management, 56*, 73–85.

Brennan, C., & Lunn, W. (2016). *Blockchain*. White paper, Credit Suisse, United Kingdom. https://tinyurl.com/y5f5k8ma

Brickson, S. L. (2007). Organizational Identity Orientation: The genesis of the role of the firm and distinct forms of social value. *Academy of Management Review, 32*(3), 864–888.

Briggs, R. O., Nunamaker Jr., J. F., & Sprague Jr., R. H. (2004). Special issue: Information systems design—theory and methodology. *Journal of Management Information Systems, 20*, 5–8. https://doi.org/10.1080/07421222.2004.11045781

vom Brocke, J., Fay, M., Schmiedel, T., Petry, M., Krause, F., & Teinzer, T. (2017). A journey of digital innovation and transformation: The case of Hilti. In G. Oswald & M. Kleinemeier (Eds.), *Shaping the digital enterprise* (pp. 237–251).

Brown, S. (2017). The new diversity: Working with Nonhumans. *IEEE Computer, 50*(3), 90–95.

Brown, T. J., Dacin, P. A., Pratt, M. G., & Whette, D. A. (2006). Identity, intended image, construed image, and reputation: An interdisciplinary framework and suggested terminology. *Journal of the Academy of Marketing Science, 34*(2), 99–106.

Brynjolfsson, E. (1993). The productivity paradox of information technology. *Communications of the ACM, 35*(12), 66–77.

Buckl, S., Ernst, A. M., Lankes, J., Schneider, K., & Schweda, C. M. (2007). A pattern based approach for constructing enterprise architecture management information models. In *Wirtschaftinformatik Proceedings* (pp.65).

Buckl, S., Ernst, A. M., Matthes, F., & Schweda, C. M. (2009a). An information model capturing the managed evolution of application landscapes. In *Advances in enterprise engineering III* (pp. 85–99). Springer.

Buckl, S., Ernst, A. M., Matthes, F., & Schweda, C. M. (2009b). Visual roadmaps for managed enterprise architecture evolution. In *10th International Conference on Software Engineering,* *Artificial Intelligences, Networking and Parallel/Distributed Computing (SNPD)* (pp. 352–357). IEEE.

Burgemeestre, B., Hulstijn, J., & Tan, Y. (2009). Rule-based versus principle-based regulatory compliance. In *Proceedings of the 2009 Conference on Legal Knowledge and Information Systems: JURIX 2009: The Twenty-Second Annual Conference* (pp. 37–46).

Calero, C., Ruiz, J., & Piattini, M. (2005). Classifying web metrics using the web quality model. *Online Information Review, 29*(3), 227–248. https://doi.org/10.1108/14684520510607560

Camarinha-Matos, L. M., & Afsarmanesh, H. (2004). *The emerging discipline of collaborative networks*, volume 149 of *IFIP International Federation for Information Processing* (pp. 3–16). Springer. ISBN: 978-1-402-08138-5. https://doi.org/10.1007/1-4020-8139-1_1

Camillus, J. C. (2008). Strategy as a wicked problem. *Harvard Business Review, 86*(5), 98–101. ISSN: 0017-8012

Capgemini. (2009). *TechnoVision 2012—bringing business technology to life*. Research report, Utrecht, the Netherlands.

Carvalho, J. (2012). Validation criteria for the outcomes of design science research. http://hdl.handle.net/1822/21713

Catlin, T., Patiath, P., & Segev, I. (2014). IInsurance companies' untapped digital opportunity. *Harvard Business Review*, (pp. 1–5).

CDD. (2017). Defesa Nacional. http://www.defesa.pt

Ceccagnoli, M., Forman, C., Huang, P., & Wu, D. J. (2012). Cocreation of value in a platform ecosystem: The case of enterprise software. *MIS Quarterly, 36*(1), 263–290.

Cennamo, C., Ozalp, H., & Kretschmer, T. (2018). Platform architecture and quality trade-offs of multihoming complements. *Information Systems Research, 29*(2), 461–478.

Cesarotti, V., Giuiusa, A., Kwan, S., Introna, V., & Spohrer, J. (2016). Designing multichannel value propositions to enhance value-cocreation phenomenon. *International Journal of Service Science, Management, Engineering, and Technology, 5*, 14–44. https://doi.org/10.4018/ijssmet.2014010102

CGD. (2017). CGD. https://www.cgd.pt

Chandra, L., Seidel, S., & Gregor, S. (2015). Prescriptive knowledge in IS research: Conceptualizing design principles in terms of materiality, action, and boundary conditions. In *Proceedings of the 48th Hawaii International Conference on System Sciences*. https://doi.org/10.1109/HICSS.2015.485

Charette, R. (1991). The risks with risk analysis. *Communications of the ACM, 34*(6), 106–107.

Chen, P. P. (1976). The entity–relationship model: Towards a unified view of data. *ACM Transactions on Database Systems, 1*(1), 9–36.

Chew, E. K. (2016). iSIM: An integrated design method for commercializing service innovation. *Information Systems Frontiers, 18*(3).

Ciborra, C. (1996). The platform organization: Recombining strategies, structures and surprises. *Organization Science, 7*(2), 103–118.

CoA. (2014). *Annual Report of the Netherlands Court of Auditors (Verslag 2014 Algemene Rekenkamer)*. Technical report, Government of the Netherlands. In Dutch.

Collis, D. J. (1994). How valuable are organizational capabilities? *Strategic Management Journal, 15*, 143–152.

Colombetti, G., & Torrance, S. (2009). Emotion and ethics: An inter-(en)active approach. *Phenomenology and the Cognitive Sciences,, 8*, 505–526.

Constantinides, P., Henfridsson, O., & Parker, G. G. (2018). Platforms and infrastructures in the digital age. *Information Systems Research, 29*(2), 381–400. https://doi.org/10.1287/isre.2018.0794

Cook-Greuter, S. (2005). Ego development: Nine levels of increasing embrace. https://tinyurl.com/cookgreuter2005egodevelopment

Cooper, H. M. (1988). Organizing knowledge syntheses: A taxonomy of literature reviews. *Knowledge in Society, 1*, 104. https://doi.org/10.1007/BF03177550

Corcho, O., Fernández-López, M., Gómez-Pérez, A., & López-Cima, A. (2005). Building legal ontologies with METHONTOLOGY and WebODE. In *Law and the semantic web* (pp. 142—157). Springer. ISBN: 978-3-540-32253-5

Cornelissen, J. P., Haslam, S. A., & Balmer, J. M. T. (2007). Social identity, organizational identity and corporate identity: Towards an integrated understanding of processes, patternings and products. *British Journal of Management, 18*, 1–16.

CPDP. (2014). *Privacy by design: Effective privacy management in the victorian public sector*. Technical report, Commissioner for Privacy and Data Protection, Victoria, Australia. https://tinyurl.com/victoria2014privacybydesign

D'Aveni, R. A. (1994). *Hypercompetition: Managing the dynamics of strategic maneuvering*. Free Press.

Dam, S. H. (2015). *DoD architecture framework 2.0: A guide to applying systems engineering to develop integrated, executable architectures*. CreateSpace Independent Publishing Platform.

Daneva, M. (2006). *Applying real options thinking to information security in networked organizations*. Technical Report 06-11, Centre for Telematics and Information Technology, University of Twente., Enschede, the Netherlands.

Davis, A., Dieste, O., Hickey, A., Juristo, N., & Moreno, A. M. (2006). Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. In *14th IEEE International Requirements Engineering Conference* (pp. 179–185). IEEE Computer Society.

Davis, M. M., Spohrer, J. C., & Maglio, P. P. (2011). Guest editorial: How technology is changing the design and delivery of services. *Operations Management Research, 4*, 1–5. https://doi.org/10.1007/s12063-011-0046-6

Dietz, J. L. G. (2006). *Enterprise ontology—theory and methodology*. Springer. ISBN: 978-3-540-29169-5

Dietz, J. L. G. (2008). *Architecture—building strategy into design*. The Hague. Netherlands Architecture Forum, Academic Service—SDU. ISBN: 978-9-012-58086-1. http://www.naf.nl

Dietz, J. L. G., & Hoogervorst, J. A. P. (2007). Enterprise ontology and enterprise architecture—how to let them evolve into effective complementary notions. *GEAO Journal of Enterprise Architecture, 2*(1), 121–149.

Dietz, J. L. G., Hoogervorst, J. A. P., Albani, A., Aveiro, D., Babkin, E., Barjis, J., Caetano, A., Huysmans, P., Iijima, J., van Kervel, S. J. H., Mulder, H., Op 't Land, M., Proper, H. A., Sanz, J., Terlouw, L., Tribolet, J. M., Verelst, J., & Winter, R. (2013). The discipline of enterprise engineering. *International Journal Organisational Design and Engineering, 3*(1), 86–114.

DiMaggio, P. J., & Powell, W. W. (1983). The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields. *American Sociological Review, 48*(2), 147–160.

Dix, A. (2009). Human–computer interaction: A stable discipline, a nascent science, and the growth of the long tail. *Interacting with Computers, 22*(1), 13–27. https://doi.org/10.1016/j.intcom.2009.11.007

DoD Deputy Chief Information Officer. (2011). The DoDAF Architecture Framework Version 2.02. http://tinyurl.com/zhw3kaf

Dolin, R. H., Alschuler, L., Boyer, S., Beebe, C., Behlen, F. M., Biron, P. V., & Shabo, A. (2006). HL7 clinical document architecture, release 2. *Journal of the American Medical Informatics Association, 13*(1), 30–39. https://doi.org/10.1197/jamia.M1888

Drews, G., & Hillebrand, N. (2002). *Lexikon der Projektmanagement-Methoden*. Haufe-Lexware.

Dukerich, J. M., Golden, B. R., & Shortell, S. M. (2002). Beauty is in the eye of the beholder: The impact of organizational identification, identity, and image on the cooperative behaviors of physicians. *Administrative Science Quarterly, 47*, 507–533.

Dunbar, R. L. M., Romme, A. G. L., & Starbuck, W. H. (2008). Creating better understanding of organizations while building better organizations. In D. Barry & H. Hansen (Eds.), *Sage handbook of new approaches in management and organization*. Sage Publications.

DutchICT. (2014). Conclusions and recommendations of the Dutch temporary committee on government ICT projects. https://tinyurl.com/y4ena42c

Dutton, J. E., & Dukerich, J. M. (1991). Keeping an eye on the mirror: Image and identity in organizational adaptation. *Academy of Management Journal, 34*(3), 517–554.

Earl, B. (2015). *The practice of social research*. Cengage Learning. ISBN: 978-1305104945

Eaton, B., Elaluf-Calderwood, S., & Sorensen, C. (2015). Distributed tuning of boundary resources: The case of Apple's iOS service system. *MIS Quarterly, 39*(1), 217–243.

Edvardsson, B., Tronvoll, B., & Gruber, T. (2011). Expanding understanding of service exchange and value co-creation: A social construction approach. *Journal of the Academy of Marketing Science, 39*(2), 327–339.

van Eijnatten, F. M. (2004). Co-jumping on a trampoline: The attractor landscape as a graphic metaphor for organizational transformation. In *Fourth Annual Meeting of the European Chaos and Complexity in Organisations Network (ECCON)*.

Eisenhardt, K. M. (1989). Building theories from case-study research. *Academy of Management Review, 14*(4), 532–550. https://doi.org/10.2307/258557

Eisenhardt, K. M., & Martin, J. (2000). Dynamic capabilities: What are they? *Strategic Management Journal, 21*(10/11), 1105–1121.

El Emam, K., & Koru, G. A. (2008). A replicated survey of it software project failures. *IEEE Software, 25*(5), 84–90.

Emery, F. E. (1977). *Futures we are in* volume 5. Springer.

Emery, F. E., & Trist, E. L. (1973). *Towards a social ecology: Contextual appreciations of the future in the present*. Plenum Publishing.

Engle, R., van der Aalst, W. M. P., Zapletal, M., Picher, C., & Werthner, H. (2012). Mining interorganisational business process models from EDI messages: A case study from the automotive sector. In J. Ralyté, J., X. Franch, S. Brinkkemper, & S. Wrycza (Eds.), *Proceedings of the 24th International Conference on Advanced Information Systems Engineering (CAiSE 2012), Gdansk, Poland*, volume 7328 of Lecture notes in computer science (pp. 222–237). Springer. ISBN: 978-3-642-31094-2

Erevelles, S., Fukawa, N., & Swayne, L. (2015). Big data consumer analytics and the transformation of marketing. *Journal of Business Research, 69*(2), 897–904.

Eusterbrock, T., & van Steenbergen, M. (2016). *Principle-based approach in enterprise architecture practice; finding the sweet spot*. Technical report, Sogeti. https://tinyurl.com/eusterbrock2016principlebased

Feltus, C. (2019). Deriving information system security and privacy from value cocreation theory: Case Study in the financial sector. *International Journal of Service Science, Management, Engineering, and Technology (IJSSMET), 10*(4). https://doi.org/10.4018/IJSSMET.2019100101

Feltus, C., & Proper, H. A. (2017a). Conceptualization of an abstract language to support value co-creation. In M. Ganzha, L. A. Maciaszek, & M. Paprzycki (Eds.), *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3–6, 2017* (pp. 971–980). ISBN: 978-83-946253-7-5. https://doi.org/10.15439/2017F9

Feltus, C., & Proper, H. A. (2017b). Towards a security and privacy co-creation method. In D. Fischer (Ed.), *12th International Conference for Internet Technology and Secured Transactions, ICITST 2017, Cambridge, United Kingdom, December 11–14, 2017* (pp. 75–80). IEEE Computer Society. ISBN: 978-1-908320-93-3. https://doi.org/10.23919/ICITST.2017.8356351

Feltus, C., Petit, M., & Dubois, E. (2009). Strengthening employee's responsibility to enhance governance of IT: COBIT RACI chart case study. In *Proceedings of the first ACM Workshop on Information Security Governance* (pp. 23–32). ACM. https://doi.org/10.1145/1655168.1655174

Feltus, C., Dubois, E., Proper, H. A., Band, I., & Petit, M. (2012). Enhancing the ArchiMate standard with a responsibility modeling language for access rights management. In M. Singh Gaur, A. Elçi, O. B. Makarevich, M. A. Orgun, & V. Singh (Eds.), *5th International Conference of Security of Information and Networks, SIN '12, Jaipur, India, October 22–26, 2012* (pp. 12–19). ACM Press. ISBN: 978-1-4503-1668-2. https://doi.org/10.1145/2388576.2388577

Feltus, C., Fontaine, F. X., & Grandry, E. (2015). Towards systemic risk management in the frame of business service ecosystem. In P. A. & S. J. (Eds.), *Advanced Information Systems*

Engineering Workshops, CAiSE 2015, number 215 in Lecture Notes in Business Information Processing (pp. 27–39). Springer. https://doi.org/10.1007/978-3-319-19243-7_3

Feltus, C., Grandry, E., Kupper, T., & Colin, J. N. (2018a). Model-driven approach for privacy management in business ecosystem. In *MODELSWARD 2017* (pp. 392–400). https://doi.org/10.5220/0006142203920400

Feltus, C., Lessard, L., Vernadat, F. B., Amyot, D., & Proper, H. A. (2018b). Conceptualization of a value cocreation language for knowledge-intensive business services. In E. Ziemba (Ed.), *Information Technology for Management. Ongoing Research and Development—15th Conference, AITM 2017, and 12th Conference, ISM 2017, Held as Part of FedCSIS, Prague, Czech Republic, September 3-6, 2017, Extended Selected Papers*, volume 311 of *Lecture Notes in Business Information Processing* (pp. 3–23). Springer. ISBN: 978-3-319-77720-7. https://doi.org/10.1007/978-3-319-77721-4_1

Feltus, C., Proper, H. A., & Haki, K. (2018c). Towards a language to support value cocreation: An extension to the ArchiMate modeling framework. In M. Ganzha, L. A. Maciaszek, & M. Paprzycki (Eds.), *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9–12, 2018* (pp. 751–760). ISBN: 978-83-949419-5-6. https://doi.org/10.15439/2018F27

Feltus, C., Proper, H. A., Metzger, A., Garcia Lopez, J. C., & Gonzalez Castineira, R. (2018d). Value cocreation (VCC) language design in the frame of a smart airport network case study. In L. Barolli, M. Takizawa, T. Enokido, M. R. Ogiela, L. Ogiela, & N. Javaid (Eds.), *32nd IEEE International Conference on Advanced Information Networking and Applications, AINA 2018, Krakow, Poland, May 16–18, 2018* (pp. 858–865). IEEE Computer Society. ISBN: 978-1-5386-2195-0. https://doi.org/10.1109/AINA.2018.00127

Fernández-López, M., Gomez-Perez, A., & Juristo, N. (1997). Methontology: From ontological art towards ontological engineering. *Engineering Workshop on Ontological Engineering (AAAI97)*.

Fernández López, M., Gómez-Pérez, A., Pazos Sierra, A., & Pazos Sierra, J. (1999). Building a chemical ontology using Methontology and the ontology design environment. *IEEE Intelligent Systems and Their Applications, 14*, 37–46.

Ferrin, B., & Plank, R. (2002). Total cost of ownership models: An exploratory study. *Journal of Supply Chain Management, 38*(3), 18–29.

Filipe, J. A., Coelho, M. F., Ferreira, M. A. M., & Figueredo, I. C. (2011). Social responsibility and environmental sustainability: The case of Caixa Geral de Depósitos (Portugal) and Vale (Brazil). *Editorial Board Members, 10*(5), 352–375.

Fischer, C., Winter, R., & Aier, S. (2010). What is an enterprise architecture design principle? Towards a consolidated definition. In *Proceedings of the 2nd International Workshop on Enterprise Architecture Challenges and Responses* Yonezawa, Japan.

Fonseca, F., & Martin, J. (2007). Learning the differences between ontologies and conceptual schemas through ontology-driven information systems. *Journal of the Association for Information Systems, 8*(2), 129–142.

Foorthuis, R. M., Hofman, F., Brinkkemper, S., & Bos, R. (2009). Assessing business and it projects on compliance with enterprise architecture. In *Proceedings of the Second International Workshop on Governance, Risk and Compliance held in conjunction with CAiSE'09 Conference, Amsterdam, the Netherlands*. http://ceur-ws.org/Vol-459/paper6.pdf

Fowler, M. (2004). *UML Distilled—a brief guide to the standard object modeling language* (3rd ed.). Object Technology Series. Addison-Wesley. ISBN: 0-321-19368-7

Fowler, J. W., Streib, H., & Keller, B. (2004). *Manual for faith development research*. Technical report, Center for Research in Faith and Moral Development; Center for Biographical Studies in Contemporary Religion.

Fragidis, G., & Tarabanis, K. (2011). Towards an ontological foundation of service dominant logic. In *International Conference on Exploring Services Science*.

Frank, U. (1999). Conceptual modelling as the core of the information systems discipline—perspectives and epistemological challenges. In D. W. Haseman, D. Nazareth, & D. Goodhue

(Eds.), *Proceedings of the Fifth America's Conference on Information Systems (AMCIS 99), Milwaukee, Wisconsin* (pp. 695–697). https://aisel.aisnet.org/amcis1999/240

Frank, U. (2011). Some guidelines for the conception of domain-specific modelling languages. In M. Nüttgens, O. Thomas, & B. Weber (Eds.), *Enterprise Modelling and Information Systems Architectures: Proceedings of the 4th International Workshop on Enterprise Modelling and Information Systems Architectures, EMISA 2011, Hamburg, Germany, September 22–23, 2011*, number 190 in Lecture Notes in Informatics (pp. 93–106). Gesellschaft für Informatik. ISBN: 978-3-88579-284-0 https://dl.gi.de/handle/20.500.12116/18504

Frank, U. (2013). Domain-specific modeling languages: Requirements analysis and design guidelines. In I. Reinhartz-Berger, A. Sturm, T. Clark, S. Cohen, & J. Bettin (Eds.), *Domain engineering—product lines, languages, and conceptual models*. Springer. ISBN: 978-3-642-36653-6

Freund, J., & Rücker, B. (2012). *Real life BPMN*. Camunda.

Friedman, T. L. (2005). *The world is flat: A brief history of the twenty-first century*. Farrar, Straus and Giroux. ISBN: 0-374-29288-4

Friend, S. B., & Malshe, A. (2016). Key skills for crafting customer solutions within an ecosystem: A theories-in-use perspective. *Journal of Service Research, 19*(2), 174–191.

Frow, P., Nenonen, S., Payne, A., & Storbacka, K. (2015). Managing co-creation design: A strategic approach to innovation. *British Journal of Management, 26*(3), 463–483. https://doi.org/10.1111/1467-8551.12087

FXSSI. (2019). Most valuable companies in the world—2019. https://fxssi.com/top-10-most-valuable-companies-in-the-world

Gadatsch, A., & Mayer, E. (2004). *Grundkurs IT-Controlling: Grundlagen—Strategischer Stellenwert—Kosten- und Leistungsrechnung in der Praxis.* Vieweg. In German.

Gailly, F., Roelens, B., & Guizzardi, G. (2016). The design of a core value ontology using ontology patterns. In *International Conference on Conceptual Modeling*.

Galbraith, J. R. (2000). *Designing the global corporation*. Jossey-Bass. ISBN: 978-0-787-95275-4

Galvagno, M., & Dalli, D. (2014). Theory of value co-creation: A systematic literature review. *Managing Service Quality, 24*(6), 643–683.

Gama, N., Mira da Silva, M., & Francisco, R. A. (2011). A conceptual framework for structuring an IT enterprise. In *16th Annual Conference of the UKAIS, Oxford, England, United Kingdom*.

Gama, N., Sousa, P., & Mira da Silva, M. (2015). A case of integration between ITIL and TOGAF. In *The 9th TEE Workshop on Transformation & Engineering of Enterprises (TEE'15), co-located with the 17th IEEE Conference on Business Informatics (CBI 2015), Lisbon, Portugal*. IEEE Computer Society.

Garrido-Pelaz, R., González-Manzano, L., & Pastrana, S. (2016). Shall we collaborate? A model to analyse the benefits of information sharing. In *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security (WISCS '16)* (pp. 15–24). ACM. https://doi.org/10.1145/2994539.2994543

Gartner Group. (1997). TCO analyst. A white paper on Gartner Group's next generation total cost of ownership methodology. http://www.netvoyager.co.uk/pdf/TCO_analyst.pdf

Gartner Group. (2017). Gartner says worldwide sales of smartphones grew 9 percent in first quarter of 2017. https://www.gartner.com/newsroom/id/3725117

Garud, R., Kumaraswamy, A., & Sambamurthy, V. (2006). Emergent by design: Performance and transformation at Infosys Technologies. *Organization Science, 17*(2), 277–286.

Garud, R., Jain, S., & Tuertscher, P. (2008). Incomplete by design and designing for incompleteness. *Organization Studies, 29*(3), 351–371.

Gawer, A. (2011). *Platforms, markets and innovation*. Edward Elgar Publishing.

Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: The boundary resources model. *Information Systems Journal, 23*(2), 173–192.

Ghoshal, S., Bartlett, C., & Moran, P. (1999). A new manifesto for management. *MIT Sloan Management Review*.

Giacomin, J. (2014). What is human centred design? *The Design Journal, 17*(4), 606–623.

Giebelhausen, M., Robinson, S. G., Sirianni, N. J., & Brady, M. K. (2013). Touch versus tech: When technology functions as a barrier or a benefit to service encounters. *Journal of Marketing, 78*(4), 113–124. https://doi.org/10.1509/jm.13.0056

Gill, G., & Bhattacherjee, A. (2009). Fashion waves versus informing: Response to Baskerville and Myers. *MIS Quarterly, 33*(4), 667–671. https://doi.org/10.2307/20650321

van Gils, B., & van Dijk, S. (2014). *The practice of enterprise architecture—experiences, techniques, and best practices.* BiZZdesign Academy. ISBN: 9789079240166

van Gils, B., & Proper, H. A. (2018). Enterprise modelling in the age of digital transformation. In R. A. Buchmann, D. Karagiannis, & M. Kirikova (Eds.), *The Practice of Enterprise Modeling—11th IFIP WG 8.1. Working Conference, PoEM 2018, Vienna, Austria, October 31–November 2, 2018, Proceedings*, volume 335 of Lecture notes in business information processing (pp. 257–273). Springer. ISBN: 978-3-030-02301-0. https://doi.org/10.1007/978-3-030-02302-7_16

van Gils, B., Proper, H. A., van Bommel, P., & van der Weide, T. P. (2006). Quality makes the information market. In R. Meersman & Z. Tari (Eds.), *On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE, OTM Confederated International Conferences, CoopIS, DOA, GADA, and ODBASE 2006, Montpellier, France, October 29–November 3, 2006. Proceedings, Part I*, volume 4275 of Lecture notes in computer science (pp. 345–359). Springer. ISBN: 3-540-48287-3. http://doi:10.1007/11914853_21

Gioia, D. A., & Chittipeddi, K. (1991). Sensemaking and sensegiving in strategic change initiation. *Strategic Management Journal, 12*(6), 19–64.

Glasl, F. (1997). *The enterprise of the future*. Hawthorn Press.

Gommans, L., van Oudenaarde, B., Dijkstra, F., de Laat, C., Lavian, L., Monga, I., Taal, A., Travostino, F., & Wan, A. (2006). Applications drive secure lightpath creation across heterogeneous domains. *IEEE Communications Magazine, 44*(3), 100–106.

Gordijn, J., & Akkermans, H. (2001). Designing and evaluating e-business models. *IEEE Intelligent Systems, 16*(4), 11–17. https://doi.org/10.1109/5254.941353

Gordijn, J., & Akkermans, H. (2003). Value based requirements engineering: Exploring innovative e-commerce ideas. *Requirements Engineering Journal, 8*(2), 114–134. https://doi.org/10.1007/s00766-003-0169-x

Gordijn, J., Akkermans, H., & van Vliet, H. (2000). Business modelling is not process modelling. In *Proceedings of the Workshops on Conceptual Modeling Approaches for E-Business and The World Wide Web and Conceptual Modeling: Conceptual Modeling for E-Business and the Web*, ER ’00 (pp. 40—51). Springer. ISBN: 3540410732

Gouillart, F. J., & Kelly, J. N. (1995). *Transforming the organization*. McGraw-Hill. ISBN: 0-07-034067-6

Gould, D. P. (1986). Stratified systems theory in the design of organization-wide information systems. *International Journal of Information Management, 6*(1), 5–15.

Goutas, L., Sutanto, J., & Aldarbesti, H. (2015). The building blocks of a cloud strategy: Evidence from three SaaS providers. *Communications of the ACM, 1*(59), 90–97.

Grönroos, C. (2008). Service logic revisited: Who creates value? And who co-creates? *European Business Review, 20*(4), 298–314.

Grönroos, C. (2011). Value co-creation in service logic: A critical analysis. *Marketing Theory, 11*(3), 279–301. https://doi.org/10.1177/1470593111408177

Grönroos, C., & Ravald, A. (2011). Service as business logic: Implications for value creation and marketing. *Journal of Service Management, 22*(1), 5–22. https://doi.org/10.1108/09564231111106893

Grönroos, C., & Voima, P. J. (2013). Critical service logic: Making sense of value creation and co-creation. *Journal of the Academy of Marketing Science, 41*(2), 133–150. https://doi.org/10.1007/s11747-012-0308-3

Grandry, E., Feltus, C., & Dubois, E. (2013). Conceptual integration of enterprise architecture management and security risk management. In *2013 17th IEEE International Enterprise Distributed Object Computing Conference Workshops*: IEEE. https://doi.org/10.1109/edocw.2013.19

Grant, R. (1996). Toward a knowledge-based theory of the firm. *Strategic Management Journal, 17*, 109–122.

Grant, R. M. (1999). Prospering in dynamically-competitive environments: Organizational capability as knowledge integration. *Knowledge and Strategy* (pp. 133—153).

Greefhorst, D., & Proper, H. A. (2011a). *Architecture principles—the cornerstones of enterprise architecture*. The Enterprise Engineering Series. Springer. ISBN: 978-3-642-20278-0. https://doi.org/10.1007/978-3-642-20279-7

Greefhorst, D., & Proper, H. A. (2011b). A practical approach to the formulation and use of architecture principles. In *Proceedings of the 6th Trends in Enterprise Architecture Research workshop (TEAR 2011), held in conjunction with the 15th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2011), Helsinki, Finland* (pp. 330–339). IEEE Computer Society Press. ISBN: 978-0-769-54426-7. https://doi.org/10.1109/EDOCW.2011.18

Greefhorst, D., Proper, H. A., & Plataniotis, G. (2013). The Dutch state of the practice of architecture principles. *Journal of Enterprise Architecture, 9*(4), 20–25.

Greenwood, R., & Hinings, C. R. (1993). Understanding strategic change—the contribution of archetypes. *Academy of Management Journal, 36*(5), 1052–1081. https://doi.org/10.2307/256645

Gregor, S., & Hevner, A. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly, 37*(2), 337–355.

Greiner, L. E. (1998). Evolution and revolution as organizations grow. *Harvard Business Review, 76*(3), 55–60.

Grieves, M. (2019). Virtually intelligent product systems: Digital and physical twins. In S. Flumerfelt, K. G. Schwartz, D. Mavris, & S. Briceno (Eds.), *Complex systems engineering: Theory and practice* (pp. 175–200). American Institute of Aeronautics and Astronautics. ISBN: 978-1624105647

Grob, H. (1993). *Capital budgeting with financial plans: An introduction*. Gabler.

Grosskopf, A., Decker, G., & Weske, M. (2009). *The process: Business process modeling using BPMN*. Meghan Kiffer Press.

Grover, V., & Kohli, R. (2012). Cocreating IT value: New capabilities and metrics for multifirm environments. *MIS Quarterly, 36*(1), 225–232.

Guédria, W., Naudet, Y., & Chen, D. (2013). Maturity model for enterprise interoperability. *Enterprise Information Systems* (pp. 1–28). https://doi.org/10.1080/17517575.2013.805246

Guarino, N. (1998). Formal ontology and information systems. In N. Guarino (Ed.), *Proceedings of FOIS'98, Trento, Italy* (pp. 3–15). IOS Press.

Guerreiro, S. (2015). Enterprise dynamic control systems. In M. Khosrow-Pour (Ed.), *Encyclopedia of information science and technology* (3rd ed., pp. 7122–7132). IGI Global.

Guillén, M. F. (2001). *The limits of convergence: Globalization and organizational change in Argentina, South Korea, and Spain*. Princeton University Press.

Guizzardi, G. (2006). On ontology, ontologies, conceptualizations, modeling languages, and (meta)models. In O. Vasilecas, J. Eder, & A. Caplinskas (Eds.), *Databases and Information Systems IV—Selected Papers from the Seventh International Baltic Conference, DB&IS 2006, July 3-6, 2006, Vilnius, Lithuania*, volume 155 of Frontiers in artificial intelligence and applications (pp. 18–39). IOS Press. ISBN: 978-1-58603-715-4

Guizzardi, G., & Proper, H. A. (2021). On understanding the value of domain modeling. In G. Guizzardi, T. P. Sales, C. Griffo, & M. Furnagalli (Eds.), *Proceedings of 15th International Workshop on Value Modelling and Business Ontologies (VMBO 2021), Bolzano, Italy, 2021*, volume 2835 of *CEUR Workshop Proceedings*. CEUR-WS.org. http://ceur-ws.org/Vol-2835/paper6.pdf

Gunderson, L. H., & Holling, C. S. (Eds.) (2002). *Panarchy: Understanding transformations in human and natural systems*. Island Press.

Gunther, W. (2014). *Measuring enterprise architecture effectiveness*. Ph.D. Thesis, Leiden University, Leiden, the Netherlands.

Gupta, A. K., Tesluk, P. E., & Taylor, M. S. (2007). Innovation at and across multiple levels of analysis. *Organization Science, 18*(6), 885–897. https://doi.org/10.1287/orsc.1070.0337

Högler, T. (2008). Framework for a holistic profitability analysis for mobile systems. In I. Kushchu (Ed.), *Proceedings of the 1st International Conference on Mobile Society (mSociety 2008) in combination with 3rd European Conference on Mobile Government, EURO mGOV 2008. Antalya, Turkey*.

Högler, T. (2012). Framework for a holistic evaluation of ICT. In *Proceedings of the 25th Bled eConference Graduate Student Consortium. Slovenia, Bled (21.06.2012)*.

Högler, T. (2014). A research agenda for mobile systems evaluation. In *Proceedings of the 16th ICEIS International Conference on Enterprise Information Systems. Lisbon, Portugal*.

Högler, T., & Versendaal, J. (2014). Identification of success factors for mobile systems deployment: A method. In *Proceedings of the 27th Bled eConference. Slovenia, Bled*.

Högler, T., & Versendaal, J. (2016). Determining the target system for mobile systems as part of an integrative approach for the economic impact of ICS: Validation at an SME. In *Proceedings of the 29th Bled eConference. Bled, Slovenia*.

Högler, T., & Versendaal, J. (2017). *On the validation of an evaluation framework: Assessment by experts*. Technical Report, HBO-Kennisbank.

Högler, T., Versendaal, J., & Batenburg, R. S. (2015). Evaluation of mobile systems—an integrative framework. In *American Conference on Information Systems (AMCIS), Fajardo, Puerto Rico*.

Haeckel, S. H. (1999). *Adaptive enterprise: Creating and leading sense-and-respond organizations*. Harvard Business School Press.

Hagel III, J., & Armstrong, A. G. (1997). *Net gain—expanding markets through virtual communities*. Harvard Business School Press.

Hagel III, J., & Singer, M. (1999). Unbundling the corporation. *Harvard Business Review*.

Haki, M. K. & Legner, C. (2012). New avenues for theoretical contributions in enterprise architecture principles—a literature review. In [Aier, S., Ekstedt, M., Matthes, F., Proper, H. A., & Sanz, J. (Eds.) (2012). *Trends in Enterprise Architecture Research and Practice-Driven Research on Enterprise Transformation—7th Workshop, TEAR 2012, and 5th Working Conference, PRET 2012, Held at The Open Group Conference 2012, Barcelona, Spain, October 23-24, 2012. Proceedings*, volume 131 of *Lecture Notes in Business Information Processing*. Springer. ISBN: 978-3-642-34162-5. http://doi:10.1007/978-3-642-34163-2], (pp. 182–197). ISBN: 978-3-642-34162-5. https://doi.org/10.1007/978-3-642-34163-2_11

Haki, M. K., & Legner, C. (2013). Enterprise architecture principles in research and practice: Insights from an exploratory analysis. In *Proceedings of the 21st European Conference on Information Systems (ECIS 2013), Utrecht, the Netherlands* (p. 204).

Haki, K., & Legner, C. (2021). The mechanics of enterprise architecture principles. *Journal of the Association for Information Systems, 22*(5), 1334–1375.

Haki, K., Blaschke, M., Aier, S., & Winter, R. (2019). A value co-creation perspective on information systems analysis and design. *Business & Information Systems Engineering, 61*(4), 487–502.

Haki, K., Beese, J., Aier, S., & Winter, R. (2020). The evolution of information systems architecture: An agent-based simulation model. *MIS Quarterly, 44*(1), 155–184.

Halpin, T. A. (2001). *Information modeling and relational databases, from conceptual analysis to logical design*. Morgan Kaufmann. ISBN: 1-558-60672-6

Halpin, T. A., & Morgan, T. (2008). *Information modeling and relational databases* (2nd ed.). Data Management Systems. Morgan Kaufman. ISBN: 978-0-123-73568-3

Hamel, G., & Välikangas, L. (2003). The quest for resilience. *Harvard Business Review, 8*(9).

Hamers, R. (2017). We want to be a tech company with a banking license. https://tinyurl.com/ycn8l2kh

Hammer, M. (1990). Re–engineering work: Don't automate, obliterate. *Harvard Business Review, 68*(4), 104–112.

Hammer, M., & Champy, J. (1993). *Reengineering the corporation: A manifesto for business revolution*. Harper Business.

Han, K., Oh, W., Im, K. S., Chang, R. M., Oh, H., & Pinsonneault, A. (2012). Value cocreation and wealth spillover in open innovation alliances. *MIS Quarterly, 36*(1), 291–315.

Harmsen, A. F., Proper, H. A., & Kok, N. (2009). Informed governance of enterprise transformations. In H. A. Proper, A. F. Harmsen, & J. L. G. Dietz (Eds.), *Advances in Enterprise Engineering II—First NAF Academy Working Conference on Practice-Driven Research on Enterprise Transformation, PRET 2009, held at CAiSE 2009, Amsterdam, the Netherlands, June 11, 2009. Proceedings*, volume 28 of Lecture notes in business information processing (pp. 155–180). Springer. ISBN: 978-3-642-01858-9 https://doi.org/10.1007/978-3-642-01859-6_9

Harquail, C. V., & King, A. W. (2010). Construing organizational identity: The role of embodied cognition. *Organization Studies, 31*(2), 1619–1648.

Hawley, M., Howard, P., Koelle, R., & Saxton, P. (2013). Collaborative security management: Developing ideas in security management for air traffic control. In *International Conference on Availability, Reliability and Security, Regensburg* (pp. 802–806). IEEE Computer Society Press. https://doi.org/10.1109/ARES.2013.107

Hay, D. C. (2011). *UML and data modeling: A reconciliation*. Technics Publications. ISBN: 978-1935504191

van Heck, E., & Vervest, P. H. M. (2007). Smart business networks: How the network wins. *Communications of the ACM, 50*(6), 28–37.

Heerink, D. (2014). Should health records go mobile? Exploring a mobile health record application in its support to process and quality improvement within hospitals. Master's Thesis, Utrecht University, Utrecht, the Netherlands.

Helfat, C. E., & Peteraf, M. A. (2003). The dynamic resource-based view: Capability lifecycles. *Strategic Management Journal, 24*(10), 997–1010.

Henderson, J. C., & Venkatraman, N. (1993). Strategic alignment: Leveraging information technology for transforming organizations. *IBM Systems Journal, 32*(1), 4–16.

Henfridsson, O., & Bygstad, B. (2013). The generative mechanisms of digital infrastructure evolution. *MIS Quarterly, 37*(3), 907–931.

Herzfeldt, A. B., Rastorguev, E., Ertl, C., & Krcmar, H. (2016). Value facilitation as antecedent of service provider profitability. *International Journal of Service Science, Management, Engineering, and Technology, 7*(4), 63–76. https://doi.org/10.4018/ijssmet.2016100105

Hess, E. D. (2014). *Learn or die: Using science to build a leading-edge learning organization*. Columbia University Press. ISBN: 978-0231170246

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian Journal of Information Systems, 19*(2), 87–92.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*, 75–106.

Hine, D., Parker, R., Pregelj, L., & Verreynne, M.-L. (2013). Deconstructing and reconstructing the capability hierarchy. *Industrial and Corporate Change, 23*(5), 1299–1325.

Hitt, M., & Ireland, R. D. (1985). Corporate distinctive competence, strategy, industry and performance. *Strategic Management Journal, 6*(3), 273–293.

Hoebeke, L. (1994). *Making work systems better: A practitioner's reflections*. John Wiley & Sons.

ter Hofstede, A. H. M., & van der Weide, T. P. (1993). Expressiveness in conceptual data modelling. *Data & Knowledge Engineering, 10*(1), 65–100.

ter Hofstede, A. H. M. & Proper, H. A. (1998). How to formalize it? Formalization principles for information system development methods. *Information and Software Technology, 40*(10), 519–540. https://doi.org/10.1016/S0950-5849(98)00078-0

Hoogervorst, J. A. P. (2009). *Enterprise governance and enterprise engineering*. The Enterprise Engineering Series. Springer. ISBN: 978-3-540-92670-2

Hoppenbrouwers, S. J. B. A. (2000). A functionalist approach to conceptualisation. In *Proceedings of the Fourth International Workshop on the Language Action Perspective on Communication Modelling (LAP 2000)*, Aachener Informatik-Berichte Aachen, Germany: RWTH Aachen.

Hoppenbrouwers, S. J. B. A. (2003). *Freezing language; Conceptualisation processes in ICT supported organisations*. PhD thesis, University of Nijmegen, Nijmegen, the Netherlands. ISBN: 9-090-17318-8

Hoppenbrouwers, S. J. B. A., Proper, H. A., & van der Weide, T. P. (2005a). Formal modelling as a grounded conversation. In G. Goldkuhl, M. Lind, & S. Haraldson (Eds.), *Proceedings of the 10th International Working Conference on the Language Action Perspective on Communication Modelling (LAP'05)* (pp. 139–155). Kiruna, Sweden: Linköpings Universitet and Hogskolan I Boras, Linköping, Sweden.

Hoppenbrouwers, S. J. B. A., Proper, H. A., & van der Weide, T. P. (2005b). Understanding the requirements on modelling techniques. In O. Pastor & J. Falcao e Cunha (Eds.), *Advanced Information Systems Engineering, 17th International Conference, CAiSE 2005, Porto, Portugal, June 13-17, 2005, Proceedings*, volume 3520 of Lecture notes in computer science (pp. 262–276). Springer. ISBN: 3-540-26095-1. https://doi.org/10.1007/11431855_19

Hoppenbrouwers, S. J. B. A., Proper, H. A., & Nijssen, M. (2019). Towards key principles of fact based thinking. In C. Debruyne, H. Panetto, W. Guédria, P. Bollen, I. Ciuciu, & R. Meersman (Eds.), *On the Move to Meaningful Internet Systems: OTM 2018 Workshops—Confederated International Workshops: EI2N, FBM, ICSP, and Meta4eS 2018, Valletta, Malta, October 22-26, 2018, Revised Selected Papers*, volume 11231 of Lecture notes in computer science (pp. 77–86). Springer. ISBN: 978-3-030-11682-8. https://doi.org/10.1007/978-3-030-11683-5%5C_8

Horan, T. A. (2000). *Digital places—building our city of bits*. The Urban Land Institute (ULI). ISBN: 0-874-20845-9

Horváth, P. (1988). Grundprobleme der Wirtschaftlichkeitsanalyse beim Einsatz neuer Informations- und Produktionstechnologien. In P. Horváth (Ed.), *Wirtschaftlichkeit neuer Produktions- und Informationstechnologien. Konferenzband zum Stuttgarter Controller-Forum, 14.-15.09.1988, Stuttgart*: C.E. Poeschel, Stuttgart. In German.

Hughes, D. L., Dwivedi, Y. K., Simintiras, A. C., & Rana, N. P. (2015). *Success and failure of IS/IT projects: A state of the art analysis and future directions*. Springer.

Hugos, M. H. (2004). *Building the real-time enterprise: An executive briefing*. Wiley. ISBN: 978-0471678298

Hwang, C.-L., & Yoon, K. (1981). Methods for multiple attribute decision making. In C.-L. Hwang & K. Yoon (Eds.), *Multiple attribute decision making* (pp. 58–191). Springer.

IEEE. (1990). *IEEE standard glossary of software engineering terminology*. Technical Report ANSI/IEEE Std 610.12, IEEE Standards Department, Los Alamitos, California.

IEEE. (2000). *Recommended practice for architectural description of software intensive systems*. Technical Report IEEE P1471:2000, ISO/IEC 42010:2007, IEEE Explore, Los Alamitos, California, Piscataway, New Jersey. The Architecture Working Group of the Software Engineering Committee, Standards Department, IEEE. ISBN: 0-738-12518-0

IIBA. (2015). *A guide to the Business Analysis Body of Knowledge (BABOK Guide). 2015*. International Institute of Business Analysis. ISBN: 978-1927584026

Iivari, J. (2015). Distinguishing and contrasting two strategies for design science research. *European Journal of Information Systems, 24*(1), 107–115. https://doi.org/10.1057/ejis.2013.35

Iivari, J., Hirschheim, R., & Klein, H. K. (2000). A dynamic framework for classifying information systems development methodologies and approaches. *Journal of Management Information Systems, 17*(3), 179–218.

Iivari, J., Parsons, J., & Wand, Y. (2006). Research in information systems analysis and design: Introduction to the special issue. *Journal of the Association for Information Systems, 7*(8), 509–513.

IMG. (2015). Succeed with it projects. https://imgcorp.co.uk/projects-managed-success/

Isaac, H., & Leclercq, A. (2006). Give me a mobile phone, and I will work harder! Assessing the value of mobile technologies in organizations: An exploratory research. In *Proceedings of the International Conference on Mobile Business ICMB, Denmark, Copenhagen.*

ISO. (2013). *ISO/IEC/IEEE 42010:2011—systems and software engineering—architecture description*. Standard, International Organization for Standardization, Geneva, Switzerland.

ISO/IEC/IEEE. (2015). *Systems and software engineering—Vocabulary*. Technical Report ISO/IEC/IEEE 24765:2010(E), IEEE.

Jacobson, I. (2007). Enterprise architecture has failed in a big way! https://tinyurl.com/y2hnt8sq

Jaques, E. (1998). *Requisite organization: A total system for effective managerial organization and managerial leadership for the 21st century*. Cason Hall & Co. Publishers, revised second edition edition.

Jarzabkowski, P. (2005). *Strategy as practice: An activity based approach*. Sage.

Jenkins. (2017). Jenkins. https://jenkins.io

Johnson, J. (2007). *The public execution of miss scarlet*. The Standish Group International. ISBN: 978-1424320639

Johnson, J. (2013). *Emotional maturity research report*. Technical report, The Standish Group International.

Johnson, J. (2016). *The good sponsor*. The Standish Group International. ISBN: 9781365149016

Joiner, B., & Josephs, S. (2007). *Leadership agility: Five levels of mastery for anticipating and initiating change.* John Wiley & Sons.

Jonkers, H., Veldhuijzen van Zanten, G. E., van Buuren, R., Arbab, F., de Boer, F. S., Bonsangue, M., Bosma, H., ter Doest, H., Groenewegen, L., Guillen Scholten, J., Hoppenbrouwers, S. J. B. A., Iacob, M.-E., Janssen, W., Lankhorst, M. M., van Leeuwen, D., Proper, H. A., Stam, A., & van der Torre, L. (2003). Towards a language for coherent enterprise architecture descriptions. In M. Steen & B. R. Bryant (Eds.), *7th International Enterprise Distributed Object Computing Conference (EDOC 2003), 16-19 September 2003, Brisbane, Australia, Proceedings* (pp. 28–39). IEEE Computer Society. ISBN: 0-769-51994-6. https://doi.org/10.1109/EDOC.2003.1233835

Josefy, M. A., Harrison, J. S., Sirmon, D. G., & Carnes, C. (2017). Living and dying: Synthesizing the literature on firm survival and failure across stages of development. *Academy of Management Annals, 11*(2), 770–799.

Josey, A., Lankhorst, M., Band, I., Jonkers, H., & Quartel, D. (2016). *An introduction to the archimate 3.0 specification*. White paper, The Open Group.

Joy, B. K., & Kennedy, K. (1999). *Information Technology Research: Investing in Our Future*. Technical report, President's Information Technology Advisory Committee.

Kaisler, S. H., Armour, F., & Valivullah, M. (2005). Enterprise architecting: Critical problems. In *38th Hawaii International Conference on System Sciences (HICSS)* (pp. 224b–224b). IEEE.

Kane, G. C., Nguyen Phillips, A., Copulsky, J. R., & Andrus, G. R. (2019). *The technology fallacy—how people are the real key to digital transformation*. Management on the cutting edge series. MIT Press. https://www.ebook.de/de/product/34446834/gerald_c_kane_anh_nguyen_philips_jonathan_r_copulsky_garth_r_andrus_technology_fallacy.html

Kaplan, R. S., & Norton, D. P. (1992). The balanced scorecard—measures that drive performance. *Harvard Business Review* 71–79.

Karhu, K., Gustafsson, R., & Lyytinen, K. (2018). Exploiting and defending open digital platforms with boundary resources: Android's five platform forks. *Information Systems Research, 29*(2), 479–497. https://doi.org/10.1287/isre.2018.0786

Kazan, E., Tan, C. W., Lim, E. T. K., Sorensen, C., & Damsgaard, J. (2018). Disentangling digital platform competition: The case of UK mobile payment platforms. *Journal of Management Information Systems, 35*(1), 180–219. https://doi.org/10.1080/07421222.2018.1440772

Keen, P., & Williams, R. (2013). Value architectures for digital business: Beyond the business model. *MIS Quarterly, 37*(2), 642–647.

Kim, D. J., & Kogut, B. (1996). Technological platforms and diversification. *Organization Science, 7*(3), 283–301.

King, P. M., & Kitchener, K. S. (2004). Reflective judgment: Theory and research on the development of epistemic assumptions through adulthood. *Educational Psychologist, 39*(1), 5–8.

Kirchmer, M. (1999). *Market-and product-oriented design of business processes* (pp. 131–143). Springer.

Kishore, R., Sharman, R., & Ramesh, R. (2004a). Computational ontologies and information systems I: Foundations. *Communications of the Association for Information Systems, 14*, 158–183.

Kishore, R., Zhang, H., & Ramesh, R. (2004b). A helix–spindle model for ontological engineering. *Communications of the ACM, 47*(2), 69–75.

Klabon, M. (2007). *An investigation of the quantification of the probability of occurrence of software engineering project risks with Bayesian probability*. PhD Thesis, Naval Postgraduate School, Monterey, California.

Kogut, B., & Zander, U. (1996). What firms do? Coordination, identity, and learning. *Organization Science, 7*(5), 502–518.

Korhonen, J. J. (2016). *Digital disruption of industry: Tectonic tremor that cannot be ignored*. Disruption Brief 3, Digital Disruption of Industry Consortium.

Korhonen, J. J., & Halén, M. (2017). Enterprise architecture for digital transformation. In *IEEE 19th Conference on Business Informatics* Thessaloniki, Greece.

Korkman, O. (2006). *Customer value formation in practice: A practice-theoretical approach*. PhD thesis. ISBN: 951-555-905-7. http://urn.fi/URN:ISBN:951-555-905-7

Kosanke, K. (1995). CIMOSA—overview and status. *Computers in Industry, 27*(2), 101–109.

Kosanke, K., Vernadat, F., & Zelm, M. (1999). CIMOSA: Enterprise engineering and integration. *Computers in Industry, 40*(2), 83–97.

Krippendorff, K. (1989). On the essential contexts of artifacts or on the proposition that "Design Is Making Sense (of Things)". *Design Issues, 5*(2), 9–39.

Krippendorff, K. (2006). *The Semantic Turn: A new foundation for design*. Taylor & Francis.

Krippendorff, K. (2011). Principles of design and a trajectory of artificiality. *Journal of Production and Innovation Management, 28*, 411–418.

Krogstie, J., Lindland, O. I., & Sindre, G. (1995). Defining quality aspects for conceptual models. In E. D. Falkenberg, W. Hesse, & A. Olivé (Eds.), *Information System Concepts: Towards a consolidation of views—Proceedings of the third IFIP WG8.1 conference (ISCO–3)* (pp. 216–231). Chapman & Hall/IFIP WG8.1.

Kronsteiner, R., & Thurnher, B. (2009). Opportunities and risks for mobile decision support. In I. Khalil (Ed.), *Handbook of research on mobile multimedia* (pp. 93–104). IGI Global, Hershey, Pennsylvania. ISBN: 9781605660462. https://doi.org/10.4018/978-1-60566-046-2.ch007

de Kruijff, J., & Weigand, H. (2017). Understanding the blockchain using enterprise ontology. In E. Dubois & K. Pohl (Eds.), *Advanced information systems engineering* (pp. 29–43). Springer. ISBN: 978-3-319-59536-8. https://doi.org/10.1007/978-3-319-59536-8_3

Kuo, B. (1995). *Digital control systems*. Prentice Hall.

Kurtz, C. F., & Snowden, D. J. (2003). The new dynamics of strategy: Sense-making in a complex and complicated world. *IBM Systems Journal, 42*(3), 462–483.

Laloux, F. (2014). *Reinventing organizations: A guide to creating organizations inspired by the next stage of human consciousness*. Nelson Parker.

Langheinrich, M. (2001). Privacy by design-principles of privacy-aware ubiquitous systems. In G. D. Abowd, B. Brumitt, & S. Shafer (Eds.), *Ubicomp 2001: Ubiquitous Computing. UbiComp 2001*, number 2201 in Lecture Notes in Computer Science (pp. 273–291). Springer. https://doi.org/10.1007/3-540-45427-6_23

Lankhorst, M. M., Proper, H. A., & Jonkers, H. (2010). The anatomy of the ArchiMate language. *International Journal of Information System Modeling and Design, 1*(1), 1–32. https://doi.org/10.4018/jismd.2010092301

Lankhorst, M. M., Hoppenbrouwers, S. J. B. A., Jonkers, H., Proper, H. A., van der Torre, L., Arbab, F., de Boer, F. S., Bonsangue, M., Iacob, M.-E., Stam, A. W., Groenewegen, L., van Buuren, R., Slagter, R. J., Campschroer, J., Steen, M. W. A., Bekius, S. F., Bosma, H., Cuvelier, M. J., ter Doest, H. W. L., van Eck, P. A. T., Fennema, P., Jacob, J., Janssen, W. P. M., Jonkers, H., Krukkert, D., van Leeuwen, D., Penders, P. G. M., Veldhuijzen van Zanten, G. E., & Wieringa, R. J. (2017). *Enterprise architecture at work—modelling, communication and analysis* (4th ed.). The Enterprise Engineering Series. Springer. ISBN: 978-3-662-53932-3. https://doi.org/10.1007/978-3-662-53933-0

Lapalme, J., Gerber, A., Van der Merwe, A., Zachman, J. A., De Vries, M., & Hinkelmann, K. (2016). Exploring the future of enterprise architecture: A Zachman perspective. *Computers in Industry, 79*, 103–113.

Le Moigne, J.-L. (1985). La theorie du systeme general. In J.-L. Le Moigne (Ed.), *Theorie de la modelisation*, Collection Les Classiques du Reseau Intelligence de la Complexité. Presses Universitaires de France. In French. ISBN: 2130465153

Lehman, M. M. (1980). Programs, life cycles, and laws of software evolution. *Proceedings of the IEEE, 68*, 1060–1076. https://doi.org/10.1109/PROC.1980.11805

Lemey, E., & Poels, G. (2011). Towards a service system ontology for service science. In *International Conference on Service-Oriented Computing*.

Lepak, D. P., Smith, K. G., & Taylor, M. S. (2007). Value creation and value capture: A multilevel perspective. *Academy of Management Review, 32*(1), 180–194.

Lessard, L. (2015). Modeling value cocreation processes and outcomes in knowledge-intensive business services engagements. *Service Science, 7*(4), 181–195.

Li, F. (2017). *Performance evaluation and decision support in industrial system management: A benefit-cost-value-risk based methodology*. PhD Thesis, ENSAM, Paris, France.

Li, F., Etienne, A., Siadat, A., & Vernadat, F. (2017). A performance evaluation methodology for decision support in industrial projects. In *Proceedings of the 7th IESM Conference, Saarland University of Applied Sciences, Germany*.

Liles, D. H., & Presley, A. (1996). Enterprise modeling within an enterprise engineering framework. In *Winter Simulation Conference, San Diego, California*.

Liles, D. H., Johnson, M. E., Meade, L. M., & Underdown, D. R. (1995). Enterprise engineering: A discipline? In *Proceeding of the Society for Enterprise Engineering Conference*. Society for Enterprise Engineering.

Lillehagen, F., & Krogstie, J. (2010). *Active knowledge modeling of enterprises*. Springer. ISBN: 978-3-540-79415-8

Lindström, A. (2006). On the syntax and semantics of architectural principles. In *Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS 2006)*.

Link Consulting. (2017). Atlas. https://atlas.linkconsulting.com

Loebbecke, C., & Picot, A. (2015). Reflections on societal and business model transformation arising from digitization and big data analytics: A research agenda. *Journal of Strategic Information Systems, 24*(3), 149–157.

Loukis, E., Janssen, M., Dawes, S., & Zheng, L. (2016). Evolving ICT and governance in organizational networks—conceptual and theoretical foundations. *Electronic Markets, 26*(1), 7–14.

Loveman, G. (1994). An assessment of productivity impact of information technologies. In T. Allen & M. S. Morton (Eds.), *Information technology and the corporation of the 1990's*. Oxford University Press.

Lusch, R. F., & Nambisan, S. (2015). Service innovation: A service-dominant logic perspective. *MIS Quarterly, 39*(1), 155–175.

Lusch, R. F., & Vargo, S. L. (2006). Service-dominant logic: Reactions, reflections and refinements. *Marketing Theory, 6*, 281–288.

Lusch, R. F., Vargo, S. L., & O'Brien, M. (2007). Competing through service: Insights from service-dominant logic. *Journal of Retailing, 83*(1), 5–18. https://doi.org/10.1016/j.jretai.2006.10.002

Lusch, R. F., Vargo, S. L., & Wessels, G. (2008). Toward a conceptual foundation for service science: Contributions from service-dominant logic. *IBM Systems Journal, 471*(1), 5–14.

Lyytinen, K., Yoo, Y., & Boland, R. J. (2016). Digital product innovation within four classes of innovation networks. *Information Systems Journal, 26*(1), 47–75.

Macdonald, I., Burke, C., & Stewart, K. (2006). *Systems leadership: Creating positive organisations*. Gower.

Magalhães, R. (2011). Re-interpreting organization design in the light of enacted cognition theory. *Systems Research & Behavioral Science, 28*(6), 663–679.

Magalhães, R. (Ed.) (2014). *Organization design and engineering: Coexistence, cooperation or integration*. Palgrave-Macmillan. ISBN: 13 9781137351562

Magalhães, R. (2020). A designerly discourse for organization design: Adopting foundations from human-centred design. *Paper submitted to Design Issues*.

Magalhães, R., & Proper, H. A. (2017). Model-enabled design and engineering of organisations. *Organisational Design and Enterprise Engineeering, 1*(1), 1–12. https://doi.org/10.1007/s41251-016-0005-9

Magalhães, R., & Silva, A. R. (2009). *Organizational design and engineering*. Working paper, INOV, Lisbon, Portugal.

Magalhães, R., & Tribolet, J. M. (2009). Organization design as capability design: A service-dominant perspective. In S. G. Costa, L. M. Vieira, & J. N. Rodrigues (Eds.), *Ventos de Mudança, Editora Fundo de Cultura, Rio de Janeiro, Brasil*, volume Engenharia Organizacional: das partes ao todo e do todo às partes na dialéctica entre pessoas e sistemas (Organizational Engineering: From the parts to the whole and from the whole to the parts in a dialectic between people and systems). In Portuguese.

Magalhães, R., Zacarias, M., & Tribolet, J. M. (2007). Making sense of enterprise architectures as tools of organizational self-awareness. *Journal of Enterprise Architecture, 3*, 64–72.

Magalhães, R., Sousa, P., & Tribolet, J. M. (2008). The role of business processes and enterprise architectures in the development of organizational self-awareness. *Tékhne, 9*, 1–22. ISSN: 1645-9911

Maglio, P. P., Vargo, S. L., Caswel, N., & Spohrer, J. (2009). The service system is the basic abstraction of service science. *Information Systems and e-business Management, 7*(4), 395–406. https://doi.org/10.1007/s10257-008-0105-1

Maier, M., Emery, D., & Hilliard, R. (2001). Software architecture: Introducing IEEE standard 1471. *Computer, 34*(4), 107–109.

Maier, A. M., Moultrie, J., & Clarkson, P. J. (2012). Assessing organisational capabilities: Reviewing and guiding the development of maturity grids. *IEEE Transactions on Engineering Management, 59*(1), 138–159.

Maimbo, H., & Pervan, G. (2005). Designing a case study protocol for application in IS research. In P. Chau (Ed.), *Proceedings of the Ninth Pacific Asia Conference on Information Systems* (pp. 1281–1292).

Malone, T. (2004). Making the decision to decentralize. *Harvard business school—working knowledge for business leaders*.

Mannaert, H., & Verelst, J. (2009). *Normalized systems: Re-creating information technology based on laws for software evolvability*. Koppa. ISBN: 978-90-77160-00-8

Mannaert, H., Verelst, J., & De Bruyn, P. (2016). *Normalized systems theory, from foundations for evolvable software toward a general theory for evolvable design*. ISBN: 978-90-77160-091

Manuj, I., & Mentzer, J. T. (2008). Global supply chain risk management. *Journal of Business Logistics, 29*(1), 133–155.

Marosin, D., & Ghanavati, S. (2015). Measuring and managing the design restriction of enterprise architecture (EA) principles on EA models. In *Eighth IEEE International Workshop on Requirements Engineering and Law, RELAW 2015, Ottawa, Ontario, Canada, August 25, 2015* (pp. 37–46). IEEE Computer Society. ISBN: 978-1-5090-0104-0. https://doi.org/10.1109/RELAW.2015.7330210

Marosin, D., van Zee, M., & Ghanavati, S. (2016). Formalizing and modeling enterprise architecture (EA) principles with goal-oriented requirements language (GRL). In S. Nurcan, P. Soffer, M. Bajec, & J. Eder (Eds.), *Proceedings of the 28th International Conference on Advanced Information Systems Engineering (CAiSE 2016), Ljubljana, Slovenia*, volume 9694 of Lecture Notes in Computer Science (pp. 205–220). Springer. ISBN: 978-3-319-39695-8. https://doi.org/10.1007/978-3-319-39696-5_13

Marr, B. (2018). The 4th Industrial Revolution: How Mining Companies Are Using AI, Machine Learning And Robots. https://tinyurl.com/y2evvodz

Mata, F. J., Fuerst, W. L., & Barney, J. B. (1995). Information technology and sustained competitive advantage: A resource-based analysis. *MIS Quarterly, 19*(4), 487–505. https://doi.org/10.2307/249630

Matulevicius, R., Heymans, P., & Opdahl, A. L. (2008a). Ontological analysis of KAOS using separation of reference. In T. A. Halpin, J. Krogstie, & H. A. Proper (Eds.), *Proceedings of the 13th Workshop on Exploring Modeling Methods for Systems Analysis and Design (EMMSAD 2008), Held in Conjunction with the 20th Conference on Advanced Information Systems Engineering (CAiSE 2008), Montpellier, France*, volume 337 (pp. 395–406). CEUR-WS.org. http://ceur-ws.org/Vol-337

Matulevicius, R., Mayer, N., & Heymans, P. (2008b). Alignment of misuse cases with security risk management. In *Third International Conference on Availability, Reliability and Security, Barcelona, 2008* (pp. 1397–1404). IEEE. https://doi.org/10.1109/ARES.2008.88

McColl-Kennedy, J. R., Vargo, S. L., Dagger, T. S., Sweeney, J. C., & Kasteren, Y. V. (2012). Health care customer value cocreation practice styles. *Journal of Service Research, 15*(4), 370–389.

McMorland, J. (2005). Are you big enough your job? Is your job big enough for you? Exploring Levels of Work in organisations. *University of Auckland Business Review, 7*(2), 75–83.

Melville, N., Kraemer, K., & Gurbaxani, V. (2004). Review: Information technology and organizational performance: An integrative model of it business value. *MIS Quarterly, 28*(2), 283–322. https://doi.org/10.2307/25148636

Meriam–Webster. (2003). Meriam–webster online, collegiate dictionary. http://www.merriam-webster.com

Metzger, A., Thornton, J., Valverde, F., Lopez, J. F. G., & Rublova, D. (2019). Predictive analytics and predictive maintenance innovation via big data: The case of Transforming Transport. In *13th Intelligent Transport Systems—European Congress (ITS Europe), Brainport-Eindhoven, the Netherlands*.

Mingers, J. (2001). Combining IS research methods: Towards a pluralist methodology. *Information Systems Research, 12*(3), 240–259.

Moody, D. L. (1998). Metrics for evaluating the quality of entity relationship models. In *17th International Conference on Conceptual Modeling* (pp. 211–225). Springer.

Moody, D. L. (2003). Measuring the quality of data models: An empirical evaluation of the use of quality metrics in practice. In *11th European Conference on Information Systems (ECIS)* (pp. 78–95).

Moody, D. L. (2009). The "Physics" of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering, 35*(6), 756–779. https://doi.org/10.1109/TSE.2009.67

Moody, D. L., & Shanks, G. G. (2003). Improving the quality of data models: Empirical validation of a quality management framework. *Information Systems, 28*(6), 619–650.

Morris, C. W. (1928). Foundation of the theory of signs. In *International Encyclopedia of Unified Science*, volume 1.

MS. (2017). Microsoft Sharepoint. https://tinyurl.com/ojnyydd

Muijs, D. (2004). *Doing Quantitative Research in Education: With SPSS*. Sage Publications. ISBN: 978-0761943822

Mulder, J. B. F., & Johnson, J. (2016). The next step of IT project research in practice: The CHAOS university system. In *The 10th International Multi- Conference on Society, Cybernetics and Informatics: IMSCI 2016, July 5–8* Orlando, Florida.

Mutschler, B. (2005). Kosten-nutzen-analysen zur bewertung prozessorientierter informationssysteme. In *2. GI-Fachtagung IT-Controlling, Germany, St. Augustin* (pp. 82–92). In German.

Nakamoto, S. (2008). *Bitcoin: A peer-to-peer electronic cash system*. White paper, bitcoin.org. https://bitcoin.org/en/bitcoin-paper

Necco, C. R., Gordon, C. L., & Tsai, N. W. (1987). Systems-analysis and design—current practices. *MIS Quarterly, 11*(4), 461–476. https://doi.org/10.2307/248975

Negroponte, N. (1996). *Being digital*. Vintage Books. ISBN: 0-679-76290-6

Nelson, R. R., & Winter, S. G. (1982). *An evolutionary theory of economic change*. Belknap Press.

Nickerson, R. C., Varshney, U., & Muntermann, J. (2013). A method for taxonomy development and its application in information systems. *European Journal of Information Systems, 22*(3), 336–359.

Nielsen, J. (2012). How many test users in a usability study. *Nielsen Norman Group, 4*(6).

Normann, R. (2001). *Reframing business: When the map changes the landscape*. Wiley.

Nyman, J. (2016). What is the value of security? Contextualising the negative/positive debate. *Review of International Studies, 42*(5), 821–839. https://doi.org/10.1017/S0260210516000140

O'Kane, S. (2016). NASA's Mars rover is really good at laser-blasting rocks without human input—curiosity has been studying Mars on its own for a year. https://tinyurl.com/yaodb7p5

O'Shaughnessy, J., & O'Shaughnessy, N. (2009). The service-dominant perspective: A backward step? *European Journal of Marketing, 43*(5/6), 784–793.

Object Management Group. (2010). *Unified modeling language—superstructure*. Technical Report version 2.4.1, OMG.

Office of Government Commerce. (2007). *The introduction to the ITIL service lifecycle*. The Stationery Office (TSO).

OMG. (2007). *OMG unified modeling language (OMG UML), infrastructure, V2.1.2*. Technical report, The Object Management Group, Needham, Massachusetts. https://www.omg.org/spec/UML/2.1.2/Infrastructure/PDF

OMG. (2011). *Business process modeling notation, V2.0*. Technical Report OMG Document Number: Formal/2011-01-03, Object Management Group, Needham, Massachusetts.

OMG. (2015). *Value delivery modeling language—version 1.0*. Technical Report formal/2015-10-05, The Object Management Group, Needham, Massachusetts. http://www.omg.org/spec/VDML/1.0/

Op 't Land, M., Proper, H. A., Waage, M., Cloo, J., & Steghuis, C. (2008a). *Enterprise architecture—creating value by informed governance*. The Enterprise Engineering Series. Springer. ISBN: 978-3-540-85231-5. https://doi.org/10.1007/978-3-540-85232-2

Op 't Land, M., Proper, H. A., Waage, M., Cloo, J., & Steghuis, C. (2008b). *The results of enterprise architecting*, chapter 4. In [Op 't Land, M., Proper, H. A., Waage, M., Cloo, J., & Steghuis, C. (2008a). *Enterprise architecture—creating value by informed governance*. The Enterprise Engineering Series. Springer. ISBN: 978-3-540-85231-5. https://doi.org/10.1007/978-3-540-85232-2]. ISBN: 978-3-540-85231-5. https://doi.org/10.1007/978-3-540-85232-2

Oracle. (2017). Oracle enterprise repository. https://tinyurl.com/y2uxth6h

Orlikowski, W. J. (2000). Using technology and constituting structures: A practice lens for studying technology in organizations. *Organization Science, 11*(4), 404–428.

Pagani, M. (2013). Digital business strategy and value creation: Framing the dynamic cycle of control points. *MIS Quarterly, 37*(2), 617–632.

Parent, C., & Spaccapietra, S. (2000). Database integration: The key to data interoperability. In M. P. Papazoglou, S. Spaccapietra, & Z. Tari (Eds.), *Advances in object-oriented data modeling* (pp. 221–253). MIT Press. ISBN: 978-0262536783

Parker, G., Van Alstyne, M. W., & Jiang, X. (2017). Platform ecosystems: How developers invert the firm. *MIS Quarterly, 41*(1), 255–266.

Parsons, J., Iivari, J., & Hevner, A. R. (2005). Research in information systems analysis and design: Introduction to the special theme papers. *Communications of the Association for Information Systems, 16*(40). https://doi.org/10.17705/1CAIS.01640

Pavlou, P., & El Sawy, O. A. (2010). The 'third hand': It-enabled competitive advantage in turbulence through improvisational capabilities. *Information Systems Research, 21*(3), 443–471.

Payne, F., Storbacka, K., & Frow, P. (2008). Managing the co-creation of value. *Journal of the Academy of Marketing Science, 36*(1), 83–96. https://doi.org/10.1007/s11747-007-0070-0

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45–77.

Peppard, J., & Ward, J. (2016). *The strategic Management of information systems: Building a digital strategy* (4th ed.). Wiley.

Pereira, C. M.,& Sousa, P. (2005). Enterprise architecture: Business and it alignment. In *Proceedings of the 2005 ACM symposium on Applied computing* (pp. 1344–1345). ACM.

Pilkington, M. (2015). Blockchain technology: Principles and applications. In F. X. Olleros, Z. M., & E. Elgar (Eds.), *Research handbook on digital transformations*. SSRN. https://ssrn.com/abstract=2662660

Plataniotis, G., de Kinderen, S., & Proper, H. A. (2014a). Capturing design rationales in enterprise architecture: A case study. In U. Frank, P. Loucopoulos, O. Pastor, & I. Petrounias (Eds.), *Proceedings of the 7th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2014)*, volume 197 of Lecture notes in business information processing (pp. 133–147). Springer. ISBN: 978-3-662-45500-5. https://doi.org/10.1007/978-3-662-45501-2_10

Plataniotis, G., de Kinderen, S., & Proper, H. A. (2014b). EA anamnesis: An approach for decision making analysis in enterprise architecture. *International Journal of Information System Modeling and Design*, **5**(3), 75–95. ISSN: 1947-8186. https://doi.org/10.4018/ijismd.2014070104

Plataniotis, G., de Kinderen, S., Ma, Q., & Proper, H. A. (2015a). A conceptual model for compliance checking support of enterprise architecture decisions. In D. Aveiro, U. Frank, K.-J. Lin, & J. M. Tribolet (Eds.), *17th IEEE Conference on Business Informatics, CBI 2015, Lisbon, Portugal, July 13–16, 2015—Volume 1* (pp. 191–198). IEEE Computer Society. ISBN: 978-1-4673-7340-1. https://doi.org/10.1109/CBI.2015.46

Plataniotis, G., Ma, Q., Proper, H. A., & de Kinderen, S. (2015b). Traceability and modeling of requirements in enterprise architecture from a design rationale perspective. In C. Rolland, D. Anagnostopoulos, P. Loucopoulos, & C. Gonzalez-Perez (Eds.), *9th IEEE International Conference on Research Challenges in Information Science, RCIS 2015, Athens, Greece, May 13–15, 2015* (pp. 518–519). IEEE Computer Society. ISBN: 978-1-4673-6630-4. https://doi.org/10.1109/RCIS.2015.7128916

Plessius, H., van Steenbergen, M., & Slot, R. (2014). Perceived benefits from enterprise architecture. In *Eighth Mediterranean Conference on Information Systems, Verona*.

Poort, E., & van Vliet, H. (2011). Architecting as a risk- and cost management discipline. In *Ninth Working IEEE/IFIP Conference on Software Architecture* (pp. 2–11).

Prahalad, C. K., & Hamel, G. (1990). The core competence of the corporation. *Harvard Business Review, 68*(3), 79–91.

Prahalad, C. K., & Ramaswamy, V. (2000). Co-opting customer competence. *Harvard Business Review, 78*(1), 79–87.

Prahalad, C. K., & Ramaswamy, V. (2004). Co-creating unique value with customers. *Strategy & Leadership, 32*(3), 4–9. https://doi.org/10.1108/10878570410699249

Prat, N., Comyn-Wattiau, I., & Akoka, J. (2014). Artifact evaluation in information systems design-science research—a holistic view. In *19th Pacific Asia Conference on Information Systems (PACIS)* (pp. 23).

Prat, N., Comyn-Wattiau, I., & Akoka, J. (2015). A taxonomy of evaluation methods for information systems artifacts. *Journal of Management Information Systems, 32*(3), 229–267. https://doi.org/10.1080/07421222.2015.1099390

Priem, R. L. (2007). A consumer perspective on value creation. *Academy of Management Review, 32*(1), 219–235.

Priem, R. L., Butler, J. E., & S., L. (2013). Toward reimagining strategy research: Restrospection and prospection on the 2011 AMR decade award article. *Academy of Management Review, 38*(4), 471–489.

Proper, H. A. (1994). *A theory for conceptual modelling of evolving application domains*. PhD thesis, University of Nijmegen, Nijmegen, the Netherlands. ISBN: 9-090-06849-X. https://repository.ubn.ru.nl/handle/2066/83258

Proper, H. A. (2014). Enterprise architecture: Informed steering of enterprises in motion. In S. Hammoudi, J. Cordeiro, L. A. Maciaszek, & J. Filipe (Eds.), *Enterprise Information Systems—15th International Conference, ICEIS 2013, Angers, France, July 4-7, 2013, Revised Selected*

Papers, volume 190 of Lecture notes in business information processing (pp. 16–34). Springer. ISBN: 978-3-319-09491-5. https://doi.org/10.1007/978-3-319-09492-2_2

Proper, H. A. (2020). Digital enterprise modelling—opportunities and challenges. In B. Roelens, W. Laurier, G. Poels, & H. Weigand (Eds.), *Proceedings of 14th International Workshop on Value Modelling and Business Ontologies, Brussels, Belgium, January 16-17, 2020*, volume 2574 of CEUR Workshop Proceedings (pp. 33–40). CEUR-WS.org. http://ceur-ws.org/Vol-2574/short3.pdf

Proper, H. A. (2021). *On model-based coordination of change in organizations* (pp. 79–98). Springer. ISBN: 978-3-030-84655-8. https://doi.org/10.1007/978-3-030-84655-8_6

Proper, H. A., & Bjeković, M. (2019). Fundamental challenges in systems modelling. In H. C. Mayr, S. Rinderle-Ma, & S. Strecker (Eds.), *40 Years EMISA 2019, May 15–17, 2019, Evangelische Akademie Tutzing, Germany*, volume P-304 of Lecture notes in informatics (pp. 13–28). Gesellschaft für Informatik e.V. https://dl.gi.de/20.500.12116/33130

Proper, H. A., & Lankhorst, M. M. (2014). Enterprise architecture—towards essential sensemaking. *Enterprise Modelling and Information Systems Architectures, 9*(1), 5–21. https://doi.org/10.18417/emisa.9.1.1

Proper, H. A., & van Gils, B. (2019). Enterprise modelling in the age of digital transformation. *IFIP Select, 1*(2). http://ifip.org/select/select_1_2/proper%201_2.pdf

Proper, H. A., Bleeker, A. I., & Hoppenbrouwers, S. J. B. A. (2004). Object-role modeling as a domain modeling approach. In J. Grundspenkis & M. Kirikova (Eds.), *CAiSE'04 Workshops in Connection with The 16th Conference on Advanced Information Systems Engineering, Riga, Latvia, 7–11 June, 2004, Knowledge and Model Driven Information Systems Engineering for Networked Organisations, Proceedings, Vol. 1*, volume 3 (pp. 317–328). Faculty of Computer Science and Information Technology. ISBN: 9984-9767-1-8

Proper, H. A., Bjeković, M., Feltus, C., & Razo-Zapata, I. S. (2018a). On the development of a modelling framework for value co-creation. In J. Gordijn & H. A. Proper (Eds.), *Proceedings of the 12th International Workshop on Value Modeling and Business Ontologies, VMBO 2018, Amsterdam, the Netherlands, February 26th–27th, 2018*, volume 2239 of *CEUR Workshop Proceedings* (pp. 122–132). CEUR-WS.org. http://ceur-ws.org/Vol-2239/article_13.pdf

Proper, H. A., Bjeković, M., van Gils, B., & Hoppenbrouwers, S. J. B. A. (2018b). Towards grounded enterprise modelling. In C. Debruyne, H. Panetto, G. Weichhart, P. Bollen, I. Ciuciu, M.-E. Vidal, & R. Meersman (Eds.), *On the Move to Meaningful Internet Systems. OTM 2017 Workshops—Confederated International Workshops, EI2N, FBM, ICSP, Meta4eS, OTMA 2017 and ODBASE Posters 2017, Rhodes, Greece, October 23-28, 2017, Revised Selected Papers*, volume 10697 of Lecture notes in computer science (pp. 141–151). Springer. ISBN: 978-3-319-73804-8. https://doi.org/10.1007/978-3-319-73805-5_15

Proper, H. A., Winter, R., Aier, S., & de Kinderen, S., Eds. (2018c). *Architectural coordination of enterprise transformation*. The Enterprise Engineering Series. Springer. ISBN: 978-3-319-69583-9. https://doi.org/10.1007/978-3-319-69584-6

Proper, H. A., Guédria, W., & Sottet, J.-S. (2020). Enterprise modelling in the digital age. In V. Kulkarni, S. Reddy, T. Clark, & B. S. Barn (Eds.), *Advanced digital architectures for model-driven adaptive enterprises* (chapter 3, pp. 46–67). IGI Global. ISBN: 9781799801085. https://doi.org/10.4018/978-1-7998-0108-5.ch003

Rückle, H., & Behn, M. (2007). *Unternehmenserfolg mit Zielen: mit Checklisten, Leitfäden und Übungen*. Expert Verlag. In German.

Ralyté, J. (2004). Towards situational methods for information systems development: Engineering reusable method chunks. In *13th International Conference on Information Systems Development; Advances in Theory, Practice and Education* (pp. 271–282).

Ranjan, K. R., & Read, S. (2014). Value co-creation: Concept and measurement. *Journal of the Academy of Marking Science*. Published online 5 Aug 2014.

Razo-Zapata, I. S., Chew, E., & Proper, H. A. (2016). Visual modeling for value (co-)creation. In *Proceedings of the 10th International Workshop on Value Modelling and Business Ontologies (VMBO)*.

Razo-Zapata, I. S., Chew, E., & Proper, H. A. (2017). Towards VIVA: A visual language to model value co-creation. In D. Karagiannis (Ed.), *The 5th International Conference on Serviceology (ICServ 2017), Vienna, July 12–14, 2017*. http://www.serviceology.org

Razo-Zapata, I. S., Chew, E., & Proper, H. A. (2018). VIVA: A visual language to design value co-creation. In H. A. Proper, S. Strecker, & C. Huemer (Eds.), *20th IEEE Conference on Business Informatics, CBI 2018, Vienna, Austria, July 11–14, 2018, Volume 1—Research Papers* (pp. 20–29). IEEE Computer Society. ISBN: 978-1-5386-7016-3. https://doi.org/10.1109/CBI.2018.00012

Recker, J. R. (2009). Business process modeling: A comparative analysis. *Journal of the Association for Information Systems, 10*(4), 333–363.

Recker, J., Rosemann, M., Green, P., & Indulska, M. (2011). Do ontological deficiencies in modeling grammars matter? *MIS Quarterly, 35*(1), 57–79.

van Reijswoud, V. E., Mulder, J. B. F., & Dietz, J. L. G. (1999). Communicative action based business process and information modelling with DEMO. *The Information Systems Journal, 9*(2), 117–138.

Renkema, T., & Berghout, E. (1997). Methodologies for information system investment evaluation at the proposal stage: A comparative view. *Information and Software Technology, 39*(1), 1–13.

Reuver, M. d., Sørensen, C., & Basole, R. C. (2018). The digital platform: A research agenda. *Journal of Information Technology, 33*(2), 123–135. https://doi.org/10.1057/s41265-016-0033-3

Rochet, J. C., & Tirole, J. (2003). Platform competition in two-sided markets. *Journal of the European Economic Association, 1*(4), 990–1029. https://doi.org/10.1162/154247603322493212

Roeleven, S., & Broer, J. (2019). Why two thirds of enterprise architecture projects fail. https://tinyurl.com/y4y4yn4w

Romme, A. G. L., & Endenburg, G. (2006). Construction principles and design rules in the case of circular design. *Organization Science, 17*, 287–297.

Ross, J. W., Beath, C. M., & Sebastian, I. (2015). Why Nordstrom's digital strategy works (and yours probably doesn't). *Harvard Business Review, 1*.

Ross, J. W., Sebastian, I. M., Beath, C., Mocker, M., Fondstad, N. O., & Moloney, K. G. (2016). Designing and executing digital strategies. In *Proceedings of the International Conference on Information Systems (ICIS)* (pp. 1–16).

Roth, S., Zec, M., & Matthes, F. (2014). *Enterprise Architecture Visualization Tool Survey*. Technical report, sebis, Technische Universität München.

Rothrock, L., & Yin, J. (2008). Integrating compensatory and noncompensatory decision-making strategies in dynamic task environments. In T. Kugler, J. C. Smith, T. Connolly, & Y. J. Son (Eds.), *Decision Modeling and Behavior in Complex and Uncertain Environments*, volume 21 of Springer optimization and its applications (pp. 125–141). Springer. ISBN: 978-0-387-77130-4. https://doi.org/10.1007/978-0-387-77131-1_6

Rouse, W. B. (2005a). A theory of enterprise transformation. *Systems Engineering, 8*(4), 279–295. ISSN: 1520-6858. https://doi.org/10.1002/sys.20035

Rouse, W. B. (2005b). Enterprises as systems: Essential challenges and approaches to transformation. *Systems Engineering, 8*(2), 138–150.

Rouse, W. B., & Baba, M. (2006). Enterprise transformation—fundamental enterprise changes begin by looking at the challenges from technical, behavioral, and social perspectives. *Communications of the ACM, 49*(7), 67–72.

Rouwette, E. A. J. A., & Vennix, J. A. M. (2006). System dynamics and organizational interventions. *Systems Research and Behavioral Science, 23*, 451–466. https://doi.org/10.1002/sres.772

Rowbottom, R., & Billis, D. (1987). *Organisational design: The work-levels approach*. Gower.

Saaty, T. (1996). *Multicriteria decision making: The analytic hierarchy process*. RWS Publications.

Sandkuhl, K., Stirna, J., Persson, A., & Wißotzki, M. (2014). *Enterprise modeling: Tackling business challenges with the 4EM method*. Springer. ISBN: 978-3-662-43724-7S

Sarasvathy, S., Dew, N., Read, S., & Wiltbank, R. (2008). Designing organizations that design environments: Lessons from entrepreneurial expertise. *Organization Studies, 29*(3), 331–350.

Sarker, S., Sahaym, A., & Bjorn-Andersen, N. (2012). Exploring value cocreation in relationships between an Erp vendor and its partners: A revelatory case study. *MIS Quarterly, 36*(1), 317–338. https://doi.org/10.2307/41410419

Scheer, A.-W. (1988). *Computer integrated manufacturing: CIM*. Springer. ISBN: 3-540-19191-7

Scheer, A.-W. (1992). *Architecture of integrated information systems: Foundations of enterprise modelling*. Springer. ISBN: 354055131X

Scheer, A.-W. (2000). *ARIS—business process modeling*. Springer. ISBN: 3-540-65835-1

Scheer, A. W. & Nüttgens, M. (2000). ARIS architecture and reference models for business process management. In W. M. P. van der Aalst, J. Desel., & A. Oberweis (Eds.), *Business process management*, number 1806 in Lecture Notes in Computer Science (pp. 376–389). Springer. https://doi.org/10.1007/3-540-45594-9_24

Scherer, A., Wunderlich, N. V., & von Wangenheim, F. (2015). The value of self-service: Long-term effects of technology-based self-service usage on customer retention. *MIS Quarterly, 39*(1), 177–200.

Schiller, J. H. (2003). *Mobile communications*. Addison–Wesley.

Schultz, M., Hatch, M. J., & Larsen, M. H. (2000). Scaling the tower of babel: Relational differences between Identity, image and culture in organizations. In *The expressive organization*. Oxford University Press.

Scott, S. G. & Lane, V. R. (2000). A stakeholder approach to organizational identity. *Academy of Management Review, 25*(1), 43–62.

Seidl, D. (2005). *Organisational identity and self-transformation: An autopoietic perspective*. ISBN: 0754644588

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research. *MIS Quarterly, 35*(1), 37–56.

Serasa Experian. (2017). Serasa experian. https://www.serasaexperian.com.br

Shang, S., & Seddon, P. B. (2002). Assessing and managing the benefits of enterprise systems: The business manager's perspective. *Information Systems Journal, 12*(4), 271–299. https://doi.org/10.1046/j.1365-2575.2002.00132.x

Sharman, R., Kishore, R., & Ramesh, R. (2004). Computational ontologies and information systems II: Formal specification. *Communications of the Association for Information Systems, 14*(184–205).

Siau, K., & Rossi, M. (2011). Evaluation techniques for systems analysis and design modelling methods—a review and comparative analysis. *Information Systems Journal, 21*(3), 249–268.

Sidorova, A., Evangelopoulos, N., Valacich, J. S., & Ramakrishnan, T. (2008). Uncovering the intellectual core of the information systems discipline. *MIS Quarterly, 32*(2), 467–483.

Siering, M., Clapham, B., Engel, O., & Gomber, P. (2017). A taxonomy of financial market manipulations: Establishing trust and market integrity in the financialized economy through automated fraud detection. *Journal of Information Technology, 32*(2), 251–269. https://doi.org/10.1057/s41265-016-0029-z

Silva, N., Ferreira, F., Sousa, P., & Mira da Silva, M. (2016). Automating the migration of enterprise architecture models. *International Journal of Information System Modeling and Design (IJISMD), 7*(2), 72–90.

Simon, H. A. (1969). *Sciences of the artificial* (2nd ed.). MIT Press.

Simon, H. A. (1997). *Administrative behavior: A study of decision making processes in administrative organization*. The Free Press.

Simsion, G., & Witt, G. (2004). *Data modeling essentials*. Morgan Kaufmann.

Smith, A. (1963). *An inquiry into the nature and causes of the wealth of nations* (Vol. 4). O. Zeller.

Smits, M., & Hulstijn, J. (2020). Blockchain applications and institutional trust. *Frontiers in Blockchain, 3*, 5. www.frontiers.org, 3(5). https://doi.org/10.3389/fbloc.2020.00005

Snoeck, M. (2014). *Enterprise information systems engineering—the MERODE approach*. The Enterprise Engineering Series. Springer. ISBN: 9783319101446

Snow, C. C., & Hrebiniak, L. G. (1980). Strategy, distinctive competence, and organizational performance. *Administrative Science Quarterly, 25*(2), 317–336.

Solow, R. (1987). We'd better watch out. *New York Times Book Review*, 12(July).

Sonar. (2017). SonarSource. https://www.sonarsource.com

Sonnenberg, C., & vom Brocke, J. (2012). Evaluations in the science of the artificial—reconsidering the build-evaluate pattern in design science research. In K. Peffers, M. Rothenberger, & B. Kuechler (Eds.), *Proceedings of the 7th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2012), Las Vegas, Nevada*, volume 7286 of Lecture notes in computer science (pp. 381–397). Springer. ISBN: 978-3-642-29862-2. https://doi.org/10.1007/978-3-642-29863-9_28

Sousa, P., & Vasconcelos, A. (2014). Enterprise architecture at the portuguese government. In *EEWC 2014, Industrial Track*. https://tinyurl.com/y68f2btu

Sousa, P., Lima, J., Sampaio, A., & Pereira, C. (2009). An approach for creating and managing enterprise blueprints: A case for IT blueprints. In A. Albani, J. Barjis, & J. L. G. Dietz (Eds.), *Advances in enterprise engineering III*, volume 34 of Lecture notes in business information processing (pp. 70–84). Springer. ISBN: 978-3-642-01914-2. https://doi.org/10.1007/978-3-642-01915-9_6

Sousa, P., Gabriel, R., Tadao, G., Carvalho, R., Sousa, P. M., & Sampaio, A. (2011). Enterprise transformation: The serasa experian case. In A. F. Harmsen, K. Grahlmann, & H. A. Proper (Eds.), *Practice-Driven Research on Enterprise Transformation—Third Working Conference, PRET 2011, Luxembourg-Kirchberg, Luxembourg, September 6, 2011. Proceedings*, volume 89 of Lecture notes in business information processing (pp. 134–145). Springer. ISBN: 978-3-642-23387-6. https://doi.org/10.1007/978-3-642-23388-3_7

Sousa, P. M., Martins, R., & Sampaio, A. (2012). A clarification of the application concept: The Caixa Geral de Depósitos Case. In H. A. Proper, A. F. Harmsen, K. Gaaloul, & S. Wrycza (Eds.), *Practice-Driven Research on Enterprise Transformation—4th Working Conference, PRET 2012, Gdańsk, Poland, June 27, 2012. Proceedings*, volume 120 of Lecture notes in business information processing (pp. 1–17). Springer. ISBN: 978-3-642-31133-8. https://doi.org/10.1007/978-3-642-31134-5_1

Sousa, P., Sampaio, A., & Leal, R. (2014). A case for a living enterprise architecture in a private bank. In W. A. Molnar, H. A. Proper, J. Zdravkovic, P. Loucopoulos, O. Pastor, & S. Kinderen de (Eds.), *Proceedings of the 8th Transformation & Engineering of Enterprises workshop (TEE 2014), held in conjunction with the 16th IEEE Conference on Business Informatics (CBI 2014), Geneva, Switzerland*: CEUR-WS.org. http://ceur-ws.org/Vol-1182/

Spicer, A. (2006). Beyond the convergence–divergence debate: The role of spatial scales in transforming organizational logic. *Organization Studies, 27*(10), 1467–1483.

Statista. (2017). Most Valuable Companies in the World 2017. https://tinyurl.com/statista2017companyvalue

Stelzer, D. (2009). Enterprise architecture principles: Literature review and research directions. In *Proceedigs of the Workshop on Trends in Enterprise Architecture Research (TEAR 2009)* (pp. 21–36).

Stoop, J., Staffhorst, S., Bekker, R., & Hobma, T. (2016). *Business transformation framework—to get from strategy to execution*. Best Practice. Van Haren. ISBN: 9789401800266

Straub, D., Rai, A., & Klein, R. (2004). Measuring firm performance at the network level: A nomology of the business impact of digital supply networks. *Journal of Management Information Systems, 21*(1), 83–114.

Strube, G. (2001). Cognitive modeling: Research logic in cognitive science. In *International Encyclopedia of Social and Behavioral Sciences* (Vol. 55). https://doi.org/10.1016/B0-08-043076-7/00588-X

Swan, M. (2012). Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0. *Journal of Sensor and Actuator Networks, 1*(3), 217–253.

Swan, M. (2015). *Blueprint for a new economy*. O'Reilly.

Tan, B., Pan, S. L., Lu, X. H., & Huang, L. H. (2015). The role of is capabilities in the development of multi-sided platforms: The digital ecosystem strategy of Alibaba.Com. *Journal of the Association for Information Systems, 16*(4), 248–280.

Tapscott, D. (1996). *Digital economy—promise and peril in the age of networked intelligence*. McGraw–Hill. ISBN: 0-070-63342-8

Tapscott, D., Ticoll, D., & A., L. (2000). *Digital capital: Harnessing the power of business webs*. Harvard Business Press. ISBN: 978-1-578-51193-8

Taylor, F. W. (1911). *Principles of scientific management*. Harper & Row.

Taylor, J. R. (1982). Office communications: Reshaping our society? *Computer Communications, 5*(4), 176–180. https://doi.org/10.1016/0140-3664(82)90112-8

Teece, D. J. (2007). Explicating dynamic capabilities: The nature and of (sustainable) enterprise performance. *Strategic Management Journal, 4*(28), 1319–1350.

Teece, D. J., Pisano, G., & Shuen, A. (1997). Dynamic capabilities and strategic management. *Strategic Management Journal, 18*(7), 509–533.

The Open Group. (2011). *TOGAF version 9.1* (10th ed.). Van Haren Publishing. ISBN: 978-9-087-53679-4

The Standish Group International. (1994). *CHAOS*. Technical report, West Yarmouth, Massachusetts.

The Standish Group International. (2016). *CHAOS*. Technical report, West Yarmouth, Massachusetts.

Theoharidou, M., Mylonas, A., & Gritzalis, D. (2012). A risk assessment method for smartphones. In *IFIP International Information Security Conference* (pp. 443–456). Springer. https://doi.org/10.1007/978-3-642-30436-1_36

Thomas, L. D. W., Autio, E., & Gann, D. M. (2014). Architectural leverage: Putting platforms in context. *Academy of Management Perspectives, 28*(2), 198–219.

Tilson, D., Lyytinen, K., & Sørensen, C. (2010). Research commentary—digital infrastructures: The missing is research agenda. *Information Systems Research, 21*(4), 748–759.

Tiwana, A. (2015). Evolutionary competition in platform ecosystems. *Information Systems Research, 26*(2), 266–281.

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research, 21*(4), 675–687.

Toppen, R., Smits, M., & Ribbers, P. (1998). Financial securities transactions: A study of logistic process performance improvements. *Journal of Strategic Information Systems, 7*, 199–216.

Torbert, W. R. (2004). *Action inquiry: The secret of timely and transforming leadership*. Berrett-Koehler Publishers.

Tribolet, J. M. (2005). Organizações, Pessoas, Processos e Conhecimento: Da Reificação do Ser Humano como Componente do Conhecimento à Consciência de Si Organizacional. In L. Amaral (Ed.), *Sistemas de Informação Organizacionais*, Ediçôs Sîlabo.

Tribolet, J. M. (2014). An engineering approach to natural enterprise dynamics—from top-down purposeful systemic steering to bottom-up adaptive guidance control. ICEIS 2014 keynote (FIX ME). http://www.iceis.org/KeynoteSpeakers.aspx?y=2014%5C#4

Tribolet, J. M., Pombinho, J., & Aveiro, D. (2014a). Organizational self awareness: A matter of value. In *Organization design and engineering: Co-existence, co-operation or integration* (pp. 146–176). Palgrave Macmillan.

Tribolet, J. M., Sousa, P., & Caetano, A. (2014b). The role of enterprise governance and cartography in enterprise engineering. *Enterprise Modelling and Information Systems Architectures, 9*(1), 38–49.

Tulinayo, F. P., van Bommel, P., & Proper, H. A. (2013). Enhancing the system dynamics modeling process with a domain modeling method. *International Journal of Cooperative Information Systems, 22*(2). https://doi.org/10.1142/S0218843013500111

Umar, A. (2005). IT infrastructure to enable next generation enterprises. *Information Systems Frontiers, 7*(3), 217–256. https://doi.org/10.1007/s10796-005-2768-1

Unhelkar, B. (2009). Technical, methodological and social perspectives. In *Handbook of research in mobile business* (2nd ed.). IGI Global.

Varela, F., Thompson, E., & Rosch, E. (1991). *The embodied mind*. MIT Press.

Vargo, S. L., & Lusch, R. F. (2004). Evolving to a new dominant logic for marketing. *Journal of Marketing, 68*, 1–17.

Vargo, S. L., & Lusch, R. F. (2008). Service-dominant logic: Continuing the evolution. *Journal of the Academy of marketing Science, 36*, 1–10.

Vargo, S. L., & Lusch, R. F. (2011). It's all B2B ... and beyond: Toward a systems perspective of the market. *Industrial Marketing Management, 40*, 181–187.

Vargo, S. L., & Lusch, R. F. (2016). Institutions and axioms: An extension and update of service-dominant logic. *Journal of the Academy of Marketing Science, 44*(1).

Vargo, S. L., & Lusch, R. F. (2017). Service-dominant logic 2025. *International Journal of Research in Marketing, 34*(1), 46–67. https://doi.org/10.1016/j.ijresmar.2016.11.001

Vargo, S. L., Lusch, R. F., & Akaka, M. A. (2010). Advancing service science with service-dominant logic. In P. P. Maglio, J. C. Kieliszewsk, & J. C. Spohrer (Eds.), *Handbook of service science* (pp. 133–156). Springer.

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS Quarterly, 27*(3), 425–478.

Verdonck, M., Gailly, F., de Cesare, S., & Poels, G. (2015). Ontology-driven conceptual modeling: A systematic literature mapping and review. *Applied Ontology, 10*(3-4), 197–227.

Versendaal, J., & Högler, T. (2017). *On the validation of the initial step of an upfront system implementation evaluation framework: The Fire Brigade Case*. Technical Report, HBO-Kennisbank.

Versendaal, J., Högler, T., & Batenburg, R. (2016). *On the validation of an evaluation framework: The case of Nursemapp*. Technical Report, HBO-Kennisbank.

Vicini, S., Alberti, F., Notario, N., Crespo, A., Pastoriza, J. R. T., & Sanna, A. (2016). Co-creating security-and-privacy-by-design systems. In *2016 11th International Conference on Availability, Reliability and Security (ARES)* (pp. 768–775).IEEE Computer Society. https://doi.org/10.1109/ARES.2016.74

Villar, E. B., & Miralles, F. (2014). Beyond resources and dynamic capabilities during disaster response: Response organizations in turbulent and networked environment: The case of a disaster response organization's simulated response to chemical spill disaster. In *International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*. IEEE.

van Vrekhem, F. (2015). *The disruptive competence: The journey to a sustainable business, from matter to meaning*. Compact Publishing.

Walls, J. G., Widmeyer, G. R., & El Sawy, O. A. (1992). Building an information system design theory for vigilant EIS. *Information Systems Research, 3*(1), 36–59.

Wand, Y., & Weber, R. (1993). On the ontological expressiveness of information systems analysis and design grammars. *Information Systems Journal, 3*(4), 217–237. https://doi.org/10.1111/j.1365-2575.1993.tb00127.x

Wand, Y., & Weber, R. (2002). Research commentary: Information systems and conceptual modeling—a research agenda. *Information Systems Research, 13*(4), 363–376. https://doi.org/10.1287/isre.13.4.363.69

Wang, S., & Wang, H. (2012). *Information systems analysis and design*. Universal Publishers.

Wareham, J., Fox, P. B., & Giner, J. L. C. (2014). Technology ecosystem governance. *Organization Science, 25*(4), 1195–1215.

Weber, R. H., & Weber, R. (2010). *Internet of Things*. Springer. ISBN: 9783642117091

Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future: Writing a literature review. *MIS Quarterly, 26*(2), xiii–xxiii.

Wegmann, A. (2003). On the systemic enterprise architecture methodology (SEAM). In *International Conference on Enterprise Information Systems (ICEIS 2003)*.

Weick, K. E. (1979). *The social psychology of organizing*. Addison–Wesley.

Weick, K. E. (1995). *Sensemaking in organizations*. Sage.

Weick, K. E. (1998). Introductory essay—improvisation as a mindset for organizational analysis. *Organization Science, 9*(5), 543–555.

Weigand, H. (2009). Value encounters—modeling and analyzing co-creation of value. In C. Godart, N. Gronau, S. Sharma, & G. Canals (Eds.), *Software services for e-business and e-society* (pp. 51–64). Springer. ISBN: 978-3-642-04280-5. https://doi.org/10.1007/978-3-642-04280-5_5

Westerman, G., Bonnet, D., & McAfee, A. (2014). *Leading digital—turning technology into business transformation*. Harvard Business Review Press. ISBN: 978-62527-247-8

Wikipedia. (2019a). Cartography. https://en.wikipedia.org/wiki/Cartography

Wikipedia. (2019b). Nyquist rate. https://en.wikipedia.org/wiki/Nyquist_rate

Wikipedia. (2019c). Systems theory. https://en.wikipedia.org/wiki/Systems_theory

Wilden, R., Akaka, M. A., Karpen, I. O., & Hohberger, J. (2017). The evolution and prospects of service-dominant logic: An investigation of past, present, and future research. *Journal of Service Research, 20*(4), 345–361. https://doi.org/10.1177/1094670517715121

Wille, D., Holthusen, S., Schulze, S., & Schaefer, I. (2013). Interface variability in family model mining. In *Proceedings of the 17th International Software Product Line Conference Co-Located Workshops*, SPLC '13 Workshops (pp. 44–51). Association for Computing Machinery. ISBN: 9781450323253. https://doi.org/10.1145/2499777.2500708

Williams, G., Gunn, D., Roma, E., & Bansal, B. (2016). *Distributed ledgers in payments: Beyond the Bitcoin hype*. Technical report, Bain & Company. https://www.bain.com/insights/distributed-ledgers-in-payments-beyond-bitcoin-hype/

Winter, S. G. (2003). Understanding dynamic capabilities. *Strategic Management Journal, 24*, 991–995.

Winter, R. (2014). Architectural thinking. *Business & Information Systems Engineering, 6*(6), 361–364.

Winter, R., & Aier, S. (2011). How are enterprise architecture design principles used? In *The 6h Workshop on Trends in Enterprise Architecture Research (TEAR 2011) at the 15th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2011), Helsinki, Finland* (pp. 314–321). IEEE Computer Society.

World Economic Forum. (2016). The future of the financial industry. https://www3.weforum.org/docs/WEF_The_future_of_financial_infrastructure.pdf

van't Wout, J., Waage, M., Hartman, H., Stahlecker, M., & Hofman, A. (2010). The integrated architecture framework explained: Why, what, how. Springer. ISBN: 978-3-642-11517-2

Wright, A., & De Filippi, P. (2015). Decentralized blockchain technology and the rise of lex cryptographia. SSRN. https://doi.org/10.2139/ssrn.2580664

Yin, R. K. (2009). *Case study research—design and methods* (4th ed.). Sage Publications. ISBN: 978-1-412-96099-1

Yin, R. K. (2013). *Case study research: Design and methods (applied social research methods)*. Sage Publications.

Yoo, Y., Boland, R. J., & Lyytinen, K. (2006). From organization design to organization designing. *Organization Science, 17*(2), 215–229.

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research, 21*(4), 724–735. https://doi.org/10.1287/isre.1100.0322

Yoo, Y., Boland, R. J., Lyytinen, K., & Majchrzak, A. (2012). Organizing for innovation in the digitized world. *Organization Science, 23*(5), 1398–1408. https://doi.org/10.1287/orsc.1120.0771

Yu, E. S. K., & Mylopoulos, J. (1994). Using goals, rules, and methods to support reasoning in business process reengineering. *Intelligent Systems in Accounting, Finance and Management, 5*, 1–13. https://doi.org/10.1002/(SICI)1099-1174(199603)5:1<1::AID-ISAF99>3.0.CO;2-C

Zacarias, M., Magalhães, R., Caetano, A., Pinto, H. S., & Tribolet, J. M. (2007). Towards organizational self-awareness: An architecture and ontology framework. In P. Rittgen (Ed.), *Handbook of ontologies for business interaction*. IRM Press.

Zacarias, M., Pinto, H. S., Magalhães, R., & Tribolet, J. M. (2010). A context-aware and agent-centric perspective for the alignment between individuals and organizations. *Information Systems, 35*(4), 441–466.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal, 26*(3), 276–292.

Zachman, J. A. (1997). Enterprise architecture: The issue of the century. *Database Programming and Design, 10*(3), 44–53.

Zahn, E., Schmid, U., & Dillerup, R. (1999). Investitionsentscheidungen in flexible Fertigungssysteme. In H. Albach (Ed.), *Innovation und Investition* (pp. 43–67). Gabler. In German.

Zammuto, R. F., Griffith, T. L., Majchrzak, A., Dougherty, D. J., & Faraj, S. (2007). Information technology and the changing fabric of organization. *Organization Science, 18*(5), 749–762.

Zeithaml, V. A. (1988). Consumer perceptions of price, quality, and value: A means-end model and synthesis of evidence. *The Journal of Marketing, 2–22*. https://doi.org/10.1177/002224298805200302

Zivkovic, S., Kühn, H., & Karagiannis, D. (2007). Facilitate modelling using method integration: An approach using mappings and integration rules. In *Proceedings of the 5th European Conference on Information Systems (ECIS 2007), St. Gallen, Switzerland* (pp. 2038–2049). AIS.