Henry Linger · Julie Fisher
Andrew Barnden · Chris Barry
Michael Lang · Christoph Schneider
Editors

Building Sustainable Information Systems

Proceedings of the 2012 International Conference on Information Systems Development

Springer

Building Sustainable Information Systems

Henry Linger • Julie Fisher
Andrew Barnden • Chris Barry
Michael Lang • Christoph Schneider
Editors

Building Sustainable Information Systems

Proceedings of the 2012 International Conference on Information Systems Development

Springer

Editors
Henry Linger
Faculty of Information Technology
Monash University
Caulfield East, VIC, Australia

Julie Fisher
Faculty of Information Technology
Monash University
Caulfield East, VIC, Australia

Andrew Barnden
Centre for ICT
Box Hill Institute
Box Hill, VIC, Australia

Chris Barry
National University of Ireland
Galway, Ireland

Michael Lang
National University of Ireland
Galway, Ireland

Christoph Schneider
City University of Hong Kong
Kowloon
Hong Kong SAR

ISBN 978-1-4614-7539-2 ISBN 978-1-4614-7540-8 (eBook)
DOI 10.1007/978-1-4614-7540-8
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2013952729

Printed on acid-free paper

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

Preface

Information Systems (IS) as a discipline draws on diverse domains including, technology, organisational theory, management and social science. The field is recognized as very broad and encompassing many themes and areas. However, the development of technologically based artefacts, or information systems development (ISD), in the broadest sense, is a central concern of the discipline. Significantly, IS impacts on the organisational and societal contexts in the way the IS is deployed and consequently this impact presents evolving challenges for ISD.

Today, there is universal agreement that human behaviour has led to significant degradation of the natural environment on a global scale. This is now one of the most critical challenges confronting the world. ICT is widely recognised as one means to address this challenge, and IS can make a significant contribution as it combines the social with the technical aspects of systems. Thus the impact of ISD needs to be considered and evaluated not only in terms of its impact on the work environment but also on its effect in the broader natural and built environments as well as its social impact.

Sustainable IS is an emerging theme in academic research and industry practice in response to a variety of factors. These include recognition of the role ICT and IS can play in mitigation and adaptation strategies to address global environmental degradation and, paradoxically, how to deal with the direct contribution of ICT to that degradation, within the context of embryonic regulatory environments being enacted globally.

The ISD conference is dedicated to IS development and compliments the international network of general IS conferences (ICIS, ECIS, AMCIS, PACIS, ACIS). The development of information systems has paralleled technological developments and the deployment of those technologies in all areas of society, including government, industry, community and in the home. As a practice-based discipline, ISD has always promoted a close interaction between theory and practice that has set an agenda that is focused on the integration of people, processes and technology within a specific context.

This is the second occasion that Monash University has hosted the ISD conference. On the first occasion in 2003, the Faculty of Information Technology provided funding to invite to Melbourne members of the first Polish-Scandinavian ISD Seminar that was the forerunner of the International Conference on Information Systems Development, the ISD conference. This year a member of the original ISD Seminar, Professor Karlheinz Kautz, was a keynote speaker.

In this volume we bring together current research around the development of Sustainable IS, expressed in the theme of the 21st ISD Conference, "Building Sustainable Information Systems". The conference, held at the Monash University Prato Centre, in Tuscany, Italy, included diverse research that examined ISD from the perspective of sustainability. Keynote addresses were an important component as they reinforced and interpreted the conference theme. The two keynotes addresses were presented by Professor Doug Vogel, who spoke on Sustaining IS as a Discipline, and Professor Karlheinz Kautz, who addressed the issue of Managing Sustainable Information Systems Development—Culture, Balance and Mindfulness.

Sustainable IS, or "Green IS", is a catch-all term used to describe the development, construction, management, use and disposal of IS, and the ICT that underpins it, in a way that minimizes its impact on the environment. As a result, the term has many different meanings as was reflected in the conference tracks that included:

- The changing landscape of information systems: properties of a new IS ecology
- Methodologies for design within complex environments
- Green IS—information systems for sustainability
- Model-driven engineering in ISD
- Sustainable ISD project management
- Sustainable knowledge management

All submitted papers were double-blind reviewed by at least two reviewers, and the quality and fairness of the reviews attest to the quality of conference and the accepted papers. As is usual for ISD, conference delegates came from many countries and institutions and this contributed to the particular social aspect of the ISD conference that is its hallmark; collegiality and friendliness.

The success of any conference is dependent on many factors. First, we would like to extend our gratitude to the work of the track chairs, all the authors, the reviewers and delegates who attended the conference. Second, we would like to acknowledge and thank the work of those behind the scenes. In particular we thank Sanaz Bayati who worked tirelessly on all aspects of the organisation of the conference. We would also like to thank the staff of the Monash Prato Centre for their support in ensuring the smooth running of the conference. Last, we acknowledge the role of the International Steering Committee for their guidance and moral support.

Caulfield East, VIC, Australia Henry Linger
Caulfield East, VIC, Australia Julie Fisher
Box Hill, VIC, Australia Andrew Barnden

Conference Organisation

Conference Chairs

Henry Linger	Monash University, Australia
Julie Fisher	Monash University, Australia
Andrew Barnden	Box Hill Institute, Australia
Organising Chair	
Sanaz Bayati	Monash University, Australia

International Steering Committee

Henry Linger	Monash University, Australia
Chris Barry	National University of Ireland Galway, Ireland
Michael Lang	National University of Ireland Galway, Ireland
Christoph Schneider	City University of Hong Kong, Hong Kong

Track Chairs

Cloud Computing	
Edward Curry	National University of Ireland Galway, Ireland
Brian Donnellan	National University of Maynooth, Ireland
Philip A. DesAutels	Luleå University of Technology, Sweden
Methodologies for Design within Complex Environments	
Igor Hawryszkiewycz	University of Technology Sydney, Australia
Ghassan Beydoun	University of Wollongong, Australia
Corina Raduescu	The University of Sydney, Australia
Green IS—Information Systems for Sustainability	
Helen Hasan	University of Wollongong, Australia
Chadi Aoun	University of Technology Sydney, Australia
Lee Stewart	Fujitsu Australia Limited, Australia
Model-Driven Engineering in ISD	
Maria José Escalona	The University of Seville, Spain
Isidro Ramos Savater	Universitat Politècnica de València (UPV), Spain
Gustavo Aragón Serrano	Everis Group
Sustainable ISD Project Management	
Jill Owen	University of New South Wales, Australia
Sustainable Knowledge Management	
Frada Burstein	Monash University, Australia
Luba Torlina	Deakin University, Australia
Suzanne Zyngier	Latrobe University, Australia
Amrik Sohal	Monash University, Australia
General Track	
Andrew Barnden	Box Hill Institute of Technology, Australia

(continued)

(continued)

Conference Organisation	
Reviewers	
Abdulla Al-Ghamdi	Monash University, Australia
Naief Azyabi	Monash University, Australia
Chris Barry	National University of Ireland Galway, Ireland
Janis Barzdins	University of Latvia, Latvia
Juris Borzovs	University of Latvia, Latvia
Bob Brown	University of Wollongong, Australia
Alena Buchalcevova	University of Economics, Prague
Michael Cahalane	University of New South Wales, Australia
Marian Carcary	Innovation value Institute, Ireland
Raul Carlson	Viktoria Institute, Australia
François Charoy	University Henri Poincaré Nancy 1 - LORIA, France
Andy Connor	Auckland University of Technology, New Zealand
James Connor	University of New South Wales, Australia
Gerard Conway	University College Hospital, UK
Vanessa Cooper	RMIT University, Australia
Gabriel J. Costello	Galway-Mayo Institute of Technology, Ireland
Darren Dalcher	Middlesex University, UK
Linda Dawson	University of Wollongong, Australia
Antonio Diaz Andrade	Auckland University of Technology, New Zealand
Eileen Doherty	University of Pennsylvania, United States
Maria Jose Escalona	University of Seville, Spain
Piero Fraternali	Politecnico di Milano, Italy
Odd Fredriksson	Karlstad University, Sweden
Mark Freeman	University of Wollongong, Australia
Brent Furneaux	Maastricht University, Netherlands
Francisco Garcia Sanchez	University of Murcia, Spain
David Grant	Monash University, Australia
Remigijus Gustas	Karlstad University, Sweden
Pari Haghighi	Monash University, Australia
Chihab Hanachi	University Toulouse, France
Helen Hasan	University of Wollongong, Australia
Souleiman Hasan	National University of Ireland Galway, Ireland
Igor Hawryszkiewycz	University of Technology, Sydney, Australia
Alan Hevner	University of South Florida, United States
Jens Holgersson	University of Borås, Sweden
Grant Howard	University of South Africa, South Africa
Marcel Korte	Murdoch University, Australia
Aneesh Krishna	Curtin University, Australia, Australia
Gerard Kybryk	Cité des Sciences et de l'Industrie, France
Gabriele Lakomski	University of Melbourne, Australia
Michael Lang	National University of Ireland Galway, Ireland
Kevin Lee	Murdoch University, Australia
Sharman Lichtenstein	Deakin University, Australia
Felix Lizama	Monash University, Australia

(continued)

(continued)

Conference Organisation	
Antonio Lopez Lorca	University of Wollongong, Australia
Nikolaos Loutas	National University of Ireland Galway, Ireland
Graham Low	University of New South Wales, Australia
Kathryn Lynch	University of the Sunshine Coast, Australia
Leszek Maciaszek	Wrocław University of Economics, Poland
Jose-Norberto Mazon	Universidad de Alicante, Spain
Vanessa McDermott	University of New South Wales, Australia
Tim Miller	University of Melbourne, Australia
Simon Milton	University of Melbourne, Australia
Robert Moreton	University of Wolverhampton, UK
Kumaran Murugan	UNSW, Australia
Jan Newmarch	Box Hill Institute and Monash University, Australia
Bang Nguyen	Monash University, Australia
Anders Nilsson	Karlstad University, Sweden
Ovidiu Noran	Griffith University, Australia
Stacie Petter	University of Nebraska at Omaha, United States
Hamid Pousti	Monash University, Australia
Asif Qumer Gill	University of Sydney, Australia
Kai Riemer	University of Sydney, Australia
Margaret Ross	Southampton Solent University, UK
Gustavo Rossi	LIFIA-F. Informatica, UNLP, Argentina
Narcyz Roztocki	SUNY at New Paltz, United States
Christoph Schneider	City University of Hong Kong, Hong Kong
Arthur Shelley	RMIT, Australia
Stephen Smith	Macquarie University, Australia
Pamela Spink	Monash University, Australia
Susanne Strahringer	TU Dresden, Germany
Barney Tan	University of Sydney, Australia
Holly Tootell	University of Wollongong, Australia
Jesus Torres	Universidad de Sevilla, Spain
Umair Ul Hassan	National University of Ireland Galway, Ireland
Rosemary Van Der Meer	University of Wollongong, Australia
Frederik Von Briel	City University of Hong Kong, Hong Kong
Richard Watson	Sypaq Systems, Australia
Christine Welch	Portsmouth Business School, UK
Robert Winter	University of St. Gallen, Switzerland
Dongming Xu	The University of Queensland, Australia

Contents

Managing Sustainable Information Systems Development: Complexity, Culture, Balance and Mindfulness 1
Karlheinz Kautz

The Role of Trust in the Initial Adoption of Identity Management Systems 25
Ali Alkhalifah and John D'Ambra

A Conceptual Framework for Aligning Business and Information Systems from Organisational Learning Perspective 41
Hamad Balhareth and Kecheng Liu

Business and ICT Alignment in Higher Education: A Case Study in Measuring Maturity 51
Balbir S. Barn, Tony Clark, and Gary Hearne

Investigating Post-outsourcing Decisions: Using the Intellectual Capital View 63
Hamish T. Barney, Aybüke Aurum, Graham C. Low, and Kevin Wang

Improved Initial Class Diagrams with ATSA:OO 77
Robert B.K. Brown and Angela M.E. Piper

Information Systems for the Social Well-Being of Senior Australians 89
Lois Burgess, Helen Hasan, and Carole Alcock

A Framework for Evaluating the Impact of High-Bandwidth Internet Provision and Use on Digital Literacy and Social Life Outcomes in Australia 101
S. Dane, M. Fahey, C. Mason, R. van der Zwan, J. Tucker, D. Bradford, and C. Griffith

A Social-Creative-Cognitive (SCC) Model for Requirements Engineering 113
Linda Dawson

Beyond Interface Design: Considering Learner Cognition When Designing E-Learning Systems 127
Mark Freeman, Abdallah Alasraj, and Paul Chandler

Methodological Framework for the Transition from Requirements to Software Architectures 139
Matthias Galster, Armin Eberlein, and Li Jiang

Integrating Enterprise Architecture and IT Service Management 153
Nelson Gama, Pedro Sousa, and Miguel Mira da Silva

Software Process Management: A Model-Based Approach 167
L. García-Borgoñon, J.A. García-García, M. Alba, and M.J. Escalona

Improving NDT with Automatic Test Case Generation 179
J. Gutiérrez, M.J. Escalona, M. Mejías, F. Domínguez, and C.R. Cutilla

Architectures for Evolving System of Systems in Complex Environments 193
Igor Hawryszkiewycz

A Multimodeling Approach for Quality-Driven Architecture Derivation 205
Emilio Insfran, Silvia Abrahão, Javier González-Huerta, John D. McGregor, and Isidro Ramos

IS Degrees: Sociotechnical or Technosocial? A Reflective Analysis 219
Jenny Coady, Tessa Berg, and Rob Pooley

Model-Driven Development of Model Transformations Supporting Traces Generation 233
Álvaro Jiménez, Juan M. Vara, Verónica A. Bollati, and Esperanza Marcos

An Approach for Identifying Business Requirements for Development of Prospective ERP Systems 247
Björn Johansson and Sven A. Carlsson

Conflict in the Value Chain: An Analysis of Competitive Advantage in Requirements Management of ERPs 259
Björn Johansson and Mike Newman

Evolving IT Management Frameworks Towards a Sustainable Future 271
Marcel Korte, Kevin Lee, and Chun Che Fung

Improving Efficiency of Managing Projects by Quality-Oriented Modelling of Processes 285
Michał Kuciapski

A Performance Management Model for Agile Information Systems Development Teams 297
Garry Lohan, Michael Lang, and Kieran Conboy

ELISA: Extensible Layer for Internet Services and Applications 309
Lubos Matl, Vladimir Kloucek, Viktor B. Bohdal, Jan Kubr, and Tomas Cerny

Knowledge Networks and Innovation-Driven Collaborations: A Study in Victoria's Biotechnology Industry 323
Adel Moslehi, Henry Linger, and Kerry Tanner

Global Development, ICT4D, Education for All 337
Anders G. Nilsson and Thérèse H. Nilsson

Towards Improving Information Systems Interoperability in Disaster Management 351
Ovidiu Noran

Enterprise Architecture Modelling for Corporate Sustainability 365
Malgorzata Pankowska

Understanding IS Risk Emergence from Escalation Perspective 377
Nipon Parinyavuttichai and Angela Y. Lin

Profiles in Professional Social Networks 387
Jaroslav Pokorný

XML Query Adaptation as Schema Evolves 401
Marek Polák, Irena Mlýnková, and Eric Pardede

Handling the Complexity of ISD Projects with Agile Methods: A Conceptual Foundation 417
Corina Raduescu and Asif Qumer Gill

OptBPPlanner: Automatic Generation of Optimized Business Process Enactment Plans 429
Andrés Jiménez, Irene Barba, Carmelo del Valle, and Barbara Weber

Modelling Life Cycles of Generic Object Classes 443
Vaclav Repa

It Is About Time: Investigating the Temporal Parameters of Decision-Making in Agile Teams 455
Niamh O. Riordan, Tom Acton, Kieran Conboy, and Willie Golden

Towards Monitoring Correspondence Between Education Demand and Offer 467
Peteris Rudzajs and Marite Kirikova

IT Governance: A Baseline Implementation Roadmap of a Bank in Thailand ... 481
Sureerat Saetang, Abrar Haider, and Andy Koronios

DBMS Application Layer Intrusion Detection for Data Warehouses ... 493
Ricardo Jorge Santos, Jorge Bernardino, and Marco Vieira

Crowdsourcing Large-Scale Ecological Monitoring: Identifying Design Principles to Motivate Contributors ... 509
Christoph Schneider and Frederik von Briel

An Analysis of Enterprise Architecture Frameworks from a Coherent Enterprise Description Viewpoint ... 519
Vladimir Selcan and Alena Buchalcevova

Query-Driven Method for Improvement of Data Warehouse Conceptual Model ... 531
Darja Solodovnikova, Laila Niedrite, and Aivars Niedritis

Capabilities and Affordances of Virtual Worlds for People with Lifelong Disability ... 543
Karen Stendal and Judith Molka-Danielsen

A Triple-Helix Model of Sustainable Government Information Infrastructure: Case Study of the eProcurement System in the Indonesian Public Sector ... 555
Fathul Wahid

Usability Work in Agile Systems Development Practice: A Systematic Review ... 569
Adeola Wale-Kolade, Peter Axel Nielsen, and Tero Päivärinta

Data and Information Quality Research in Engineering Construction Projects: A Review of Literature ... 583
Soffi Westin

Aligning Communication Analysis with the Unifying Meta-model for Enterprise Modeling ... 595
Iyad Zikra, Sergio España, Marcela Ruiz, Oscar Pastor, and Janis Stirna

Index ... 609

Managing Sustainable Information Systems Development: Complexity, Culture, Balance and Mindfulness

Karlheinz Kautz

1 Introduction

Sustainable information systems development (ISD) in the context of this paper is not about products that support sustainability at large with its environmental, economic and social dimensions and little about the development of sustainable products, which are both without doubt important topics. This paper is about a prerequisite for such products, namely, a sustainable ISD process, a process which exhibits reasonable and responsible stewardship and utilisation of the existing resources for ISD—people and information in the context of scope, time/schedule, budget/cost, quality and risk.

The paper will focus on the ideas of culture, balance and mindfulness and their significance for sustainable ISD. It builds upon an understanding of ISD projects as complex endeavours and Schein's (2004) and Schneider's [1] work on organisational culture theory and its application in information systems, Vidgen and Wang's [2] work on complex adaptive systems theory and balancing the resulting challenges in ISD and Butler and Gray's [3] work on mindfulness theory and information systems. Beyond a conceptual treatment, the paper will relate these notions to agile ISD and illustrate their applicability and impact on sustainable ISD through empirical examples. In this respect the paper is also based on my earlier work on contemporary ISD projects as complex adaptive systems [4], on organisational cultures in ISD [5], on ISD project's balancing at the edge of chaos [6] and mindfulness in ISD projects [7].

K. Kautz (✉)
Department of Operations Management, Copenhagen Business School,
Solbjergplads 3, 2000 Frederiksberg, Denmark

School of Information Systems & Technology, University of Wollongong,
Wollongong, NSW 2522, Australia
e-mail: kautz@uow.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_1, © Springer Science+Business Media New York 2013

2 ISD Projects as Complex Endeavours

Contemporary ISD is acknowledged to be a complex activity (see, e.g. [8–10]). Benbya and McKelvey [11] identify and analyse sources of complexity in contemporary ISD as changing user requirements; changing organisational needs; changing external competitive conditions; increased interdependencies among the involved individuals, organisations and technologies; and the rapid evolution of IS and IT. They state that ISD is often viewed as a complex top-down process and lament that such a perspective falls short in dealing with the identified but often unexpected contingencies of the ISD process. As an alternative they propose a conception of ISD based on complexity science and theory [12–15] and propose that ISD projects, following a special branch of complexity science, should be viewed as complex adaptive systems (CAS) and are better understood through the application of CAS. According to complexity science, complexity generally refers to an emergent property of systems, which are made of large numbers of self-organising agents that interact in a dynamic and non-linear fashion and share a path-dependent history (Jacucci et al. [16] citing Cillers [17]). CAS theory is a branch of complexity science that studies how a complex system can be adaptive to its environment and how innovative properties of a system emerge from the interactions of its components [2].

A complex adaptive system consists of a large number of loosely interconnected autonomous parts or agents, each of which behaves according to some set of, sometimes, rather simple rules; these rules require agents to adjust their behaviour to that of other agents, whom they interact with, and to adapt to each other; the resulting system behaviours can be very complex [2].

Interaction is a significant concept in this context as "the behaviour of the system is determined by the nature of these interactions, not by what is contained within the components. Since the interactions are rich, dynamic, nonlinear, and are fed back, the behaviour of the system as a whole cannot be predicted from the inspection of its components. The notion of "emergence" is used to delineate this aspect" (Jain and Meso [18] citing Cilliers [19]). CAS theory rests on the idea that order emerges through the interaction of the agents [11].

Emergence is the property of complex adaptive systems, which creates some greater property of the whole, the system behaviour, from the interaction of the parts, the agent behaviours. The emergent system behaviour cannot be predicted or fully explained from the measured behaviours of the agents [9]. Interaction and emergence are closely related and link together other generally acknowledged properties of CAS. Beyond interconnected autonomous agents, self-organisation, co-evolution, the poise at the edge of chaos, time pacing and the poise at the edge of time are other concepts that are frequently used when discussing CAS.

Interconnected autonomous agents, human or non-human, have the ability to independently intervene and determine what action to take given their perception of their environment. Yet they are interconnected and interact in such a way that they collectively or individually are responsive to change around them, but not overwhelmed by the information flowing to them by this connectivity [2, 20].

Self-organisation is the capacity of interconnected autonomous agents to evolve into an optimal organised form without external force. This form results from the agents' interaction in a disciplined manner within locally defined and followed rules and as such requires a departure from command and control management [2, 21].

Co-evolution relates to a complex adaptive system, and/or its parts, changing at a sustainable rate to alter their structures and behaviours. Such changes are in response to the interactions of the parts of the complex adaptive system and to its interaction with other CAS, which coexist in an ecosystem where adaptation by one system affects the other systems. This in turn leads to further adaptations and reciprocal change where the CAS do not evolve individually, but in concert [2, 12].

Poise at the edge of chaos describes the ability of a complex adaptive system to be at the same time stable and unstable, to never quite lock into place, yet never quite fall apart. The edge is the place which provides the stimulation and freedom to experiment and to adapt, as well as for novelty to appear, but also provides the framework, with sufficient structure, to avoid disorderly disintegration. A complex adaptive system that is driven to the edge of chaos has a competitive advantage as it will outcompete those CAS which are not [2, 22–25].

Time pacing in this context indicates that a complex adaptive system creates an internal rhythm that drives the momentum of change and that change in the complex adaptive system is triggered by the passage of time rather than the occurrence of events. This stops the complex adaptive system from changing too often or too quickly [2, 22].

Finally, poise at the edge of time conceptualises a complex adaptive system's attribute of simultaneously being rooted in the present, yet aware of the future. This allows the complex adaptive system to balance and synchronise between engaging enough exploitation of existing resources and capabilities to ensure current viability with sufficient exploration of new opportunities to ensure future viability [2, 21, 22].

All these core concepts are heavily intertwined and mutually reinforcing [26]. In summary, complex adaptive systems can be characterised through the emergence of co-evolutionary, self-organised behaviour, structure and order through the interaction of interconnected autonomous agents in a time-paced rhythm balancing at the edge of time.

CAS theory has been recognised as a valuable approach to understand contemporary ISD, and several research teams have applied the theory and provide organising principles and suggestions for best practice for the ISD process.

Meso and Jain [10] identify seven principles of CAS that are intuitive and easy to grasp. They are the principles of open systems, of interactions and relationships, of transformative feedback loops, of emergent behaviour, of distributed control, of shallow structure and of growth and evolution. The authors provide a conceptual argument, which they support with various examples from the literature. They map these principles to agile development practices and from this mapping derive more concrete, yet generic recommendations for best practices for ISD. In brief, their recommendations are:

1. Let actual solutions, processes, work patterns, team configurations and interactions emerge naturally rather than from heavy planning and design.
2. Do minimal planning where and when necessary; limit emphasis on documentation.
3. Allow for manageable experimentation in product design and with processes for learning from mistakes.
4. Develop, test and validate the IS iteratively and compare with current and past solutions.
5. Allow solutions to be responsive to emerging changes in requirements by taking into account feedback from frequent releases.
6. Strive for componentisation and loosely coupled IS.
7. Allow the development process to emerge or be determined by local needs.
8. Use an iterative, time-boxed and modular development process with measurable process milestones.
9. Re-evaluate the development process and methodology frequently.
10. Delegate responsibility and decision making to local development units.
11. Involve stakeholders and fellow developers by listening to their comments.
12. Re-evaluate team configuration frequently and allow for pairing or teaming up of developers.

Vidgen and Wang [2], again with a focus on co-evolution and agile development, develop a framework, which is grounded in the work of Volberda and Lewin [21] on co-evolving self-organising organisations. They identify practices similar to those identified by Meso and Jain [10], but consider them as enablers of agility.

My work is based on these predecessors, but I extend the line of reasoning and argue that rather than being enablers for agility, these practices are properties of complex adaptive systems to cope with complexity and to be adaptive. In ISD, understood as CAS, certain characteristics of the process facilitate good performance and support a sustainable development process while others inhibit it. I contend that these characteristics are not just limited to and important for agile development methods but are generally necessary, from a CAS perspective, for a contemporary, sustainable ISD process that has to deal with turbulent and changing environments.

CAS theory sheds new light on the discussion concerning the advantages and disadvantages of mixing plan-driven and agile approaches, methods and practices [27–29] and goes beyond actual problems and practices, as are currently coined in the literature, to form a deeper theoretical understanding of ISD [30]. While integration based on such simple classifications from a pragmatic perspective might be important and is supported by CAS theory, it seems to be based on a false dichotomy between agile and other methods [31] and appears to be theoretically secondary and less relevant. It might even render redundant the plan-based versus agile approaches debate as an “either-or” matter [32] as long as the chosen approach allows for organising for adaptation and sustainable ISD based on a set of simple foundational rules.

3 Cultures of Complexity

Several authors underline the importance of culture for agile development [33–36] and argue for an innovative and open organisational culture. I now discuss my position with regard to the relationship between organisational culture and sustainable ISD.

My work is grounded on Schein [37], who defines culture as "A pattern of basic assumptions—invented, discovered or developed by a given group as it learns to cope with its problems of external adaptation and internal integration—that has worked well enough to be considered valid and, therefore, to be taught to new members as the correct way to perceive, think and feel in relation to those problems". He conceptualises an organisational culture as having three distinct layers. The first layer holds the basic assumptions possessed by all members of that culture. These assumptions concern the members' fundamental values about reality, human relationships, human nature and the environment. They are taken for granted, subconscious, invisible and most difficult to change. The second layer holds the values and beliefs to which members subscribe. They are espoused, overt and debatable unlike basic assumptions. They are visible in the organisation's adaptation to the external environment and in the integration of its internal processes. In this context the organisation's mission, its attitude towards its customers, its resources and measurements, and its approaches to improvement play an important role as do its world view, its management and communication style, its distribution of responsibilities and its employees' mutual respect for each other as well as their organisation in groups and the groups' boundaries. In the third and most visible cultural layer, labelled artefacts and creations, visual and tangible manifestations of culture are found. These manifestations include the organisations' buildings and physical appearance, their technology, art, traditions, and visible and audible behaviour patterns expressed through language and stories. I will use these characteristics to analyse some organisational culture of ISD.

Schneider [1] provides a cultural analysis framework which is used by authors such as Highsmith [34] to explain his understanding of the relation between culture and ISD. In his framework Schneider [1] differentiates an organisation's focus between actual experience, reality and present needs or anticipating future possibilities. He further differentiates whether its judgement and assessment processes are impersonal and objective or personal and subjective. Schneider distinguishes four basic organisational cultures, which he labels competence culture, control culture, collaboration culture and cultivation culture. The four core cultures can be summarised as follows (see Table 1).

In a competence culture, the organisation's focus is on future possibilities and the organisation's judgement process is governed by objective measures. The organisation appreciates its employee's performance and cherishes the competent, superior individual. It prioritises work and its guiding question therefore is how a work task can be solved. With regard to management style, the organisation is led by expertise. It is organised in projects and recruits staff to improve its competitive strength.

Table 1 The characteristics of the four organisational core cultures

	Competence	Control	Collaboration	Cultivation
Contents	Future possibilities	Reality/present needs	Reality/present needs	Future possibilities
Process	Objective	Objective	Subjective	Subjective
Appreciates	Performance	Power and certainty	Attachment	Self-realisation
Cherishes	Superior individual	Compliance with plans	Teamwork	Creative unfolding
Prioritises	Work	The system	People	Idea
Asks	How?	What?	Who?	Why?
Is lead by	Expertise	Authority	Roles	Charisma
Is organised as	Projects	Hierarchy	Permanent teams	As little as possible
Recruits by	Competitive strength	Reliability	Capability to cooperate	Genius

In a control culture, the organisation's focus is on present needs and the organisation's judgement process is governed by objective measures. The organisation appreciates power and certainty and cherishes compliance with its plans. It prioritises its predefined and structured work system and therefore does not ask how and why a task has to be done, but what task has to be performed. With regard to management style, the organisation is lead by authority and it is hierarchically organised. It recruits employees to secure the organisation's reliability.

In a collaboration culture, the organisation's focus is on present needs and the organisation's judgement process is governed by subjective measures. The organisation prioritises people and appreciates the employees' attachment to the organisation. Its guiding question is "who is performing a task, and for task performance, it cherishes teamwork. Therefore, work is organised in teams, usually in permanent teams. Managers lead the organisation as role models and recruit employees with regard to their capability to cooperate.

In a cultivation culture, the organisation's focus is on future possibilities and the organisation's judgement process is governed by subjective measures. The organisation appreciates its employees' self-realisation and cherishes creative unfolding. Therefore, it prioritises ideas and asks why something should be done. The organisation is organised as little as possible to allow for the generation of as many ideas as possible, and for this purpose, it recruits its employees by genius. The organisation is led by its managers' charisma.

I will also use Schneider's framework to provide two examples of organisational cultures that I consider supportive to sustainable ISD. My examples are based on a comparative, exploratory, qualitative case study [38] of some Danish ISD companies, the two I have chosen here are named CoopCul and CompCul to respect their request for anonymity.

3.1 Examples of Organisational Cultures: CoopCul

CoopCul was founded in 1989 and has 21 employees. It has long-term relationships with a few customers with whom they work on the basis of time and material contracts. The project which builds the basis for my example is concerned with the development of a customer relationship management system for one of their main customers, a real estate business. Although each project is free to develop its own development process, in practice the development is strongly influenced by Scrum[1] and XP. Three developers and one supporter work on the project, with one of the developers acts as project leader and Scrum master. The organisation follows a management style where the individual employee has a mandate to (mostly) independently make his or her own decisions.

CoopCul has a flat management structure. The management group consists of four people, everyone else is a developer. The individual employees have the liberty and the responsibility to make their own decisions based on the company's ethical principles about what is right and what is wrong. Knowledge and experience are shared by moving staff between different projects. Business process and project knowledge is shared through daily meetings. CoopCul believes in the strength of its staff. Thus each staff member has 100 h per year which he or she can freely use for competence improvement. The overall management style can be characterised as "all power to the developers". In principle all developers are equal and have the same decision mandate. In practice some of the developers have greater experience and on this basis exert more authority. The project leader role is taken by these developers.

CoopCul's developers appreciate the freedom and trust management and customers give them. They have an open and honest relation to their customers. As it is important to inform and involve customers as soon as problems are discovered, the developers themselves contact the customers to ask for information or to inform them about problems. In this context CoopCul tries to avoid fixed price contracts, where everything has to be specified, although the customers often do not have a clear idea of their needs and requests. To elucidate these needs together with the customer, CoopCul uses graphical mock-ups or blackboard drawings. The relationship with the customer is further supported by the software tool which registers development time quite precisely. The customers have access to this registration system and can see how much time has been used for what task. Beyond this, gut feeling and intensive communication with the customers are used to assess the status of a project.

3.2 Examples of Organisational Cultures: CompCul

CompCul is a consultant and product development company founded in 1998 with 25 employees with 8–10 staff working on the company's main product. This product

[1] As the particular ISD methods are not in the focus of the presented argument, they are not further explained in this paper. Further information can be found in the texts referenced in Sect. 1.

is a platform for digital media products which has been developed in different subsystems and is currently in version 2.1. The development tasks are performed in 14 day sprints and are distributed among the developers every second week. CompCul's development process is inspired by Scrum, XP and organisational patterns for agile development as introduced by Coplien and Harrision [35]. The company's chief technical officer acts as product owner.

CompCul's mission is to provide quick and speedy high-quality expertise. They take on time-critical projects and assignments and aim at changing their customers' development processes by actively intervening in concrete projects. CompCul wants to be seen by their customers as humble, but professional. The chief technology officer acts as product owner and project leader for the digital media product and is in constant dialogue with customers, sales personnel and support staff. The digital media product is a standardised product, but each customer gets their own unique solution. As process consultants CompCul's employees focus on software development and less on organisational development, and CompCul is characterised by a curiosity for technical novelties; everyone including the management follows up on and experiments with the newest technologies.

CompCul has long been inspired by agile principles, and they have become so much part of the company that they are not very visible on a day-to-day basis. For example, pair programming is only used to a very limited extend, primarily for education, but there are always two developers on each user project who are able to talk together. Productivity is important for CompCul. The company has a home-grown system for time registration which primarily is used for billing but which also provides an indication about a project's productivity. There is some resistance against further registration and measurements as well as against the use of project management tools such as Gantt diagrams and metrics. Daily morning meetings are the primary means of all communication with regard to the actual iteration of each project. Education on the company's product happens through introductory training and pair programming. Quality assessment and improvement is dealt with through a continuous integration server which builds the system, performs all tests and gives feedback on detected problems and errors. Each Friday, general knowledge sharing as a basis for further product and process improvement takes place.

CompCul has a very flat structure. All developers are directly under the directors in the hierarchy. CompCul's founder together with the company's chief technical officers leads the organisation by example, by always being the first to take on a task or an initiative. CompCul pursues full openness as everyone can see what everyone else earns and all profit is shared. The employees are proud of their craftsmanship. Everyone is on a high professional level and they strive for being among the best consultants in their field. To give something back to the development community at large, the employees contribute to the open source movement. There are not many formalised roles, but individual developers have to take responsibility for their development projects. The daily meetings allow for swift follow-up of the developers' areas of responsibility and their assigned tasks. Everyone in the development team has been at a Scrum master course, and they are all equal with regard to their knowledge about the development processes. In the digital media project, there is

Table 2 The core cultures of the example organisations

The organisation	CoopCul	CompCul
Contents	Reality/present needs	Future possibilities
Process	Subjective	Subjective
Appreciates	Self-realisation	Attachment
Cherishes	Teamwork	Superior individual
Prioritises	People	Work
Asks	Why?	Who?
Is lead by	Charisma	Expertise
Is organised as	Permanent teams	Permanent teams
Recruits by	Capability to cooperate	Competitive strength

nevertheless a certain difference between experienced developers and less experienced developers who are more likely to do what they are asked. At CompCul knowledge is shared and not kept back for one's own advantage. The employees care for each other, are observant and are willing to help each other beyond the boundaries of any formal groups. There is no a central architect or leading programmer, but a joint codes base which all can access. CompCul holds the following basic ethos: At CompCul, everyone is open and considerate to each other, loves to develop software and wants to be the best in doing it; at CompCul, people are more important than processes and family is as important as work.

3.3 Identifying Core Cultures

Table 2 summarises the results of the comparison of the core cultures of the example organisations and the projects that were investigated. With regard to the organisation's focus, CoopCul develops software for a customer, who has some requirements and who wants some problems solved, and although the organisation's professional developers of course also think of the future, they understand as their primary task to provide value to the customer by satisfying the customer's need and not to ponder about more extensive possibilities. CompCul is engaged in product development with no particular customer. Their developers try to exploit as many ideas and possibilities to reach as many customers as possible. At both CoopCul and CompCul, the overall development process management involves decisions concerning a particular project being taken by the development team itself and that human, subjective and emotional judgement—gut feeling—is as a rule the best to follow.

CoopCul appreciates its employees' self-realisation. Developers should dare to talk to the customer, be able to make their own decisions and should pursue further personal development. At CompCul attachment is most appreciated—the employees should feel good and at home in CompCul's developer "family". CoopCul cherishes teamwork, which is demonstrated by the developers' close cooperation and the mutual testing of their software. CompCul cherishes the superior individual with its strong technical competencies most. CompCul prioritises work highest, understood in such a way that it is deeply rooted in the culture of the two projects to strive for a solution of uttermost quality. CoopCul in extension of its emphasis on self-realisation prioritises the employees most.

The example organisations differ with regard to their central question: At CoopCul the emancipated, individual developer independently gets acquainted with a task or challenge and asks “why” a certain approach is better than another. At CompCul the question is “who”, they stress that the right developers have to work together on the right assignments.

CoopCul is lead by charisma, as with its value-based management form, the management team represents a catalyst and inspiration for the staff. At CompCul, the experienced developers and the chief technology officer act as Scrum masters and supervisors for the younger staff. This can be considered as management through expertise. At CoopCul and CompCul, the projects are organised as permanent teams.

Finally, at CoopCul it is more important that staff fit into the culture and like to have a relationship with the customer than that they have technical capabilities; thus, people are recruited based on their capability to cooperate. At CompCul more importance is attached to the technical competence of the employees—they select the best developers for the task and thus emphasise their competitive strength most.

Now applying Schneider’s [1] framework, we find the following core cultures.

CompCul—a competence culture. CompCul presents a distinct competence culture with some traits of cooperation and cultivation. Consultancy companies are typically competence cultures [1], and at the same time, this culture is well suited for the development of niche products for selected customers—such as CompCul’s digital media product [1]. CompCul does not have any traits of a control culture as this would be against “gut feeling” [1] which CompCul readily and often utilises. “Family is as important as work” is one of CompCul’s basic assumptions, and here, the culture shows characteristics of a cooperation culture which in contrast to a competence culture has the family as its archetype [1]. Also CompCul’s policy concerning transparent salary information and profit sharing shows more traits of a cooperation culture than of a competence culture. It is a characteristic of a competence culture that management has the same or even better technical competences than staff [1]. CompCul’s managing director has been a proficient developer and today he helps other organisations to implement sustainable ISD approaches. Competence as core culture is also mirrored in the fact that CompCul experiments with new technology and advanced prototypes of their product. CompCul has been founded with a view to be a front runner of the trade. This idea is still central and attracts competent staff. The company is a consultant company organised around technical specialists which is reflected in their product. CompCul does however not only recruit experienced, technically superior staff but also employs and trains newly graduated people without experience, but with perceived potential.

In conclusion, I argue that ISD thrives not only in competence and collaboration cultures but also in cultures which might have a strong element of control or cultivations—as long as they recognise ISD projects as complex adaptive systems.

4 Balancing at the Edge: Project Management Challenges

Another concept I relate to sustainable ISD is inspired by Vidgen and Wang's [2, 26] work on CAS theory and ISD practice. They summarise the key concepts of CAS in three guiding principles for the organisation of ISD to thrive in a competitive, unpredictable environment of continuous and unanticipated change, an ability which Goldman et al. [39] and Dove [40] link to the concept of agility. These principles are matching co-evolutionary change rate, optimising self-organisation and synchronising concurrent exploitation and exploration. Matching co-evolutionary change deals with handling stability and instability at the same time and can be investigated through looking how requirement changes are monitored and tracked and how the user requirements' change rate is matched or exceeded. Optimising self-organisation considers the distribution of decision making and local autonomy and answers to questions such as how is management distributed, how are capabilities of individuals maximise and how are communication and collaboration facilitated. Synchronising exploitation and exploration inquires into how a project is rooted in the present, but aware of the past and possible futures, and how exploitation and exploration of resources, especially knowledge, are synchronised.

Wang and Vidgen [41] also identify the edge of chaos as a key concept of CAS as the edge of chaos is at the same time stable and unstable [24]. Edge of chaos is the part of a system, which never quite locks into place, yet never quite dissolves into turbulence [42]. It is also the place which provides organisations with both the stimulation and freedom to experiment and to adapt but also to avoid disorderly disintegration [25] and thus outcompete those systems that are not at the edge [13, 23]. Edge of chaos is the place where the really interesting behaviour occurs [41]. In this context Wang and Vidgen [41] developed another framework. It takes as a starting point four pairs of values, which according to the advocates of agile development guide ISD in its complex setting. The value pairs are (1) *individuals and interactions* over processes and tools, (2) *working software* over comprehensive documentation, (3) *customer collaboration* over contract negotiation and (4) *responding to change* over following a plan. With regard to these values, two critical boundaries demarcate the edge of chaos as the area of having "just enough structure" where ISD projects are poise at the edge of chaos: the region of emergent complexity and where the projects are able to balance order and chaos. These boundaries identify where too much structure leads to bureaucracy with too rigid processes and rules, too much documentation, too much emphasis on contracts and their negotiation and too much focus on following project plans. On the other hand, too little structure leads to chaos with too loose, if at all, defined processes and rules; too much emphasis on working software; too much focus on collaboration; and too much response to change requests. In Highsmith's [34] words, the edge of chaos is where ISD projects show agility, which he defines as the ability to balance flexibility and stability. On this background I will now briefly illustrate the three guiding principles of CAS and their role in sustainable ISD.

4.1 Matching Co-evolutionary Change at the Edge of Chaos Through Time Pacing

Accepting change has already been identified as a cultural feature in the examples presented above. In another project, change, especially change of requirements, was an accepted fact of life. This example is based on an empirical case study of a commercial ISD project in a large German public sector organisation performed by a German software company. For more information on this project, see the Appendix and again the literature referenced in Sect. 1.

In this project, dealing with change was an integral part of the daily activities. This involved balancing how the project responded to changed circumstances against following a plan. Beyond the scheduled meetings, some of the customer organisation's subproject managers turned up at the project nearly on a daily basis while others came regularly and looked over the shoulders of the developers. The development organisation's project manager commented on the customer behaviour and formulated the project's overall principle to handle change requests: "Yesterday he said something and today he says something else. Requirements have to be clear at the beginning of an iteration and cannot change right in the middle of it. We are agile but not on a daily or hourly change request rate". Thus, different feedback mechanisms were introduced and provided structures for collecting ideas and change requests. Change requests were presented by the customers and users on a short time frame via weekly and bi-weekly feedback sessions and scheduled weekly meetings for all subproject managers. These meetings contributed to an iteration and were based on presentations and demonstrations of working software. The customer representatives also performed regular "road shows" in the user departments to collect feedback and ideas and proposals for improvements. Finally, change requests were detected and collected through the scheduled acceptance test sessions for each iteration or a release with customer representatives. Change requests, as a rule, were dealt with and implemented in the next iteration(s) with the exception of minor changes, which, after negotiations and mutual agreement, were implemented as part of the current iteration.

The different feedback mechanisms provided some structure to handle the changes, but plans and planning although not impeding more spontaneous actions played an important role as well. Even the weekly sessions were to some extent planned, as were of course the acceptance tests. As one customer's subproject leaders relating to the size and complexity of the project said "Planning is essential in such kind of projects". Therefore, the project also had an overall long-term plan covering a 14-month period anticipating 3–6 releases depending on the subprojects. A more fine-grained plan was developed for the individual iterations, which together made up a release, that was detailed to a single week. Planning sessions, also called planning games and story cards, then offered the means to perform planning at the most detailed level for very short periods of time. The frequent planning sessions embedded in a "larger" and coarser plan together with the different means to handle change provided just enough structure for the project to move forward and to balance stability and flexibility at the same time.

In this project, a two-phased development approach and the overall project plan set the time frame for the development process. Project activities were performed continuously in accordance with an internally set pace. This included the frequent releases, formal weekly and bi-weekly meetings and feedback sessions, as well as the daily meetings, and in particular the iterations, which were of varying duration but were comparably short and of a predefined length. Regular planning and, if necessary, rescheduling with regard to releases, iterations and daily activities provided the appropriate intervals to match and handle the changes that were brought up continuously during the project. Together with the manageable size of the requirements on the story cards, it supported the emergence of a lasting working rhythm, which becomes most evident in the way the pair programming worked both with regard to shifting partners and sharing the keyboard (see next subsection 4.2). This way, time pacing, as also reported by Vidgen and Wang [26], was an organising principle created on an internal rhythm which drove change in the project and simultaneously allowed for stability and flexibility [2].

The project was continuously in a state of bounded instability, which means that it—paradoxically—was simultaneously stable and unstable [24]. The two-phased development approach, the formal contract, the initial project plan and the first requirements specification as well as the overall implemented project organisation all acted as superordinate structuring mechanisms that created a relative stable space within which the development process and the working software could unfold. Beyond the direct reaction on manageable clarifications and refinements of existing requirements, the handling of the constant requests for changes and new requirements illustrates how the project manoeuvred in a region of emergent complexity and balanced at the edge of chaos. The organising principles used in the project all allowed for the flexible handling of the large amount of new, incoming demands and ideas and at the same time provided a frame for stability as they structured the project participants' day-to-day activities and helped them to know what to do, when and what to expect from others. These mechanisms included short iterations with frozen requirements, the frequent planning including the estimation of tasks with regard to the developers' capabilities, the prioritisation of tasks with regard to the actual situation, the stand-up meeting and feedback sessions and not least the size of the individual tasks and user stories whose descriptions could exceed not an A5 story card.

4.2 Optimising Self-Organisation Through Autonomy

Self-organisation and self-management in the project were brought about through the distribution of decision making and local autonomy to facilitate the team member's interaction and communication and to maximise their capabilities. This is in line with the value of individuals and interaction over processes and tools. Pair programming was a prominent process to support the interaction of the individuals and organised the individuals of the development teams to work in shifting pairs of developers in front of a screen while implementing the requirements written down as user stories on story cards as executable code. The developer pairs emerged through spontaneous formations, as one developer put it “first I look who is free and then I go and work with him”. Two mechanisms here were important: (1) to regularly shift a partner and (2) to regularly shift possession of the keyboard within a pair. The developers found it difficult to find the appropriate synchronisation points at which to change a partner in the team of four developers. No common practice existed. However, they did not follow an overly bureaucratic rule such as shifting partner every morning regardless of the status of a story card. To avoid both too much red tape and too much disorder, some developers preferred to stay with a partner until a card was closed. “…changing a partner was always a problem, it still is as changing in the middle of a card seems foolish to me and I don’t really like doing it…”, said one developer. However, a subproject leader might intervene if a pair had worked together for too long, say, 3 days. In doing so, a balance, “just enough structure” was created between shifting too often and not shifting at all. The developers also started out with a practice, which did not really support the objectives of interaction, namely, that one developer exclusively held the keyboard and programmed, while the other watched and sometimes commented. To avoid such situations, a process was introduced using a stopwatch so that after 20 min the keyboard had to switch. This was however abandoned as too bureaucratic and not fruitful in the work environment. The pairs found their own rhythm. “We don’t do that anymore. It didn’t function. Well, now it also functions without any explicit rule.” was how one developer commented the emerged practice.

This had also been the case with stand-up meetings, which like the planning games were also used to structure the distribution of work tasks. The developers either chose tasks themselves or in common consent were assigned tasks—sometimes on the spot—by their subproject manager without any involvement of the project manager. One subproject manager described this as follows: “In a stand-up meeting a developer had a good idea and I assigned the task to him. The project manager allows such decisions made by ourselves”. The stand-up meetings were originally performed by all teams together every day with the purpose to keep everyone up to date with the current status of the project and to exchange useful information. To start with these sessions were quite detailed and long and considered as not very efficient and useful. The team members themselves refined this practice, and shorter meetings, distinguishing between the overall project and the subprojects, were introduced. This arrangement was acknowledged as very helpful. Other intensive interaction took place at the beginning of each iteration, where all story cards were jointly discussed. This illustrates how disciplined individuals and teams behaved throughout the project and how self-discipline and peer-triggered discipline, together with all other mechanisms, contributed to self-organisation. Moreover, it also illustrates how such behaviours supported a balance between order and chaos as well as stability and flexibility and provided enough structure for the project to progress by keeping the project teams informed and decreased the need for documentation.

In the project, staff acted as autonomous, interconnected agents. Their autonomy was expressed in numerous ways. The autonomy of the customer staff showed when they took part in prioritising, estimating, designing and especially approving design decisions where the customers had a mandate to make decisions and exerted power. Autonomy also became apparent in the liberty the developers had when distributing and picking tasks. When implementing the story cards, the developers were largely self-governing with regard to the design decisions they made. Finally, the developers were sovereign when choosing a partner for the pair programming. Still, the project participants were highly interlinked and maintained their relationships through the structures and measures described above. These structures and measures supported various forms of direct interactions to achieve interconnectivity across and within the four development teams. The emergence of learning of the entire team was in general encouraged by a number of factors. These included the physical presence and closeness of staff in the project facilities which were onsite at the customer premises, the sharing of project-relevant knowledge in the scheduled weekly meetings, feedback sessions and daily stand-up meetings, the involvement by all team members in project management, in design decisions and in all development activities, their self-assignment of tasks based on competence and interest as well as the rotation of direct collaboration partners in the programming pairs. Such emergent learning is also reported by Vidgen and Wang [2].

The concept of self-organisation is closely related to the concept of interconnected, autonomous agents [2, 21]. It is thus not unexpected that the same structures and mechanisms were prominent in the support of self-organisation. The description of how the practices of stand-up meetings and pair programming were first implemented, and later refined, illustrates the emergence of self-organisation of autonomous project members and the subsequent emergence of order in the development process of the project. It also shows that individual and team discipline are not in conflict with, but a vital element of, self-organisation.

The role of the project managers also deserves some attention. With largely the same development tasks and their sharing of responsibility, they appeared more like peers in terms of team interaction and a part of the team who were not controlling but primarily acted as facilitators and created an environment that fostered self-organisation. This however creates a challenge for the project managers since they need to live as members inside the project team as well as providing direction and guidance. Finally, feedback, other than the self-organised regular feedback sessions, also reflects the self-organisation. Such feedback was provided and gathered from the customers who gave their comments in passing, by phone and by e-mail. This variety of forms was subsequently handled in an orderly fashion and contributed as part of the planning activities.

4.3 The Edge of Time and the Emergent Balance of Exploitation and Exploration

In the project at the centre of attention were always the current iteration and the current user stories, while taking the existing working software and the design for future extensions into account. The iterations were based on using the existing requirement lists to produce story cards and write working software while investigating prospective options with information from and in consultation with customer staff. Such iterations have their root in the presence but are simultaneous awareness of the past and the future. Moreover, they balance the concurrent exploitation of existing knowledge and the exploration of new knowledge. This has been characterised as the edge of time by Brown and Eisenhardt [22]. The daily meetings and the pair programming served the purpose of keeping a focus on today at the same time as preparing for the future. Other project structures and mechanisms also supported a focus on and constituted a manifestation of both the past and the future, including the frequent presentations of working software. The overall project plan, as well as the frequent planning sessions, were structured around releases, iterations, planning games and the implementation of working software. Beyond product-oriented exploitation and exploration, and providing the opportunity for direct reaction, the frequent manager meetings and feedback sessions were also used by the project participants to think about their own behaviour and to review and improve the development process. This is reflected in the flexible, but for each iteration, fixed time boxes and the developed formats for the meetings and the pair programming.

In conclusion of this section, we argue that in sustainable ISD projects the overall plans and requirements documents and the possibility to change, revision and refinement of these plans and requirements create conditions in which adaptive activities can flourish as they are not plan-driven, but planning driven [2]. We find that recurrent planning is a consequence of frequent feedback due to the close relationships and the interactions between all those involved in a project and that both high-level, sketchy, long-term plan and detailed, accurate, short-term plans are necessary and beneficial in such a process. This is consistent with the finding reported by Vidgen and Wang [2]. In such a context, managers need to care for their teams and nudge, remind and reinforce behaviours through communication with team members, and they need a subtler approach than command and control. They must handle the paradox of control as they are simultaneously in control and not in control. In other words, managers must be mindful.

5 Mindfulness in ISD Projects

Vidgen and Wang [2] in the context of self-organisation also identify collective mindfulness as a capability. In this last part of this paper, I show how individual and collective mindfulness and mindful behaviour may contribute to successful and sustainable ISD. Butler and Gray [3] suggest that individual and collective mindfulness increase organisations' ability to perform in problem-solving situations and dynamic, changing and unstable environments such as ISD. According to these authors "Collective mindfulness requires organizations to couple the ability to quickly detect issues, problems, or opportunities with the power to make organizationally significant decisions. This may be accomplished by moving decision-making authority or

Table 3 Aspects of individual mindfulness

Aspect	Characteristic
Openness to novelty	Ability to reason about and to cope with novel kinds of stimuli
Alertness to distinction	Ability to compare, contrast and judge about how things are the same or different
Sensitivity to different contexts	Awareness of situational characteristics to notice when and whether a setting changes
Awareness of multiple perspectives	Appreciation of things from different and opposing points of view
Orientation in the present	Individual's attention to their immediate situation and their actual surroundings

creating an organizational environment that enables the smooth interaction of perception and action". These characteristics are also associated with ISD when understood as CAS as I have argued above. Mindfulness includes the attention to detail, the willingness to consider alternatives and responsiveness to changes [43]. It endorses openness to novelty and an orientation in the present. As such, it also shows similarities to the prior presented four pairs of values.

According to Langer [43, 44][2] at the individual level, mindfulness lays emphasis on the ability to constantly create and use new categories in the perception and interpretation of the world. In contrast, Chanowitz and Langer [45] describe mindlessness as a state of reduced attention, which results from premature commitment to beliefs that may not accurately reflect the phenomena at hand. To conceptualise mindfulness, Langer [43, 44] distinguishes five constituent components, which are presented in Table 3: openness to novelty, alertness to distinction, sensitivity to different contexts, awareness of multiple perspectives and orientation in the present.

Openness to novelty is the ability to reason about, and to cope with, novel kinds of stimuli. Alertness to distinction is the ability to compare new categories with existing ones and decide if things are the same or different. This is specifically important when defining the nature of a problem as it can help to decrease the risk of misdiagnosing a problem. Sensitivity to context is an awareness of the characteristics of any specific situation which an individual faces. This is a prerequisite to being able to notice when situational traits change. Awareness of multiple perspectives enables people to perceive and analyse things from different and opposing points of view. Finally, individuals, who are oriented to the present, devote more of their attention to their immediate situation. In this context, Sternberg [46] argues that people, who pay attention to their actual surroundings, behave more mindfully than those who are careless about what is happening.

People who are mindfully engaged in a procedure perceive changes in an environment. Therefore, there is a probability that they are more creative and that they are more likely to adopt new ways of working, and consequently, it is also more

[2] The idea of mindfulness is however much older; it is an ancient Buddhist concept linked to Buddhist meditation and has been related to "the art of conscious living" (see ref. [50]).

Table 4 Aspects of collective mindfulness

Aspect	Characteristic
Preoccupation with failure	Utilisation of errors and failures as a way of improvement
Reluctance to simplify	Organisational aspiration to perceive problems from different points of view
Attention to operations	Individuals' capability to have an integrated overall picture of the operations in an organisation or project
Commitment to resilience	Ability to cope with problems and dangers as they occur
Migration of decisions to expertise	Migrating the problems to the experts, who are most capable of solving them, regardless of hierarchical levels

likely that they will find innovative solutions to problems and that they by altering their actions will take advantage of new situations.

Collective mindfulness presents a theoretical elaboration of the cognitive concepts related to individual mindfulness on the organisational level such as a business unit, a work group or a whole organisation. Collective mindfulness is also known as organisational mindfulness [3]. In collective mindfulness, existing expectations are continuously scrutinised and refined according to new experiences in order to be able to invent new expectations for dealing with unprecedented situations to improve foresight and current functioning [47]. Examples include hospitals providing services under tight resource constraints but also mundane organisations such as restaurants. According to Weick and Sutcliffe [47], there are five key aspects which characterise collective mindfulness: preoccupation with failure, reluctance to simplify, attention to operations, commitment to resilience and migration of decisions to expertise (see Table 4).

Preoccupation with failure focuses on the utilisation of errors and failures as a way of improvement. Emphasising errors and failures is helpful to prevent overconfidence in, and inattention to, a given situation. The organisational desire to continuously view problems from different points of view is referred to as reluctance to simplify. This is helpful in order to recognise minor anomalies and errors and to react appropriately to prevent larger failure in the future. Attention to operations focuses on individuals' (capability to develop an) integrated overall picture of the operations in an organisation or project. Commitment on resilience as opposed to focus on planning is the ability to cope with problems and dangers as they occur. Finally, the migration of decisions to expertise is the means to circumvent organisational hierarchies and for migrating problems to those experts who are most capable of solving them.

In summary, Butler and Gray [3] argue that collective mindfulness is associated with cultures and structures that promote open discussion and it increases the organisation's ability to perform in dynamic, unstable environments. They put forward that "Collective mindfulness is not simply the result of having individually mindful personnel. In general, mindfulness involves the ability to detect important aspects of the context and take timely, appropriate action".

5.1 Examples of Mindfulness

To demonstrate the role of mindfulness, I will revisit two of the earlier value pairs to deal with ISD in turbulent environments and to support sustainability. The value pairs are individuals and interactions over processes and tools, and responding to change over following a plan. I will map the value pairs to the constituent components and key aspects of individual and collective mindfulness and illustrate them with the selected examples from the project we introduced in the previous section.

The value “individuals and interactions over processes and tools” underlines that business people, customers, future users and developers should work together as closely as possible and as much as possible, preferably daily throughout a development endeavour, to achieve the project’s goals. The value emphasises that projects should be built around motivated individuals who should be trusted and should be provided with an environment and the support they need to get their jobs done. In this context, self-organising teams are promoted as the organisational unit from which the best architectures, requirements and designs emerge. The value also alludes that face-to-face conversation is the most efficient and effective method of conveying information to and within a development team. Thus, at regular intervals, the team should reflect on how to become more effective, then tune and adjust its behaviour accordingly. From a perspective of mindfulness theory, this emphasis on individuals and interactions is mirrored in the concepts of openness to novelty, alertness to distinction, sensitivity to different contexts and awareness of multiple perspectives. This underlines the individual’s capabilities to create and handle novelty, to assess similarities, dissimilarities and situational change and which salute contrasts and opposition. Taking collective mindfulness into account, the allocation of problems to those who are most qualified to solve them is obviously supported by such an approach as is the skill to handle predicaments as they happen and the organisational ambition to understand problems from different points of view. This shows that at least reluctance to simplify, the commitment to resilience and the migration of decision to expertise are echoed in this value pair.

The behaviour that led to the emergence of rhythm in the stand-up meetings is exemplified by the following statement of one project manager: “In the beginning we did this all together, but we found out that it can become too much, as some are doing something that is not of interest for other teams. But it is good to know what others do. It does not have to be in detail. And that is what the teams do now, all teams, but we keep it short”. Pair programming in accord with self-organisation and autonomy, while balancing at the edge of chaos and time, exhibits both characteristics of collective and individual behaviour in the form of alertness to distinction, sensitivity to different contexts, awareness of multiple perspectives, orientation to the present as well a willingness to learn from errors, an aspiration to perceive problems from different perspectives, an ability to cope with problems and the migration of decision to expertise. These characteristics are in line with what we argued contributed to the progress of the project in a sustainable manner.

The value "responding to change over following a plan" welcomes changing requirements, even late in the development process, and harnesses change for the customer's competitive advantage while keeping the amount of work performed to implement a change and produce a new release as low as possible. The frequent delivery and evaluation of working software, persistent, lasting development as well as good design and technical excellence, certainly facilitate responding to change as does the cooperation of the different stakeholder groups and a working environment which cherishes trust, expertise and direct communication. In such a context, all components of individual mindfulness are taken into consideration as well as all key aspects of collective mindfulness are dealt with.

In the project described above, the recurrent planning sessions, embedded in a large and coarser plan, together with the different means to initiate change, are instances of mindful behaviour, which helped moving the project forward. Initiating change by collecting ideas and suggestions through road shows in the user departments and to handle change request brought forward on a shorter time scale in weekly and bi-weekly feedback loops built into an iteration has the following consequence, as put by one developer: "…often we show the customer rep something once a week and then he's going 'well, I thought this would be different' … thus there are always small changes …". These frequent feedback loops also had the effect that minor misunderstandings were caught and dealt with as changes early before they could grow into something larger, as the same developer explained: "Until now it has not happened that everything was totally wrong; there are of course some refinements or a bug is found or something similar. There is always something". The feedback was taken seriously and immediately responded to with action: "Through the feedback we got, we could react directly." as described by one developer. Overall, the project's response to change and following of plans exhibits all aspects of mindfulness that can contribute to sustainable ISD.

6 Conclusion

In this paper I discuss the concept of sustainable ISD and demonstrate the strength of an approach which acknowledges the complexity of ISD projects and activities. This approach understands the organisation of ISD from the perspective of complexity science and complex adaptive systems (CAS) theory. As part of such an approach, I discuss the roles of culture, balance and mindfulness and illustrate this discussion with selected empirical examples. The discussion and examples show how they contribute to the organising principles, practices and capabilities which assist good performance in sustainable ISD. Thus, while my work contributes to a complexity theory of sustainable ISD, the presentation and examination of our examples also provides practical advice derived from this perspective to successfully cope with complexity in sustainable ISD in an adaptive manner.

7 Appendix

The empirical data for the case study which is presented in Sects. 4 and 5 was collected in semi-structured, open-ended interviews, which were conducted by a team of two researchers in a 3-day period on the development site. The study investigated a commercial ISD project in a large German public sector organisation. The project developed an operations management system (OMS) for the waterworks of a large German city by a German software company. The system was developed with a web-based graphical user interface and a back end to interface the technical infrastructure as defined by an underlying ERP system. The project was described by both the customer and the development company as a success. It was organised in four subprojects to provide IT support ranging from customer management to the maintenance of the sewer system.

A research team of two investigators performed 12 interviews with 11 individuals—the development organisation's project manager was interviewed twice. This included nearly a third of the development team covering project managers and developers with a wide range of experience and skills and a representative sample of key players and future users in the customer organisation. The interviews were tape-recorded and subsequently transcribed. For the qualitative data analysis, a software tool was used. The interview data was supplemented with company and project documents such as method, requirements and release descriptions, as well as project plans. The analysis of the data was performed in collaboration with a third researcher who was not part of the original interview team.

The ISD project was organised in two phases. When the study was performed, phase one, which had lasted 12 months, had been successfully closed, and phase two had been going on for 4 months. Responding to an inquiry during our first data analysis, the responsible project manager stated that phase two ended 10 months after our study, on time and budget with all parts of the IS being operational. At the time of the project, the development organisation consisted of about 25 employees, 20 of them were developers, and based its development approach on the development method extreme programming [48, 49] which provides a number of practices and tools to structure the ISD process. The formalised method includes planning techniques for releases and short iterations called planning games, A5-sized user stories and story cards to specify user requirements, onsite customers to support customer-developer communication, daily stand-up meetings for all project members, pair programming, collective ownership, re-factoring of the developed software as well as continuous integration and testing to develop the software proper. The development organisation had extended the method with some project management processes to cater for their larger projects such as an elaborate overall project plan, formal reporting mechanisms and a formal contract based on an initial requirements specification produced by the customer.

Information system	Operations management system (OMS) with web-based GUI user interface and ERP back end
Formalised method	XP:
	Short releases and iterations of 3–6 months/3–6 weeks
	Planning games, user stories, story cards, onsite users
	Pair programming, collective ownership, stand-up meetings
	Continuous integration, testing, re-factoring
Involved development team and developers	2 overall project managers (1 development organisation, 1 waterworks)
	Development organisation:
	Up to 12 staff with multiple roles as
	project manager, analyst, customer contact and developer
	Highly motivated and educated, limited XP experience
	4 subproject development leaders also as customer contacts, analysts, and developers with XP experience
	Waterworks:
	4 customer subproject leaders also as user representatives
	At least 1 additional user representative for each subproject, not the whole time onsite
Structural context	Formalised method clearly communicated to customer
	Development organisation: no experience with larger ISD projects using XP
	Waterworks: 1 project using XP
	Project team onsite in a waterworks building
	General requirements document as basis for contract
	Failed ERP implementation

Summary table of the structural elements of the OMS project

References

1. Schneider W (1994) The reengineering alternative: a plan for making your current culture work. Irwin Professional Publishing, New York, NY
2. Vidgen R, Wang X (2009) Coevolving systems and the organization of Agile software development. Inform Syst Res 20(3):355–376
3. Butler BS, Gray PH (2006) Reliability, mindfulness and information systems. MIS Quarterly 30(2):211–224
4. Kautz K (2012) Beyond simple classifications: contemporary information systems development projects as complex adaptive systems. In: The international conference on information systems (conditionally accepted for publication), Orlando, FL, USA, December 16–19, 2012
5. Kautz K, Pedersen CF, Monrad O (2009) Cultures of agility—Agile software development in practice. In: Proceedings of the 20th Australasian conference on information, Melbourne, Australia
6. Kautz K, Zumpe S (2008) Just enough structure at the edge of chaos: Agile information systems development in practice. In: Abrahamsson P et al (eds) Proceedings of the international

conference XP 2008, Limerick, Ireland. Springer Notes in Business Information Processing, Berlin, Germany, pp 137–146
7. Matook S, Kautz K (2008) Mindfulness and Agile software development. In: Proceedings of the 19th Australasian conference on information, Christchurch, NZ, pp 638–647
8. Truex DP, Baskerville R, Klein H (1999) Growing systems in emergent organizations. Comm ACM 42(8):117–123
9. Highsmith J (2000) Adaptive software development: a collaborative approach to managing complex systems. Dorset House Publishing, New York
10. Meso P, Jain R (2006) Agile software development: adaptive systems principles and best practices. Inform Syst Manag 23(3):19–30
11. Benbya H, McKelvey B (2006) Toward a complexity theory of information systems development. Inform Tech People 19(1):12–34
12. Kauffman S (1993) The origins of order: self-organization and selection in evolution. Oxford University Press, New York, NY
13. Kauffman S (1995) At home in the Universe. Oxford University Press, New York, NY
14. Holland JH (1995) Hidden order: how adaptation builds complexity. Addison-Wesley, Reading, MA
15. Holland JH (1998) Emergence: from chaos to order. Perseus Publishing, Cambridge, MA
16. Jacucci E, Hanseth O, Lyytinen K (2006) Taking complexity seriously in is research. Introduction to the special issue. Inform Tech People 19(1):5–11
17. Cilliers P (1998) Complexity and postmodernism: understanding complex systems. Routledge, London
18. Jain R, Meso P (2004) Theory of complex adaptive systems and Agile software development. In: Proceedings of the 10th Americas conference on information systems, New York, NY, pp 1661–1668
19. Cilliers P (2000) What can we learn from a theory of complexity? Emergence 2(1):23–33
20. Mitleton-Kelly E (2003) Ten principles of complexity and enabling infrastructures. In: Mitleton-Kelly E (ed) Complex systems and evolutionary perspectives on organisations: the application of complexity theory to organisations. Elsevier Science Ltd, Oxford, pp 3–20
21. Volberda HW, Lewin AY (2003) Guest editors' introduction coevolutionary dynamics within and between firms: from evolution to co-evolution. J Manag Stud 40(8):2111–2136
22. Brown S, Eisenhardt K (1998) Competing on the edge: strategy as structured chaos. Harvard Business School Press, Boston, MA
23. Anderson P (1999) Complexity theory and organization science. Organ Sci 10(3):216–232
24. Stacey RD (2003) Strategic management and organisational dynamics: the challenge of complexity, 4th edn. Financial Times, Prentice Hall, Harlow
25. McMillan E (2004) Complexity, organizations, and change. Routledge, London
26. Vidgen R, Wang X (2006) Organizing for agility: a complex adaptive systems perspective on Agile software development process. In: Ljunberg J, Andersson M (eds) Proceedings of the 14th European conference on information systems, Göteborg, Sweden, pp 1316–1327
27. Abrahamsson P, Warsta J, Siponen MT, Ronkainen J (2003) New directions on agile methods: a comparative analysis. In: Proceedings of the 25th IEEE international conference on software engineering, Portland, OR, pp 244–254
28. Boehm B, Turner R (2004) Balancing agility and discipline: evaluating and integrating agile and plan-driven methods. In: Proceedings of the 26th IEEE international conference on software engineering. Edinburgh, Scotland
29. Port D, Bui T (2009) Simulating mixed agile and plan-based requirements prioritization strategies: proof-of-concept and practical implications. Eur J Inform Syst 18(4):317–331
30. Kautz K, Madsen S, Nørbjerg J (2007) Persistent problems and practices in information system development. Inform Syst J 17(3):217–239
31. Boehm B, Turner R (2003) Using risk to balance agile and plan driven methods. Computer 36(6):57–66
32. Austin RD, Devin L (2009) Research commentary—weighing the benefits and costs of flexibility in making software: toward a contingency theory of the determinants of development process design. Inform Syst Res 20(2):462–477

33. Cockburn A (2001) Agile software development. Addison-Wesley, Boston, MA
34. Highsmith J (2002) Agile software development ecosystems. Addison-Wesley, Boston, MA
35. Coplien JO, Harrision NB (2005) Organizational patterns of agile software development. Prentice Hall, Upper Saddle River, NJ
36. Nerur S, Mahapatra R, Mangalaraj G (2005) Challenges of migrating to agile methodologies. Comm ACM 58(5):73–78
37. Schein E (2004) Organizational culture and leadership, 3rd edn. Wiley, San Francisco, CA
38. Creswell JW (2003) Research design—qualitative, quantitative and mixed methods approaches. Sage Publications, Thousand Oak, CA
39. Goldman S, Nagel R, Preis K (1995) Agile competitors and virtual organizations. Van Nostrand Reinhold, New York
40. Dove R (2001) Response ability: the language, structure and culture of the agile enterprise. Wiley, New York
41. Wang X, Vidgen R (2007) Order and chaos in software development: a comparison of two software development teams in a major IT company. In: Winter R et al (eds) Proceedings of the sixteenth European conference on information systems, St. Gallen, Switzerland, pp 807–818
42. Waldrop MM (1994) Complexity: the emerging science at the edge of chaos. Penguin, London
43. Langer EJ (1997) The power of mindful learning. Perseus Publishing, Cambridge, MA
44. Langer EJ (1992) Matters of mind: mindfulness/mindlessness in perspective. Conscious Cognit 1(3):289–305
45. Chanowitz B, Langer EJ (1980) Knowing more (or less) than you can show: understanding control through the mindlessness/mindfulness distinction. In: Seligman ME, Garber J (eds) Human helplessness. Academic, New York
46. Sternberg RJ (2000) Images of mindfulness. J Soc Issues 56(1):11–26
47. Weick K, Sutcliffe KH (2001) Managing the unexpected: assuring high performance in an age of complexity. Jossey-Bass, San Francisco, CA
48. Beck K, Fowler M (2001) Planning extreme programming, 2nd edn. Addison-Wesley, Boston, MA
49. Beck K, Andreas C (2004) Extreme programming explained: embrace change, 2nd edn. Addison-Wesley, Boston, MA
50. Kabat-Zinn J (1994) Where ever you go there you are—mindfulness meditation in everyday life. Hyperion, New York

The Role of Trust in the Initial Adoption of Identity Management Systems

Ali Alkhalifah and John D'Ambra

Abstract Trust plays a critical role in exchange relationships, including unknown risks in the online environment. With respect to the adoption of identity management systems (IdMS), users face similar situations and must rely on trust to overcome their risk perceptions because many online interactions require the user to disclose identity information. Previous studies showed that trust significantly influenced and increased the individual's behavioral intention toward a particular Web-based service or technology. However, researchers have not explored the role of trust in IdMS adoption. This study identifies multidimensional trust perceptions in the initial adoption stage of the IdMS and aims to understand and examine their affect toward user behavioral intentions to adopt IdMS. A research model of initial trust formation is developed and includes trusting beliefs (trust in IdMS providers and IdMS artifact) and institution-based trust (trust in the Internet). The conclusion of this paper outlines the implications and suggests further directions for future research in this area.

1 Introduction

As digital identity becomes more and more important in the online world, identity management is an essential component for the successful growth and development of the next, so-called "2.0," user-centric Internet services. Therefore, the emergence of identity management systems (IdMS) has brought about primary changes to online transactions. IdMS are information systems (IS) comprising the process, polices, and emerging technologies used to manage information about the identity of users and control access to online resources [18]. The factors affecting adoption

A. Alkhalifah (✉) • J. D'Ambra
University of New South Wales, Sydney, Australia
e-mail: a.alkhalifah@student.unsw.edu.au; j.dambra@unsw.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_2,

of an emerging information technology (IT) artifact such as IdMS have been of interest among IS researchers, but they remain unknown [21]. Although many studies have focused on the factors that influence the adoption of IT products (e.g., [4]) or online services such as online banking and online shopping for the past decade (e.g., [6, 21]), the issue associated with IdMS adoption has received little attention in the academic literature [34, 35]. There is a lack of in-depth published research that considers both use and context (from an IS perspective) related to the topic of IdMS [35]. Digital identity authentication schemes from the providers' perspective have been well documented [34]. However, recent research agrees that there is little exploration of the perception of IdMS from the user's perspective. Thus, further research to explore users' perceptions of IdMS is called for [1, 2, 15, 32, 34]. In addition, the provision of suitable frameworks or models to better understand IdMS from both business and user perspectives is equally limited [35]. Therefore, the objective of this study is to develop a model to provide a better understanding of the factors influencing IdMS adoption by users.

Recent trust studies have described trust as a fundamental construct for understanding user perceptions of technology and a primary predictor of technology usage [20, 21, 39]. Researchers suggest that understanding how initial trust toward an IT is formed is essential for promoting the adoption of a new IT [20]. Recent identity management researchers also suggested that the trust in IdMS should be investigated and formalized [23, 33, 35]. Previous identity management studies addressed trust issue from the technical perspective. Theoretical, social, and regulatory dimensions have not been well addressed in the literature [1]. Therefore, this research fills this gap and investigates user perceptions of trust and its influence on user adoption of IdMS.

There is a lack of theoretical and conceptual frameworks that guide IT artifact research [13]. Gefen et al. [13] argued that "Yet, we do not have a conceptual framework, nor taxonomy of constructs, to guide such research by identifying the constructs of importance and for deriving propositions" (p. 281). Therefore, they suggested developing a theory or a number of theories that identify the IT artifact constructs that have the potential to increase trust in online environment, and further specify how these constructs can be incorporated into technology adoption models. Little empirical evidence exists to support the nomological validity (construct validity) of applying trust to technological artifacts. Wang and Benbasat's [39] research is the first empirical study to examine trust in online recommendation agents. However, to the best of our knowledge, the nomological validity of trust in IdMS has not yet been empirically examined. Therefore, empirical testing is needed regarding whether or not trusting beliefs hold true for IdMS. This study is an attempt to fill this important gap and to contribute to the literature by proposing an integrated trust model to investigate and understand the initial trust toward user adoption of IdMS. This paper addresses the following questions:

- What are the antecedents of initial trust in the IdMS context?
- How does trust affect users' intention to adopt IdMS at the individual level?

This study theoretically contributes to IS literature relating to trust and emerging IT artifact adoption. Prior studies that examine trust on individual behavior have been limited and typically test it as a single construct [8, 19]. There is a lack of studies that investigate and test the trust factor as multidimensional constructs [21]. Therefore, this study contributes to the literature by examining users' beliefs about multidimensional trust constructs in IdMS context. It also develops a theory for better understanding the adoption and use of IdMS. The current study contributes to the literature by incorporating the integral roles of trust into innovative technology adoption. It extends the technology acceptance theory in the context of IdMS and provides a foundation for further research on user adoption of IdMS as, to the best of our knowledge, it is the first attempt to examine trust toward IdMS adoption from the user's perspective. According to Melone [43], "for the most part, the IS literature is silent on how users form initial attitudes about technologies and how these attitudes are modified over time" (p. 77). Since then, although a growing body of IT adoption research has examined formation of initial beliefs, to date, very little research has been directed at understanding initial adoption of IdMS. This study synthesizes and extends previous work of innovation adoption research into a specific context of IdMS. It attempts to contribute to a better theoretical understanding of the antecedents of user acceptance and user resistance to adoption and usage of an emerging class of IT artifact particularly IdMS. Furthermore, this study can pragmatically inform IdMS providers of how individuals interact with IdMS services and technologies so that providers can assess trust and reduce risk perception toward the adoption. Moreover, the proposed framework offers an increased understanding of user's trust, which in turn will provide designers with a tool that can be used to develop trust-building mechanisms and risk-reducing strategies that will encourage IdMS adoption.

The remainder of the paper is organized as follows. The next section provides an overview of IdMS and defines trust in this study. This is followed by the development of the model proposed in this study. Next, the employed research method is described. Finally, the conclusion and future directions of the study are presented.

2 Literature Review

2.1 Identity Management Systems

IdMS have been defined as the integration of important personal information from multiple systems into one collaborative and unique identity [28]. IdMS represent solutions that are employed to manage end-user authentication, access rights and restrictions, account profiles, and other attributes that provide an individual with more control over his/her identity information [29]. This study focuses on Web-based IdMS and defines IdMS as the business processes, polices, and emerging technologies for the creation, maintenance, and use of online identities across the Internet and within online service providers. IdMS are services available on the Web that enable users to create and manage their online identity. In the offline world, a person carries multiple forms of identification in his/her wallet, such as driver's license, health insurance card, credit cards, and affinity cards such as frequent flyer and loyalty cards. Similarly, IdMS enable individuals to create a number of digital cards which they use to identify themselves with Web services that accept them. If a user subscribes to an identity management service, they can access Web sites affiliated with the identity management service. The user can manage their identity information among various Web sites in an integrated way through this service. Examples of IdMS include Microsoft Passport, OpenID, Information Card (CardSpace), Facebook Connect, Web Single Sign-on, and PayPal Access.

Diverse parties participate in the IdMS in different ways. Their participation can be classified by roles, taking into consideration that any individual participant or set of participants can play multiple roles (both at the same time and at different times) [5]. These roles within the IdMS are the following:

- *Subjects*: users of digital services. Subjects may act on their own behalf (as individual citizens, customers), or in roles within organizations, companies, or government departments.
- *Identity providers (IdPs)*: issue identities. For example, individuals might use self-issued identities in contexts such as signing on to Web sites, credit card providers might issue identities that enable payment, businesses might issue identities to their customers, and governments might issue identities to citizens.
- *Relying parties (RPs) or service providers (SPs)*: an individual, organization, or service that depends on claims issued by a claims provider about a user to control the access to and personalization of services.

2.1.1 The Importance of Trust in Identity Management Systems

Trust plays a critical role in exchange relationships, including unknown risks in the online environment [12]. With respect to the adoption of IdMS, users face similar situations and must rely on trust to overcome their risk perceptions because many online interactions require the user to disclose identity information. Trust in IdMS is a behavioral belief related to the perception of security in using a particular IdMS technology. This may be guaranteed by a security IdP and a SP by using specific mechanisms such as encryption or digital signatures. Although IdMS is reliable and includes measurement against risks [32], potential risks may come from multiple sources, such as the vulnerability of Internet communication platforms and reasonable technical and operational safeguards of the IdP or a SP. Many identity management researchers suggested that the trust relationship among the parties (user, IdP, and SP) using IdMS should be investigated and formulized [1, 23, 33, 35].

2.2 Trust

2.2.1 Trust in Technological Artifacts and IdMS

In most previous studies, trust was initially advanced in the context of interpersonal relationships in which trust targets are human [13, 20, 39]. Most trust research has focused on virtual team members or a Web vendor, and therefore the trustee has been human or a group of humans [20]. The role and nature of trust in technological artifacts remain unclear [20, 21, 39]. In the technological artifact context, trust has been identified similarly to interpersonal trust in that it reveals the willingness of the truster to "behaviourally depend on a piece of software (e.g., a statistical system) to do a task" [25]. Recent IS trust research has been applied to the relationship between humans and technology and has identified trust in technology when the trustee is a technological artifact, such as a recommendation agent [39], an information system [20], and mobile banking [21]. Research findings have showed that components of trust in technological artifacts and in humans do not differ significantly. This indicates that individuals not only use technological artifacts as tools but also form trusting and social relationships with them [39]. Therefore, we identify trust in IdMS as an extension of interpersonal trust and a technological artifact that has been studied both in other contexts and in the recent literature on IS.

Trust in IdMS is different from the IT artifacts that have been examined in previous studies such as a recommendation agent [39] and online banking [21]. IdMS are much broader in nature. IdMS include not only the enabling technology but also the content, services, architecture, distributed environments, and even institutions. IdMS enable interactions and security of financial transactions, as well as the exchange of private and sensitive information among people, service providers, organizations, and institutions via the Internet. IdMS require three parties: the users, the IdMS provider, and the Web service (whatever it maybe which we access via the IdMS). Therefore, with an IdMS, users provide their identity details to the IdMS who will then maintain and manage the identity data of the user providing the user access to Web services. Therefore, with an IdMS, trust is placed with the provider of the IdMS, not the Web service (i.e., Internet banking, online shopping, credit card management); thus the relationship is different and therefore, the nature of trust is different. Users can access, control, and manage their digital identity from anywhere and at any time with complete freedom. Because of the uniqueness of IdMS, it is imperative to examine the trust and acceptance of this complex new technology from the users' perspective.

2.2.2 Definition of Trust

Trust has been explored extensively and defined differently in many research studies. Because of the varied definitions of this complex construct, we choose to use a well-referenced and respected definition of trust set forth in numerous studies. This study adopts Mayer et al.'s [24] definition of trust as "the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the truster, irrespective of the ability to monitor or control that other party" (p. 712). This study follows the call from Gefen et al. [12, 13] for further studies on trust in new IT artifacts.

The current study focuses on users' *initial trust belief*, which is formed after users have a first experience with specific system [45] or previous experience with other systems [20]. *Initial trust* refers to trust in an unfamiliar trustee [27]. The *truster* refers to the party that grants trust and the *trustee* refers to the party that receives trust [26]. In initial trust, the truster depends on other sources, such as secondhand information or contextual factors, in order to make trust inferences [20]. For example, before using a particular IdMS, a user can build initial trust in this system based on their experiences with the Internet or other Web-based technologies, such as online services or others' opinions about the system. Because IdMS are in the early stage, initial trust is needed in a relationship in which the user does not yet have meaningful or credible information about the technology [3, 27]. This study focuses on the initial adoption of IdMS and therefore aligns with the research on initial trust [20]. Because trust is a dynamic concept that develops over time, some researchers have noted the importance of studying initial trust, particularly in cases of novel technology such as IdMS, where users must overcome uncertainty and perceptions of risk before using the technology [13, 20, 21, 27, 39]. Moreover, all these studies showed that initial trust has a strong influence on individual behavioral intentions toward novel technologies. Therefore, an investigation of the initial trust circumstance will enhance our understanding of why users initially trust in the context of IdMS and IS.

Numerous researchers have examined the effect of trust on an individual's behavioral intention to use Web-based services and technology in different contexts, including the online market place [32], online services [8, 19, 41], e-commerce [12, 14, 22, 27], and mobile banking [21]. These studies showed that trust significantly influenced and increased the individual's behavioral intention toward a particular Web-based service or technology. However, researchers have not empirically explored the role of trust in IdMS adoption. Similar to the above studies, initial trust will be the important potential influencer to examine the initial adoption of IdMS.

This study follows McKnight et al. [27], Pavlou and Gefen [31], Cho [8], Be´langer and Carter [3], and Luo et al. [21] and suggests that trust is the belief that allows individuals to be willing to react after having taken the characteristics of the providers and the underlying Internet infrastructure into consideration. Therefore, the user should consider both the characteristics of the Web vendor and characteristics of the enabling technological infrastructure before using an electronic service [30]. Trust in IdMS is therefore derived from the traditional view of trust in a specific provider (trust of the identity and service provider) and trust in the reliability of the supporting technology (trust of the Internet) [3, 21, 30]. Overall, initial trust here includes trusting beliefs (trust in IdMS artifact and the IdMS providers) and institution-based trust (trust in the Internet).

3 Conceptual Model

According to the Theory of Social Responses to Computers [44], people unconsciously treat technological artifacts as social actors and apply social rules to them. Although there is an absence of more humanlike features, such as voice and

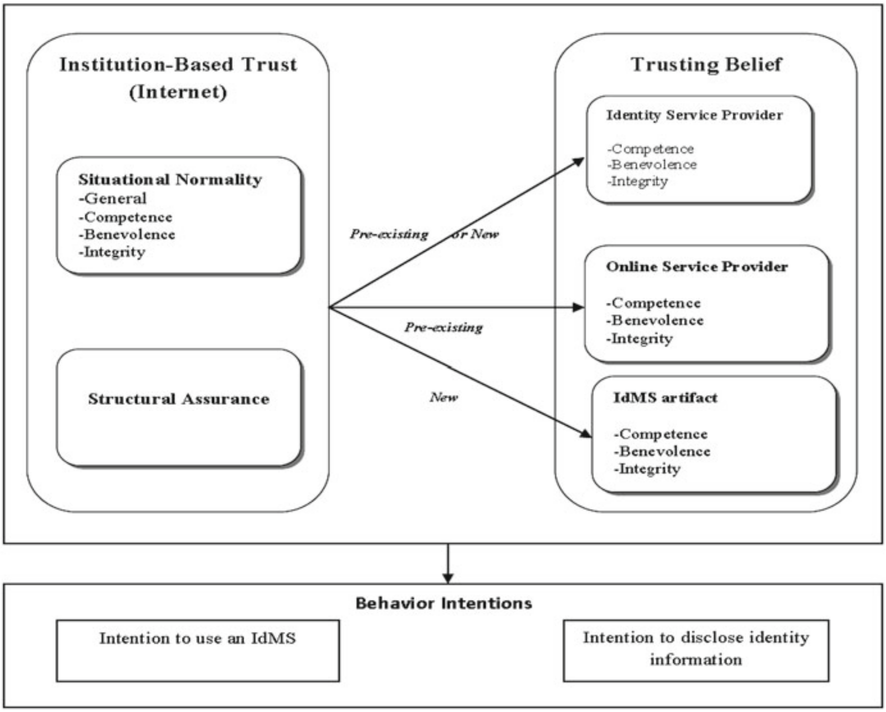


Fig. 1 Research model

embodiment, humans tend to anthropomorphize technology [44]. Consistent with this theory, Sztompka [36] argued that in the case of trust in a technological artifact, "we trust those who design the technology, those who operate them and those who supervise the operations" (p. 46). Hence, this study defines trust as the users' belief of the competence, benevolence, and integrity of the IdMS artifact, IdMS providers, and Internet platform. The initial trust model developed in this study is shown in Fig. 1. This model consists of three levels of constructs: (1) trusting beliefs, (2) institution-based trust, and (3) behavioral intentions to adopt IdMS. The research model distinguishes two types of trust: initial trust perception of IdMS artifact and an identity provider (may be a preexisting trust) and preexisting trust of service providers and Internet platform.

3.1 Trust Construct

Trust is often viewed as a three-dimensional construct consisting of competence, integrity, and benevolence [27]. Although many trust dimensions have been posited in the literature, these three trust beliefs have been well accepted in many recent studies [8, 21, 27, 39]. *Competence* refers to the ability of the trustee to do what the truster needs, *benevolence* is defined as trustee caring and motivation to act in the truster's interests, and *integrity* means trustee honesty and promise keeping [27]. As mentioned earlier, the current study focuses on *initial trust*, which develops after a user has a first experience with IdMS [45] or his/her experiences with other systems and other's opinions about the system [20]. These three trust beliefs apply to diverse temporal contexts, such as the initial stage of trust formation [16, 39]. Users may have concerns about the competence of IdMS providers to satisfy their needs, as well as concerns about whether they are working on their behalf rather than on the behalf of others. Trust can help users overcome these concerns and encourage them to adopt the IdMS. The benevolence of IdMS providers can be engendered by informing users that they care about user needs and therefore protect their identity information. Thus, their integrity can be promoted by providing technical ability.

3.2 Trust in Identity Management Systems Artifact and Providers

Trust in IdMS artifact and IdMS providers refers to trusting beliefs. The term *trusting beliefs* is defined as the confident truster perception that the trustee, such as a specific Web-based vendor, has attributes that are beneficial to the truster [27]. Trust in the IdMS providers refers to the user's perceptions regarding the integrity and ability of the providers providing the online identity service [3, 24, 27]. Be´langer and Carter [3] argued that citizen confidence in the ability of an e-government service provider to give e-services is crucial for the widespread adoption of e-government initiatives. Moreover, [12] hypothesized that trust in the agency has a strong affect on the adoption of a technology. Luo et al. [21] posited that trust in the bank has a significant influence on user acceptance of mobile banking. Similarly, users must believe that identity providers and e-service providers have the technical and astuteness resources necessary to protect and respect their identity information. Truthful and non-fraudulent interactions with identity service providers will enhance users' trust and adoption of IdMS.

Trust in the IdMS artifact refers to the user's perceptions regarding the integrity and ability of the IdMS providing managing and maintaining their online identities [25]. Previous trust studies have been showed that trust in technological artifacts has a positive effect toward user intention to adopt a new IT artifact, such as a recommendation agent [39] and mobile banking [21].

It is necessary to distinguish between *trust beliefs* about an online service provider and trust in the technology as a platform for online transactions [21, 27]. Trust in IdMS is different than other Web-based services and technologies. IdMS are not owned by the users, and there is a provider relationship between an IdP, a SP, and its users. Thus, the concept trust in this case is important and complicated, and it must be built in all IdMS roles (i.e., the user, IdP, and SP) because the IdMS enables mutual authentication [11, 17, 32]. In this research context, identity service provider, service provider, and IdMS artifact specific trust beliefs are one type of initial trust. However, they differ when users interact with an IdMS for the first time. This study focuses on factors that drive existing online services users to adopt IdMS. Therefore, trust in a SP is one type of preexisting belief. As an IdP is another party but could be a SP at the same time [5], trust in IdP includes both new and preexisting beliefs. Trust in an IdP is one type of preexisting belief when the user interacts with the IdP as a SP at the same time or the user has had an experience with IdP. An example of this case is Google Accounts. Google utilizes the OpenID protocol as an IdP and allows users to authenticate as a SP to use Google services and applications on behalf of the user identity (Google account) [37]. Thus, users can authenticate to any SP that uses the OpenID protocol by identifying their Google accounts as an OpenID. Conversely, trust in an IdP is one type of new belief when the user interacts with an IdP for the first time. For example, users may be uncomfortable linking their e-mail account to their bank [34]. The trust of an IdMS is a new belief when users interact with an IdMS service or technology for the first time (see Fig. 1).

3.3 Trust in the Internet

Trust of the Internet refers to institution-based trust [12, 27]. *Institution-based trust* refers to “the belief that needed structural conditions are present (e.g., in the Internet) to enhance the probability of achieving a successful outcome in an endeavour like e-commerce” [27, p. 339]. Institution-based trust is especially important in situations where there is no previous interaction or where the truster is not familiar with the trustee [20]. When studying trust in an Internet-based IT artifact such as IdMS, trust in the Internet as an institution should be considered. Trust of the Internet is consistently defined as a key predictor of Internet-based technology adoption [3, 21, 27, 30]. Trust in the Internet is based on trust in the security measures, safety nets, and performance structures of this electronic channel [3]. IdMS adoption is contingent upon users’ belief that the Internet is a dependable medium, capable of protecting identity information and securing transactions.

Trust in an IdMS platform (Internet) for managing online identity through online transactions is one type of preexisting trust. Institution-based (Internet) trust is defined as having two dimensions: situational normality and structural assurance. In *situational normality*, “one believes that the environment is in proper order and success is likely because the situation is normal or favorable” ([27] p. 339). For example, a user who perceives high situational normality would believe the Internet environment is suitable, well ordered, and favorable for disclosing identity information and using IdMS. In *structural assurance*, “one believes that structures like guarantees, regulations, promises, legal recourse, or other procedures are in place to promote success” ([27] p. 339). A user with high Internet-related structural assurance would believe that technological and legal Internet protections, such as data encryption, would safeguard him/her from loss of money, privacy, or sensitive information. In general, users would believe that IdMS providers on the Internet have the attributes of competence, benevolence, and integrity.

3.4 Trust and Behavior Intention

This study proposes *behavioral intention* as a substitute for actual behavior, and it establishes the importance of behavioral intention as a determinant of the user's adoption of IdMS. Behavioral intention is a measure of the strength of one's intention to perform a specific behavior [9]. Because using IdMS requires users to disclose their personal information in order to manage and control them, both intention to disclose and intention to use must be considered. Thus, we define behavioral intention as *a user's intending to disclose his/her identity information and to use IdMS*.

The extant literature on trust showed that trust belief is a key determinant of people's behavioral intentions to conduct activities involving risk [21]. In addition, trust belief is a fundamental construct in understanding users' perceptions of technology and overcoming perception of risk [20, 31]. Gefen et al. [12] found that trust belief increased customers' behavioral intention to buy CDs and books from an online vendor. Malhotra et al. [22] showed that trust belief has a significant influence on the intention to release personal information through the Internet. Luo et al. [21] showed that trust beliefs about a bank have a significant influence on consumers' behavioral intention to adopt mobile banking. Be´langer and Carter [3] proposed a model of e-government trust involving disposition to trust, trust of the Internet, trust of the government, and perceived risk. The results of their citizen survey indicated that disposition to trust positively impacts trust of both the Internet and the government, which in turn affect intentions to use an e-government service. Based on cognitive dissonance theory [42], if the Internet environment itself is perceived as dangerous, the individual may be less willing to disclose his/her identity information in that environment regardless of the trust he/she has in a particular provider. Similarly, in the context of IdMS, users with favorable trust perceptions toward the Internet and IdMS providers are more likely to disclose their identity information and to use IdMS. Therefore, we hypothesize that trust belief toward IdMS providers, IdMS artifact, and Internet positively affects users' behavioral intentions to adopt an identity management system technology or services.

4 Research Methodology

The research instrument to measure the constructs of the model was developed by adapting existing measures to the research context. Items that will be used in the final instrument are shown in Appendix. All trust construct measures were adopted from McKnight et al. [27]. Intention-to-use measures were adopted from Davis [9] and Venkatesh et al. [38]. Intention-to-disclose identity information measures were adopted from Be´langer and Carter [3] and Fogel and Nehmad [10]. All the variables will be measured on a 7-point Likert scale.

The data collection process will be carried out using an *online survey* method, and the questionnaire will be *Web based*. The participants will be users of Facebook and LinkedIn as well as people who engage in online shopping and use online payment services. The targeted participants were selected because they have experience with online identity services where individuals create online profiles, make connections with others, and share identity information. It is decided to collect data from the aforementioned target population to ensure that there would be a sufficient number of potential IdMS adopters in the sample. After a survey pretest stage is conducted, participants will be recruited via advertisements on Facebook and LinkedIn. Google AdWords will also be used to advertise the survey to recruit respondents who engage in online shopping and use online payment services.

5 Conclusion

The objective of this study was to develop a model of how multidimensional trust perceptions, at the individual level, affect the adoption of emerging IT artifacts in terms of IdMS. Such a model has not been suggested in past literature. It identified the initial trust roles (trust in IdMS providers, trust in IdMS artifact, and trust in the Internet) and explained how they may affect user behavioral intentions to adopt IdMS. The proposed model hopefully would stimulate further research and would provide a useful lens for examining trust in the context of acceptance of emerging innovative technology in its early adoption stage.

This study will be complemented by conducting a quantitative examination in order to validate the framework by empirical measurement and will test the causal network of correlation in the model using survey instruments. Factor analysis will be performed to evaluate dimensionality and discriminant validity. The data will be analyzed using the well-established structural equation modeling (SEM) technique – partial least squares (PLS) – to examine relations between the constructs of the conceptual model and to assess the overall fit of the structural model [7, 40].

6 Appendix: Research Instrument

6.1 Trust in IdMS Providers (Services Providers and Identity Providers)

1. IdMS providers are/would be competent and effective in managing my online identity.
2. IdMS providers perform/would perform its role of managing my online identity very well.
3. Overall, IdMS providers are/would be a capable and proficient Internet online identity provider.
4. In general, IdMS providers are/would be very knowledgeable about the online identity.

5. IdMS providers are/would be truthful in its dealings with me.
6. I characterize/would characterize IdMS providers as honest.
7. IdMS providers keep/would keep its commitments.
8. IdMS providers are/would be sincere and genuine.
9. I believe that IdMS providers act/would act in my best interest.
10. If I required help, IdMS providers do/would do its best to help me.
11. IdMS providers are/would be interested in my well-being, not just its own.

6.2 Trust in IdMS Artifact

1. An IdMS would be competent and effective in managing my online identity.
2. An IdMS would perform its role of managing my online identity very well.
3. Overall, an IdMS would be a capable and proficient Internet online identity service.
4. In general, an IdMS would be an intelligent solution for managing my online identity.
5. An IdMS would be truthful in its dealings with me.
6. I would characterize an IdMS as honest.
7. An IdMS would keep its commitments.
8. An IdMS would be sincere and genuine.
9. I believe that an IdMS would act in my best interest.
10. If I required help, an IdMS would do its best to help me.
11. An IdMS would be interested in my well-being, not just its own.

6.3 Trust in the Internet

6.3.1 Situational Normality

1. I feel good about how things go when I do purchasing or other activities on the Internet.
2. I am comfortable making online transactions on the Internet.
3. I feel that most Internet providers would act in a customers' best interest.
4. If a customer required help, most Internet providers would do their best to help.
5. Most Internet providers are interested in customer's well-being, not just their own well-being.
6. I am comfortable relying on Internet providers to meet their obligations.
7. I feel fine doing different transactions including business on the Internet since Internet providers generally fulfill their agreements.
8. I always feel confident that I can rely on Internet providers to do their part when I interact with them.
9. In general, most Internet providers are competent at serving their customers.
10. Most Internet providers do a capable job at meeting customer needs.
11. I feel that most Internet providers are good at what they do.

6.3.2 Structural Assurance

1. The Internet has enough safeguards to make me feel comfortable using it to perform different transactions.
2. I feel assured that legal and technological structures adequately protect me from problems on the Internet.
3. I feel confident that encryption and other technological advances on the Internet make it safe for me to do transaction there.
4. In general, the Internet is now a robust and safe environment in which to perform online transactions.

6.4 Behavioral Intentions

6.4.1 Intention to Use

1. I will definitely consider using an IdMS.
2. I predict I would use an IdMS to manage my online identity.
3. I am willing to use an IdMS in the future.

6.4.2 Intention to Disclose Identity Information

1. I would not hesitate to provide my identity information to an online service provider.
2. It is important to me to protect my online identity.
3. I am concerned with the consequences of sharing my identity information online.
4. I am likely to share my identity information online in the future.

References

1. Adjei KJ, Olesen H (2011) Keeping identity private-establishing trust in the physical and digital world for identity management systems. Vehicular Tech Mag IEEE 6(3):70–79
2. Aichholzer G, Straub S (2010) National electronic identity management: the challenge of a citizen-centric approach beyond technical design. Int J Adv Intell Syst 3(1–2):12–23
3. Be´langer F, Carter L (2008) Trust and risk in e-government adoption. J Strategic Inf Syst 17(2):165–176
4. Behrenda TS, Wiebeb EN, Londonb JE, Johnsonc EC (2011) Cloud computing adoption and usage in community colleges. Behav Inform Technol 30(2):231–240
5. Cameron K, Posch R, Rannenberg K. (2009). Proposal for a common identity framework: a user-centric identity metasystem. In: Rannenberg K, Royer D, Deuker A (eds) The future of identity in the information society. Springer, Berlin, Germany, pp 477–500
6. Cheng TC, Lam D, Yeung A (2006) Adoption of internet banking: an empirical study in Hong Kong. Decis Support Syst 42(3):1558–1572

7. Chin WW (1998) The partial least squares approach for structural equation modeling. In: Marcoulides GA (ed) Modern methods for business research. Lawrence Erlbaum Associates, Mahwah, NJ, pp 295–336
8. Cho V (2006) A study of the roles of trusts and risks in information-oriented online legal services using an integrated model. Inf Manage 43(4):502–520
9. Davis FD (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quart 13(3):319–340
10. Fogel J, Nehmad E (2009) Internet social network communities: risk taking, trust, and privacy concerns. Comput Hum Behav 25(1):153–160
11. Friedman A, Crowley P, West D (2011). Online identity and consumer trust: assessing online risk: The Brookings Institution. http://www.brookings.edu/~/media/Files/rc/papers/2011/0111_online_identity_trust/0111_online_identity_trust.pdf. Accessed 1 Mar 2012
12. Gefen D, Karahanna E, Straub DW (2003) Trust and TAM in online shopping: an integrated model. MIS Quart 27(1):51–90
13. Gefen D, Benbasat I, Pavlou PA (2008) A research agenda for trust in online environments. J Manage Inform Syst 24(4):275–286
14. Ha S, Stoel L (2009) Consumer e-shopping acceptance: antecedents in a technology acceptance model. J Bus Res 62(5):565–571
15. Ivy K, Conger S, Landry BJ (2010). 30P. Federated identity management: why is adoption so low? Paper presented at the international conference on information resources management. Jamaica
16. Koufaris M, Hampton-Sosa W (2004) The development of initial trust in an online company by new customers. Inf Manage 41:377–397
17. Landau S, Moore T (2011) Economic tussles in federated identity management. http://privacyink.org/pdf/weis11identity.pdf. Accessed 10 Aug 2011
18. Lee SC (2003) An Introduction to identity management: SANS Institute InfoSec reading room. http://www.sans.org/reading_room/whitepapers/authentication/introduction-identity-management_852 Accessed 13 May 2011
19. Lee M (2009) Predicting and explaining the adoption of online trading: an empirical study in Taiwan. Decis Support Syst 47(2):133–142
20. Li X, Hess TJ, Valacich JS (2008) Why do we trust new technology? A study of initial trust formation with organizational information systems. J Strategic Inf Syst 17:39–71
21. Luo X, Li H, Zhang J, Shim JP (2010) Examining multi-dimensional trust and multi-faceted risk in initial acceptance of emerging technologies: an empirical study of mobile banking services. Decis Support Syst 49(2):222–234
22. Malhotra NK, Kim SS, Agarwal J (2004) Internet users' information privacy concerns (IUIPC): the construct, the scale, and a causal model. Inform Syst Res 15(4):336–355
23. Marmol FG, Girao J, Perez GM (2010) TRIMS, a privacy-aware trust and reputation model for identity management systems. Comput Networks 54(16):2899–2912
24. Mayer R, Davis J, Schoorman D (1995) An integrative model of organizational trust. Acad Manage Rev 20(3):709–734
25. McKnight DH (2005) Trust in information technology. In: Davis GB (ed) The Blackwell encyclopedia of management information systems, vol 7. Blackwell, Malden, MA, pp 329–331
26. McKnight DH, Cummings LL, Chervany NL (1998) Initial trust formation in new organizational relationships. Acad Manage Rev 23(3):473–490
27. McKnight DH, Choudhury V, Kacmar C (2002) Developing and validating trust measures for e-commerce: an integrative typology. Inform Syst Res 13(3):334–359
28. Meints M, Zwingelberg H (2009) D3.17: Identity management systems—recent developments: FIDIS. http://www.fidis.net/fileadmin/fidis/deliverables/new_deliverables3/fidis-wp3-del3.17_Identity_Management_Systems-recent_developments-final.pdf. Accessed 5 May 2011
29. Mont MC, Bramhall P, Pato J (2003) On adaptive identity management: the next generation of identity management technologies. HP Laboratories, Bristol
30. Pavlou PA (2003) Consumer acceptance of electronic commerce: integrating trust and risk with the technology acceptance model. Int J Electron Comm 7(3):101–134

31. Pavlou P, Gafen D (2004) Building effective online marketplaces with institution-based trust. Inform Syst Res 15(1):37–59
32. Poetzsch S, Meints M, Priem B, Leenes R, Husseiki R (eds) (2009). D3.12: Federated identity management—what's in it for the citizen/customer?: FIDIS. http://www.fidis.net/fileadmin/fidis/deliverables/new_deliverables/fidis-wp3-del3.12.Federated_Identity_Management.pdf. Accessed 15 Dec 2011
33. Roussos G, Peterson D, Patel U (2003) Mobile identity management: an enacted view. Int J Electron Comm 8(1):81–100
34. Satchell C, Shanks G, Howard S, Murphy J (2011) Identity crisis: user perspectives on multiplicity and control in federated identity management. Behav Inform Technol 30(1):51–62
35. Seltsikas P, O'Keefe RM (2010) Expectations and outcomes in electronic identity management: the role of trust and public value. Eur J Inform Syst 19(1):93–103
36. Sztompka P (1999) Trust: a sociological theory. Cambridge University Press, Cambridge
37. Valkenburg P, Meijers W, Lycklama D, Jansen V (2010) E-identity as a business: case studies and lessons learned in networked identity. Innopay, Amsterdam, The Netherlands
38. Venkatesh V, Morris MG, Davis GB, Davis FD (2003) User acceptance of information technology: toward a unified view. MIS Quart 27(3):425–478
39. Wang W, Benbasat I (2005) Trust in and adoption of online recommendation agents. J Assoc Inf Syst 6(3):72–101
40. Wetzels M, Odekerken-Schroder G, Oppen C v (2009) Using PLS path modeling for assessing hierarchical construct models: guidelines and empirical illustration. MIS Quart 33(1):177–195
41. Bansal G, Zahedi FM, Gefen D (2010) The impact of personal dispositions on information sensitivity, privacy concern and trust in disclosing health information online. Decision Support Systems 49(2):138–150
42. Festinger LA (1957) A theory of cognitive dissonance. Row, Peterson, Evanston, IL
43. Melone N (1990) A Theoretical Assessment of the User-Satisfaction Construct in Information Systems Research. Manage Sci 36(1):76–91
44. Nass C, Moon Y, Fogg BJ, Reeves B, Dryer DC (1995) Can computer personalities be human personalities? Int J Hum-Comput Stud 43(2):223–239
45. Wu I, Chen J-L (2005) An extension of Trust and TAM model with TPB in the initial adoption of on-line tax: An empirical study. Int J Hum-Comput Stud 62(6):784–808

A Conceptual Framework for Aligning Business and Information Systems from Organisational Learning Perspective

Hamad Balhareth and Kecheng Liu

Abstract The three decades of ongoing executives' concerns of how to achieve successful alignment between business and information technology shows the complexity of such a vital process. Most of the challenges of alignment are related to knowledge and organisational change, and several researchers have introduced a number of mechanisms to address some of these challenges. However, these mechanisms pay less attention to multilevel effects, which results in a limited understanding of alignment across levels. Therefore, we reviewed these challenges from a multilevel learning perspective and found that business and IT alignment is related to the balance of exploitation and exploration strategies with the intellectual content of individual, group and organisational levels.

1 Introduction

The challenges of aligning business and information systems have been identified in numerous studies; however, alignment continues to be one of the top five concerns of CIOs [18]. These challenges have been summarised into lack of knowledge among executives, unawareness of alignment importance, unknown strategy and lack of industry knowledge and organisational change [6]. According to Lee and Bai [19], these challenges are directly linked to the strategic alignment of business and information systems failure, which raises the need for an organisational learning mechanism. Fewer studies have introduced a mechanism at individual alignment [34], group alignment [31] and organisational alignment [12]. However, most of the proposed mechanisms have focused on one alignment level (see Table 1), which gives a contrary view on the nature of alignment multilevel effects [4].

H. Balhareth (✉) • K. Liu
Business Informatics, Systems and Accounting,
University of Reading, Reading, RG6 6UD, UK
e-mail: h.h.balhareth@pgr.reading.ac.uk; k.liu@reading.ac.uk

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_3,

Table 1 Sample of published mechanisms in business–IT alignment

Alignment	Source: Henderson and Venkatraman [11]	Reich and Benbasat [31]	Sabherwal and Chan [33]	Tan and Gallupe [34]	Lee et al. [20]
Mechanism	Single loop and double loop	Social dimension and intellectual dimension	Strategy process and strategy content	Extension of social dimension and intellectual dimension	Social dimension and technical dimension
Focus level	Organisation	Group	Organisation	Individual and group	Group and organisation

The dynamic nature of achieving alignment requires the ongoing process of organisational learning to ensure continuous alignment between business and information systems [16,21].

Therefore, in order to understand alignment mechanisms from a multilevel perspective, a framework is needed to describe the flow of knowledge creation, knowledge interpretation, knowledge integration and knowledge utilisation across the intellectual content of individual, group and organisational levels [35]. The suggestions emphasise the need for learning forms that distinguish between utilising existing knowledge and assembling a new knowledge [23,28].

This paper therefore aims to introduce a comprehensive framework that takes into account alignment from a multilevel learning perspective with exploration (assembling a new knowledge) and exploitation (using existing knowledge) concepts.

2 Background

2.1 Strategy and Structure Mode

Research on alignment has witnessed changes that go beyond automation which begun by moving alignment research forward to strategic alignment between business and IS [15]. The early motivation for planning was categorised as a top-down planning where IT strategy has to be associated with business strategy [6]. Other researchers found that strategic alignment requires structural alignment between top management and IT management [30]. Structural alignment refers to the importance of structural harmony between business and IT, particularly in IT decision-making rights, deployment of IS personnel and the reporting relationship [28].

In the early 1990s, Morton [25] developed the MIT90s framework of alignment, which influenced a number of researchers such as Henderson and Venkatraman [12], MacDonald [22] and Baets [3] in creating their models [6]. However, the strategic alignment model (SAM) has been supported empirically through different studies

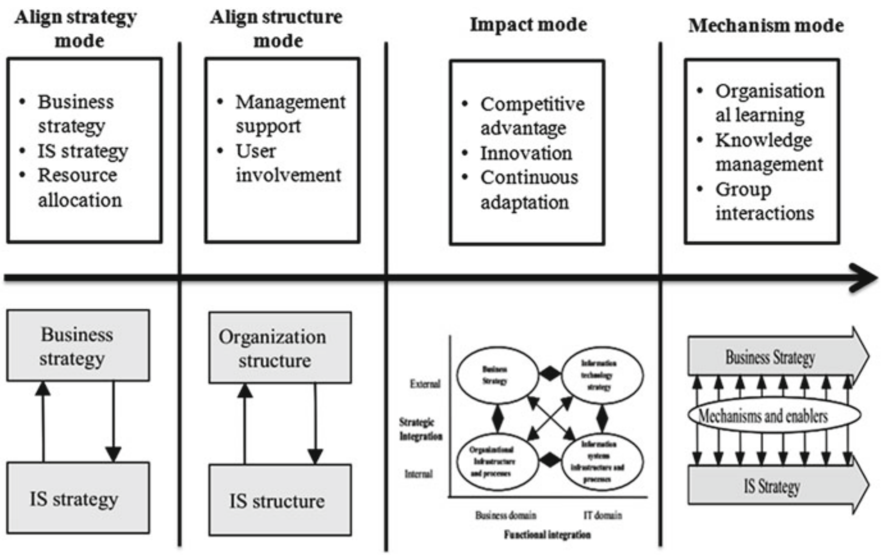


Fig. 1 Evolution of strategic alignment between business and information systems (adapted from refs. [19,28])

and shown to have a practical value (e.g. [2]). According to Henderson and Venkatraman [12], SAM helps to describe the value of emerging IT and create theoretical frameworks and guide managers to a practical understanding of the relationship between strategic and functional levels. However, the fundamental premises of this strategic alignment model do not consider learning as central to planning which decreases, therefore understating the opportunity and utilisation of IS. In recent years, studies have focused on mechanisms and enablers to further enhance collaboration and group interaction among business and IT (see Fig. 1). The next section describes a number of mechanisms that have been attempted by other research studies.

2.2 *Alignment Mechanisms and Enablers*

Many business executives realise that some business successes are the result of accidental circumstances, which promotes the need for more control in the future [7,16]. Luftman et al. [21] conducted a study for almost 5 years and identified enablers and inhibitors for business–IT alignment and referred to the complex nature of addressing both. The question of how to achieve alignment was and still is a concern for alignment research. Therefore, understanding strategic alignment from an organisational context has encouraged researchers to use strategic management and organisational learning perspectives (see Table 1). Henderson and Venkatraman [11] used single loop and double loop as mechanisms to describe strategic IS planning [1].

Reich and Benbasat [31] combined business–IT alignment with Horovitz's [13] duality of social dimension and intellectual dimension to explain alignment mechanisms and argue that the combination aims to reveal the complexity of alignment. Sabherwal and Chan [33] extend strategic alignment research through strategy process and strategy content, which correspond with social and intellectual dimensions of alignment. Strategy process and content were built based on strategic management literature and work in contrast to the single-loop and double-loop mechanism. Although extensive research has been carried out on mechanisms of alignment, no single study exists which adequately covers alignment from a multilevel perspective. In the following section, we highlight the alignment level of each mechanism.

2.3 Alignment Levels

Ideally, the planning content in business and IT alignment should be further expanded to encompass individual, group and organisation [6]. The constant change of business environment has raised the needs for dynamic adaption of organisations. Therefore, the process of creating knowledge from individual, group and organisation becomes a central for strategy creation [36] as opposed to a process of conception [37]. For this reason, organisation learning has received increasingly more attention for strategy research [8]. Strategic alignment is an ongoing process [12] that needs continuous learning processes across all organisational levels. Notwithstanding of apparent multilevel perspective in related literature, alignment research still focused on one or two alignment levels with less attention to effects of other levels which creates unsustainable harmony.

The main concern is understanding the complex nature of strategic alignment [27]. Consequently, organisational learning can be used as a mechanism to overcome the shortcomings of alignment research. This mechanism defines strategy from multilevel perspective, which describes individual, group and organisation as intellectual content and their linkage with the learning process.

3 Organisational Emphasis on Organisational Learning

Organisational learning has been increasingly acknowledged as a dynamic perspective for many disciplines such as strategic management, business and innovation [26]. The dynamic concept and multilevel perspective have given other studies a unique way to understand the nature of organisation. Crossan et al. [9] have reviewed a number of applied propositions in organisational learning literature and argue that learning is a multilevel perspective. The learning levels encompass individual learning which leads to group learning, and become translated at organisational level. Organisational learning conceives exploration–exploitation perspectives as a learning flow which is linked to the intellectual levels [24].

According to Crossan et al. [9], the learning process begins with knowledge creation which is situated within the individual level where individuals' experience and their capability determine the way tasks are carried out. The second step in the learning process is knowledge interpretation which involves other individuals by communication and dialogue. The transformation from implicit to explicit occurs at this step where ideas become clear and incorporable among individuals. The third step is knowledge sharing which is located at group level. The main concern is developing a mutual understanding between individuals. The final step of the learning process transfers individual learning and group learning towards organisation level by ensuring that tasks are defined, actions are established and learning is embedded and utilised in organisation strategies, structures and systems.

4 Organisational Learning as a Mechanism for Aligning Business and Information Systems

As described earlier in this paper, business–IT alignment research has used a number of mechanisms that were taken from strategic management literature (e.g. [10,13]) to understand how to achieve alignment (see Table 1). However, most of these studies do not explicitly show how strategy process shapes strategy content from a multilevel perspective and how content is sensitive to strategy process [17]. Therefore, we aim to further extend the research to understand the intertwined relationship between learning flow and intellectual content at multilevels from an organisational learning perspective.

As stated by Huysman et al. [14], the process of organisational learning can play a central role in strategic IS planning. The learning flow aims for a better understanding of IS/IT in term of opportunity and utilisation. Hence, the learning perspective must be considered in IS/IT strategic planning [38]. According to Lee and Bai [19], strategy creation is a learning process in which strategists themselves are learning at multilevels. In other words, the question "how to create a strategic alignment" is addressed by the learning flow. Then again, the intellectual content is concerned with planning methodologies, which reflect peoples' experience and knowledge [16,32]. This definition of intellectual content corresponds with the definition of Crossan et al. [9] for learning stock with the addition of a multilevel perspective that includes individual learning, group learning and organisational learning.

The decisions and actions are constantly adapted to meet external and internal changes. This dynamic nature of alignment needs a mechanism that balances short-term and long-term visions. March [24] proposed an exploration and exploitation perspectives of organisational learning which links organisation adaptation in alignment with its population. Exploration is concerned with a long-term vision that goes beyond utilising the current knowledge through assembling a new learning and discovering opportunities. In contrast, exploitation aims for short-term goals by using the current knowledge and capabilities. Organisational learning links learning flow and intellectual content as a dynamic mechanism and creates a value for strategic alignment [8,28].

5 A Conceptual Framework Based on Multilevel Perspective

As suggested, alignment should be studied from a dynamic perspective; we therefore developed a framework that considers the following premises: learning flow of alignment is an exploration and exploitation process; the intellectual content of alignment involves individual level, group level and organisational level; and the social process of alignment includes knowledge creation, knowledge interpretation, knowledge sharing and knowledge utilisation. These assume that a multilevel perspective is strongly associated to organisational performance based on other research findings (e.g. [5]).

As the challenges of alignment are often related to knowledge [6], the framework is developed on a multilevel perspective to understand how knowledge transfers across business and IT members. Exploration begins to assemble a new knowledge from individual to group and becomes embedded at the organisational level. According to Nonaka and Takeuchi [26], knowledge is created by individuals. The first attempt towards individual alignment was by Tan and Gallupe [34] who conducted research at the individual level and indicated that shared cognition of business and IT executives has a significant influence on business–IT alignment. Other scholars have included shared language and experience and found a positive relationship to executives' understanding at an individual level [29].

The individual knowledge has to be shared between business and IT members in order to be group knowledge. The group level is concerned about shared understanding and mutual adjustment [9]. Reich and Benbasat [31] focus on shared understanding of business and IT which is located at group level. The research findings suggest that shared domain knowledge has the strongest impact on alignment between business and IT executives.

The individual and group knowledge becomes crystallised at the organisational level which includes strategy, systems and processes [9]. Most studies on alignment focus on the organisational level with less attention to other alignment levels [3,12]. Exploitation refers to what has already been embedded at organisational level which becomes utilisable as existing knowledge from the organisation towards group and individual levels. Hence, in order to improve organisational performance, both strategic learning perspectives have to be balanced (see Fig. 2).

6 Discussion

The distinction suggests that the strategy process and social dimension were drawn into alignment research to understand how to achieve alignment, while the strategy content and intellectual dimension aim to specify what achieves alignment. Initial steps towards considering short-term and long-term influence on alignment was introduced by Reich and Benbasat [31] who focus only on shared

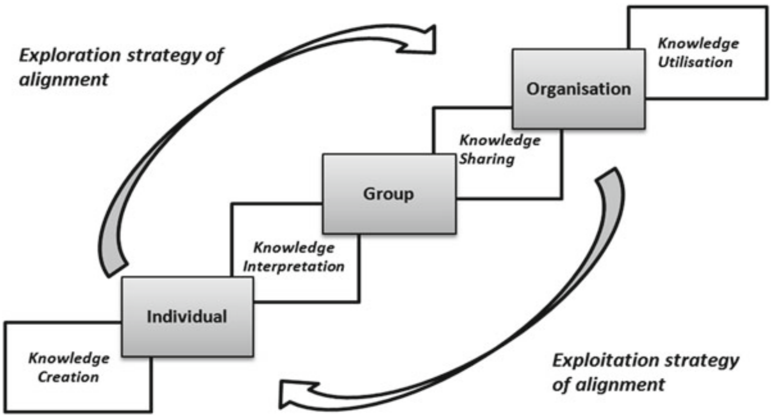


Fig. 2 A conceptual framework for aligning business and information systems from organisational learning perspective (adapted from ref. [9])

understanding of business and IT objectives. However, this is inadequate as business and IT have other concerns such as creating shared understanding and translating that understanding into strategic alignment. From an organisational learning perspective, exploration and exploitation represent a learning flow. Both exploration and exploitation have dissimilar modes under the social construct of learning and are essential for organisational survival. Therefore, we are further expanding research on the learning flow of strategic alignment from a deterministic perspective to an organisational learning perspective. The exploration strategy of alignment aims for new alternatives that enable the replacement of existing strategies, technologies and knowledge. This perspective is a distant in time and risky but expected to be higher in performance. For example, the focus of this perspective is on developing a new product and services. Whereas the exploitation strategy of alignment focuses on utilising existing technologies and knowledge, which in turn improve capabilities and skills, the exploitation concept in alignment emphasises, for example, the need for reducing costs, process re-engineering and adopting standardised procedures.

The intellectual content refers to knowledge that resides at the individual level, group level and organisational level [5]. The conceptual precision in the framework is created by the learning flow, and intellectual content aims to distinguish between knowledge within a level and knowledge that is transferred across levels. Any misalignment between learning flow and intellectual content may cause an effect on performance. For example, individuals might become reluctant to introduce their knowledge if the flow of learning flow is lower than the intellectual content of the individual. Thus, aligning learning flow and intellectual content is a vital approach for organisational performance.

7 Conclusion

This paper goes beyond the current concept of alignment by introducing a comprehensive mechanism that recognises the need for balancing alignment. The balance concept of exploration and exploitation overcomes environmental uncertainty and creates a strategic value. The extension of this research unifies the concept of learning flow and intellectual content and describes how process shapes the content of individual, group and organisation. The intellectual content reflects the current knowledge on what creates a strategic alignment, while the exploration and exploitation forms reflect knowledge flow that is concerned with how to create a strategic alignment. This research shows that multi-learning levels are critical for aligning business and information systems and leads to organisational performance.

Currently, we are in the process of conducting a case study to validate the framework. The analysis will include the impact of multilevel towards strategic alignment and exploration–exploitation concepts influence on performance.

References

1. Argyris C (1977) Double loop learning in organizations. Harvard Bus Rev 55(5):115–125
2. Avison D, Jones J et al (2004) Using and validating the strategic alignment model. J Strategic Inf Syst 13(3):223–246
3. Baets W (1992) Aligning information systems with business strategy. J Strategic Inf Syst 1(4):205–213
4. Benbya H, McKelvey B (2006) Using coevolutionary and complexity theories to improve IS alignment: a multi-level approach. J Inform Technol 21(4):284–298
5. Bontis N, Crossan MM et al (2002) Managing an organizational learning system by aligning stocks and flows. J Manage Stud 39(4):437–469
6. Chan YE, Reich BH (2007) IT alignment: what have we learned? J Inform Technol 22(4):297–315
7. Ciborra C (1994) The grassroots of IT and strategy. In: Ciborra C, Jelessi T (eds) Strategic information systems: a European perspective. John Wiley & Sons, Chichester, pp 3–24
8. Crossan MM, Berdrow I (2003) Organizational learning and strategic renewal. Strategic Manage J 24(11):1087–1105
9. Crossan MM, Lane HW et al (1999) An organizational learning framework: from intuition to institution. Acad Manage Rev 522–537
10. Earl MJ (1989) Management strategies for information technology. Prentice Hall, Upper Saddle River, NJ, USA
11. Henderson JC, Venkatraman N (1989) Strategic alignment: a framework for strategic information technology management
12. Henderson JC, Venkatraman N (1993) Strategic alignment: leveraging information technology for transforming organizations. IBM Syst J 32(1):4–16
13. Horovitz J (1984) New perspectives on strategic management. J Bus Strat 19–33
14. Huysman MH, Fisher SJ, Heng MS (1994) An organizational learning perspective on information systems planning. J Strategic Inf Syst 3:165–177
15. IBM Corporation (1981) Business systems planning–information systems planning guide, GE20-0527-3
16. Kearns G, Lederer A (2001) Strategic IT alignment: a model for competitive advantage. Proceedings of 22nd international conference on information systems (ICIS). New Orealns, USA

17. Ketchen DJ, Shook CL (1996) The application of cluster analysis in strategic management research: an analysis and critique. Strategic Manage J 17(6):441–445
18. Lawinski J (2011) "Business, IT alignment a top priority for CIOs", CIO insight. http://www.cioinsight.com/c/a/IT-Management/Business-IT-Alignment-a-Top-Priority-for-CIOs-547064/. Accessed 3 May 2012
19. Lee GG, Bai RJ (2003) Organizational mechanisms for successful IS/IT strategic planning in the digital era. Manage Decis 41(1):32–42
20. Lee SM, Kim K et al (2008) Developing a socio-technical framework for business-IT alignment. Ind Manage Data Syst 108(9):1167–1181
21. Luftman J, Papp R et al (1999) Enablers and inhibitors of business-IT alignment. Commun AIS 1(11)
22. MacDonald H (1991) The strategic alignment process. In: Morton S, Michael S (eds) The corporation of the 1990s: information technology and organizational transformation, 1st edn. Oxford Press, London, pp 310–322
23. Maes K, De Haes S et al (2011) How IT enabled investments bring value to the business: a literature review, 44th Hawaii International Conference on Systems Science, pp.1, 10
24. March JG (1991) Exploration and exploitation in organizational learning. Organ Sci 2:71–87
25. Morton MSS (1991) The corporation of the 1990s: information technology and organizational transformation. Oxford University Press, Cary, NC
26. Nonaka I, Takeuchi H (1995) The knowledge-creating company: how Japanese companies create the dynamics of innovation. Oxford University Press, Cary, NC
27. Papp R (2004) Assessing strategic alignment in real time. J Inform Educ Res 6(1):11–28
28. Peppard J, Breu K (2003) Beyond alignment – a co-evolutionary view of the IS strategy process. In: Proceedings of 24th international conference on information systems (ICIS), Seattle, WA, Dec 2003, pp 743–750
29. Preston DS, Karahanna E (2009) Antecedents of IS strategic alignment: a nomological network. Inform Syst Res 20(2):159–179
30. Pyburn PJ (1983) Linking the MIS plan with corporate strategy: an exploratory study MIS Quart 7(2):1–14
31. Reich B, Benbasat I (1996) Measuring the linkage between business and information technology objectives. MIS Quart 20(1):55–81
32. Reich B, Benbasat I (2000) Factors that influence the social dimension of alignment between business and information technology objectives. MIS Quart 24(1):81–113
33. Sabherwal R, Chan YE (2001) Alignment between business and IS strategies: a study of prospectors, analyzers, and defenders. Inform Syst Res 12(1):11–33
34. Tan FB, Gallupe RB (2006) Aligning business and information systems thinking: a cognitive approach. IEEE T Eng Manage 53(2):223–237
35. Kearns G, Sabherwal R (2007) Strategic alignment between business and information technology: A knowledge-based view of behaviors, outcome, and consequences. Journal of management information systems, 23(3):129–162
36. Muthusamy SK, Palanisamy R (2004) Leveraging cognition for competitive advantage: A knowledge-based strategy process. Journal of information & knowledge management 3(3):259–272
37. Feurer R, Chaharbaghi K (1995) Strategy development: Past, present and future. Management Decision 33(6):11–21
38. Auer T, Reponen T (1997) Information systems strategy formation embedded into a continuous organizational learning process. Information resources management journal 10(2):32–43

Business and ICT Alignment in Higher Education: A Case Study in Measuring Maturity

Balbir S. Barn, Tony Clark, and Gary Hearne

Abstract Aligning information communication technology (ICT) to business goals is a common issue cited by senior executives. In the UK, the Joint Information Systems Committee (JISC), a government funded body for driving innovation in UK education and research by championing digital technologies, has also focussed its attention on the need for alignment in higher education. This paper reports on the deployment of a strategic ICT (SICT) toolkit that aims to provide a maturity measure of the extent of strategic alignment of an institution. The results from the survey distributed to 65 senior managers in the organization provide a maturity measure and indicate that Enterprise Architecture and integration requirements between business and IS planning are central to increasing the maturity levels for universities engaged in business and IT alignment (BIA).

1 Introduction

Higher education institutions (HEIs) in England and Wales are faced with a challenging and dynamic business environment where public funding of HEIs has been reduced by up to 70 %. This lost funding is being replaced by the introduction of a new student fees regime beginning in September 2012 following a bill introduced in the UK parliament in November 2010. While these recent changes are specific to England and Wales, like the rest of Europe and elsewhere, the HE landscape has also changed with the emergence of private providers and increasing distance and on-line education from international institutions all creating additional competition.

B.S. Barn (✉) • T. Clark • G. Hearne
Middlesex University, Hendon, London, NW4 4BT, UK
e-mail: b.barn@mdx.ac.uk; t.n.clark@mdx.ac.uk; g.hearne@mdx.ac.uk

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_4,

This environment places greater expectations on operational efficiency and one mechanism for addressing this is to ensure that business strategy and IS/IT strategy is better aligned [9]. In recognition of these drivers, the UK Joint Information Systems Committee (JISC), a government funded body for driving innovation in UK education and research by championing digital technologies, has funded a range of projects that have firstly, articulated a business case for the need for greater business and IT alignment (BIA) [8] and then developed toolkits for exploring the maturity of strategic ICT (SICT) alignment in higher education [1].

In a commercial (non-academic) environment the relationship between business and Information Systems/Information Technology (IS/IT) performance has received much attention from as far back as 1977 [16]. There is ample empirical evidence that support the hypothesis that those organizations that have aligned successfully their business and IT strategy will outperform those that have not [6, 7]. Relatively, the academic domain has received less research focus on the study of the business of higher education and alignment with IS/IT. To date, Sabherwal and Kirs investigated the alignment of critical success factors with IT capability and confirmed that alignment facilitates both perceived IT success and organizational performance (perceived IT success refers more specifically to the extent to which the senior managers of the organization believe IT capability contributes to the organization's success). However they also found that integration facilitates neither alignment nor perceived IT success. Integration, they maintain is "accomplished through such integrative mechanisms as task forces, interdepartmental committees, and liaison personnel to coordinate the activities of interdependent departments" and has corresponding information systems structures to support these activities [20]. Later work by Chan et al. used the data collected in the earlier study along with data from other research to explore issues of BIA across different organizational types [7]. A key finding from an academic institution perspective was that:

In business firms, organizational size affected alignment significantly as expected, but environmental uncertainty did not. By contrast, in academic institutions, environmental uncertainty affected alignment significantly as anticipated, but organizational size did not. [7, p. 10].

One explanation for this is that whilst academic institutions vary in size (e.g. student enrollment) they have a similar organizational structure and business processes such as course validation processes. This has implications for the projects funded by JISC and in particular the appropriateness of a SICT toolkit for across the HE sector. The UK HE sector includes small specialist institutions such as music colleges as well as large research and teaching universities. Other isolated work exploring BIA in higher education includes research by Motjolopane and Brown who found that a critical factor in achieving alignment was the integration between business planning and IS planning (BP-ISP integration) [17]. More recently, Vargas et al. using a consolidated framework of strategic alignment models (SAMs) conducted case study research of four public institutions in Nicaragua and some of the relevant findings from that research include: the importance of "Architecture Integration" and the role of IT in strategic business planning [21].

1.1 Contribution of This Paper

While it is evident that HEIs have been the subject of studies to determine the critical features/requirements for ensuring BIA, research on the measures for the maturity of alignment has not been conducted. Thus the experimental data arising from the deployment of a SICT toolkit presents an excellent opportunity to expose current thinking in this area. Further, published research in the HE domain both in the UK and elsewhere exploring ICT/Business alignment is relatively small and so theoretical models in which to contextualise results are relatively sparce. Given the proven importance of BIA for industry this is potentially important for this sector and consequently this paper makes a useful contribution to better understanding BIA in the tertiary sector.

The use of the JISC SICT toolkit in our case study institution is presented and the results analysed with respect to preceding related research. Given that the toolkit will be available for general use under a creative commons licence, the use of the toolkit, its limitations and emerging issues is of relevance to researchers considering applying the SICT toolkit elsewhere.

The remainder of the paper is structured as follows: Sect. 2 presents the JISC SICT Toolkit, the basis for selecting elements of the toolkit to use and its relationship to other SAMs. Section 3 presents the design of the research instruments, the constraints of a case study organization and the constraints of the over-arching programme of research. Section 4 provides a validation of the use of the SICT toolkit and Sect. 5 presents concluding remarks, a discussion of some of the recognized limitations and future work.

2 The JISC S-ICT Toolkit

Alignment between business and IT is an oft cited issue and challenge facing organizations made even more elusive as a consequence of even the term alignment having many definitions reflecting the different foci including integration, linkage, bridge, fusion or even fit. For the purposes of this paper we will utilize the definition derived from an early and leading model for alignment originally proposed by Henderson and Venkatraman, the SAM [10]. They state that alignment is the degree of fit and integration among business strategy, IT strategy, business infrastructure and IT infrastructure. Other researchers have extended the SAM and its associated definitions with varying success and usage. Luftman proposed the strategic alignment maturity model (SAMM) that measures maturity levels of strategic alignment along several key dimensions such as governance and skills and categorizes levels into strategic, tactical and operational states [15]. Vargas et al. have produced a model that has been consolidated from a review of existing models and studied the use of the model in case studies of public universities in Nicaragua as

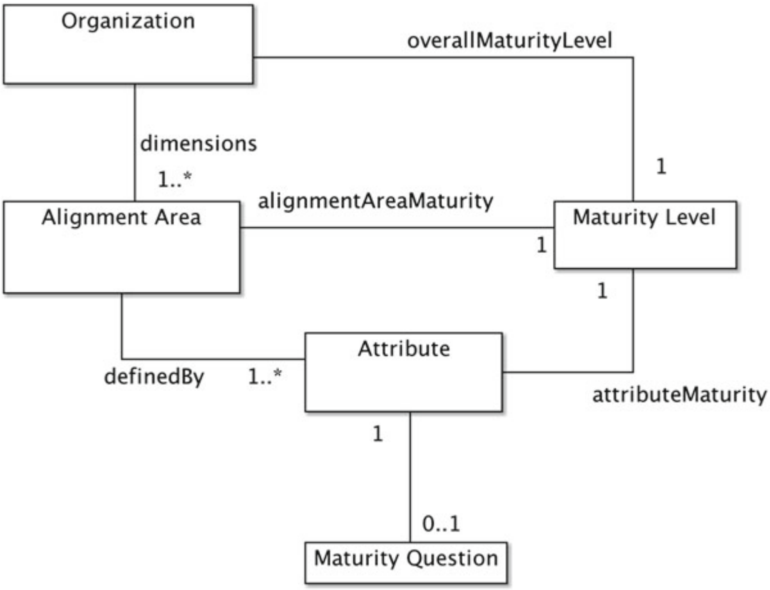


Fig. 1 Common conceptual model for business and ICT alignment

discussed in Sect. 1 [21]. Other models for strategic alignment have focused on executive feedback measures [2], organizational structure [4] and social dimensions such as vision [19].

From these a common conceptual model emerges (see Fig. 1). As the model indicates, an Organization has an overall maturity level. Key areas are denoted as alignment areas within the organization and are defined by attributes. Both the alignment area and the attribute also have a maturity level. Attributes making up an alignment area have a specific question associated with them to which the response contributes to the maturity level.

This model also manifests itself in the JISC SICT toolkit developed to help HEIs to analyse, assess and develop their strategic use of IT. The SICT was commissioned by JISC and the Leadership Foundation for Higher Education to support all senior decision-makers in developing towards more strategically directed ICT deployment in recognition that the UK Higher Education (HE) sector will continue to face major challenges as a result of political, socio-economic, demographic and technological pressures. The toolkit is available in a downloadable form under a creative commons licence [http://tinyurl.com/c9puwfb].

Despite the high levels of deployment of ICT, there is perception (not widely cited in the literature) that ICT does not always add sufficient "value" to an institution either operationally or strategically in providing the capacity to realise the changing needs of today's HEIs. A mature ICT strategy allows an institution to be more agile in meeting strategic demands within appropriate timeframes. Following on, the maturity of SICT is defined as a measure of how well ICT is integrated and aligned with the enterprise wide institutional strategy and therefore able to deliver maximum value and agility. An institution's Strategic Plan may require ICT to support the delivery of operational, strategic or transformational objectives [1].

2.1 Components of the SICT Toolkit

The SICT provides three key components: (1) A knowledge base of information that provides awareness of the factors that may have an impact or influence the strategic use of ICT. These are "enablers" of SICT; (2) A set of case studies drawn from institutions within the HE sector with models of operation, strategic and transformational maturity and (3) a self-analysis framework that uses a structured method to measure the an institution's current maturity—the primary focus of this paper.

The enablers described in the SICT are the areas of maturity that can be used to describe the state of alignment maturity of an organization. The areas are:

Strategic Leadership	Vision and articulation of an institution's mission and ethos The role of senior leadership in developing strategic use of ICT.
ICT Services	ICT-based delivery of key operational functions—examples are email, HRM, Website, Purchasing.
Shared ICT Services	Shared ICT-based delivery of functions—across institutions made possible because of the similarity of structure of business processes.
ICT Governance	Specification of decision rights and accountability framework: What decisions, Who makes them, and How (inc.: monitoring).
Comms and Engagement	Communication of SICT throughout the institution.
Enterprise Architecture	A coherent whole of principles, methods and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems and infrastructure.

These areas correspond to the "Alignment Area" concept in Fig. 1 and are a grouping of attributes whose measures contribute to a maturity level of the area. The attributes are used in the design of a weighted questionnaire that computes an ordinal maturity value that classifies each alignment area/dimension into one of three levels:

Operational	ICT supports the organization providing reliable and satisfactory performance. The use of ICT is primarily for managing costs and there few, if any institutional ICT initiatives.
Strategic	ICT provides good value and supports the institution at a business process level. There is good business alignment between institutional needs and ICT and

	operational capabilities exhibit a high dependence on ICT. The institution is not an *agile* university but is able to deliver against some strategic needs.
Transformational	An agile university with a strong focus on institutional needs and strong capabilities in horizon scanning and using ICT to drive business innovation. The ICT strategy is highly integrated and aligned with institutional strategy with clear and managed ICT governance.

In contrast to Luftman's SAMM model [15] the model underpinning the SICT does not clearly identify the various attributes that contribute to each of the areas. For example, there are six attributes that contribute to the Communication area in SAMM such as understanding the business of IT and inter-organisational learning. These attributes themselves have a maturity level. (This is shown by the conceptual model in Fig. 1).

SICT is intended to be flexible in its use and the toolkit is organized so that individual dimensions can be selected for measurement. Like other maturity models, one objective of SICT is to provide a high level aggregated summary of strategic BIA to senior management. This is achieved by an aggregation procedure for calculating an overall "maturity" level for each areas depending on the responses from the questionnaires. Each dimension has a fixed number of questions that support responses using a five point Likert scale. The responses are discrete and mutually exclusive; however, the five point scale is mapped down into the three level maturity scale. Thus some responses do not contribute to the overall maturity of a dimension and hence of the overall organization.

The next section describes our usage of SICT in the context of a typical UK HEI.

3 Validation

Our validation of SICT had some similarity to the approach taken in the work done by Khaiata and Zulkerman in their adaptation of SAMM [13]. We also adopted a case study-based approach as there are several examples in IS research where there is evidence that case study-based methodologies are well suited for exploring business processes in an organizational setting (e.g. [12]).

The HEI we chose is a university that received university status in 1992 after conversion from polytechnic status. The university has academic departments organized into four Schools and a range of centralized services such as IT. The university is primarily a teaching university with a developing research capability. A key strength of the university is its international presence with two physical campuses overseas and a large range of partnerships with overseas academic providers. The corporate strategy of the institution addresses both the UK and overseas campuses and the centralized IT services have similar responsibilities. The head of IT is part of the senior management of the institution.

3.1 Methodology

A key initial question facing the project team was to determine which of the dimensions of SICT should be deployed. The decision was in the end governed by a review of the literature and the background of the research team (actively researching enterprise architecture). We chose to use all the dimensions except for the "Shared Services" element. We felt that the issue of shared services was insufficiently articulated in the supporting documentation supplied with the toolkit. Preliminary meetings with some informed senior managers also confirmed that understanding the concepts around shared services would be challenging.

Our initial aims of the project also focused on this research being primarily one about exploration of strategic alignment of ICT and consequently we did not initiate any particular hypothesis. The delivery capability of the toolkit (supplied as an EXCEL spread sheet) also influenced us towards an exploratory study rather than one of hypothesis testing.

The spreadsheet form of the toolkit required a lot of manual intervention for collating data so the project team converted the spreadsheet-based questionnaire to a SurveyMonkey account and provided a single URL/Collector. At this stage, some of the questions were re-worded to improve clarity with permission from the SICT designers. This collector link was tested before deployment. To support the web-based deployment (we planned to send the URL to our target respondents), we provided summaries of the key concepts and principles of each of the elements of the questionnaire (strategic leadership, enterprise architecture, etc.) using as a source, the toolkit documentation.

The project team agreed that respondents should primarily be senior managers of the institution, that is, the Senior Executive (University level), The School Executive (Deans, Associate Deans, Directors of Resources, Heads of Departments) and Service Heads such as Library, Information Services, Resources, International recruitment and Estates. Most of these target respondents were informed of potential selection at a variety of University Committees such as Business Development Foresight Group (BDFG), Research Committee, Academic Committee and IT Strategy Committee. The BDFG and IT Committee was in particular used to provide a background context to the proposed research.

After selection discussions, 65 respondents were selected and sent emails asking them to participate in the research. The email included a supporting statement from one of the DVCs (who was also the project champion), assurance about anonymity and optionality, the URL for the web survey and deadline for completion. Respondents had 3 weeks to complete the survey.

In the end 34 valid responses were received and subject to analysis. These were later reduced to 32 after undertaking a data cleansing and preparation for SPSS exercise. This response rate was deemed to be adequate to support the exploratory nature of the project but was insufficient to provide any meaningful hypothesis testing.

During the collation of the results into a single value of SICT alignment we discovered that the original SICT toolkit spreadsheet had a computation error in the weightings allocation for one of the questions. This was raised in the shared blog and the bug was corrected by the SICT designers.

Once we had a single value for each of the dimensions we were able to use the SICT toolkit to determine the indicative maturity level for our case study institution. After import of the data into a statistical package—SPSS 19, the data was prepared by adjusting Likert variables from scalar to ordinal. Variables for which there were no responses were also removed. Importantly, new variables representing each of the five dimensions (or maturity areas) were added. These variables computed the dimension total (unweighted sum of responses) and mean. These variables gave the opportunity for maximum analysis.

This overview maturity level and analysis was then documented and presented to the IT Committee of the institution towards the end of the project. Feedback from the Committee forms part of the analysis in this paper.

Published research in the HE domain exploring ICT/Business alignment is relatively small and so theoretical models in which to contextualise results are relatively sparce. Thus the focus has been one of exploratory activity rather than the testing of hypotheses. As the SICT becomes more widely deployed in relatively standard ways, such empirical research will be a more appropriate and valid form of research.

3.2 Results and Discussion

For the purposes of this paper, we have chosen to focus the results and discussion around three key areas. Firstly, consistent with the aims of the SICT, it is useful to present an executive focussed overview of the SICT alignment maturity of our case study institution. Secondly, we present a subset of our results that contribute to the discussion on BIA in higher education. As the current literature indicates there are issues in: integration between business planning and IS planning (BP-ISP integration) [17]; importance of "Architecture Integration" and the role of IT in strategic business planning [21]. Thirdly, we discuss some of the operational and practical findings around the use of the actual toolkit.

3.2.1 A High Level Overview of Maturity

The aggregated responses from the completed questionnaires resulted in an institutional maturity level classified as Operational. As the figure below illustrates: the organization exhibits an overall score of 26 (operational level). The diagram represents some interesting and conflictual results. A high score on ICT governance (the specification of decision rights) contradicts a low score for Enterprise Architecture. Discussion with senior managers following presentations at the ICT committee indicate that Enterprise Architecture is a poorly understood term outside the IT Staff, certainly, the all encompassing nature is not recognized [11, 14, 18, 22].

The overview chart is useful in very clearly identifying areas which need examination if the HEI has accepted that as in the commercial sector, BIA is important. If an organization does not review its maturity level nor plan for improving its maturity level then, it is likely that the HEI is going to be less successful than those that do [6, 7]. At an organizational level, the survey and its presentation to senior managers of the institution had an immediate impact. This was the first time that there had been a strategic, managed process to measure the value of ICT provision. There was further intention to develop institutional specific tools to measure ICT provision going forward.

3.2.2 Reliability and Correlation

For the analysis conducted with SPSS 19, the additional variables introduced to represent derived information of the dimensions such as EA were subject to reliability analysis so that we could assess the internal consistency of the responses in each section. Cronbach's alpha test was used to get an indication of the randomness of answers [5]. Only the ICT Services dimension did not pass the threshold value of 0.7. These additional variables also allowed the possibility of conducting correlation tests between these dimensions which showed general correlation.

3.2.3 Specific Issues of Architecture Integration and BP-ISP

Previous research has reported that both Enterprise Architecture (EA) and the integration between Business planning and IS planning have been cited as important for higher education [21]. Our research confirms this finding. Enterprise Architecture (EA) in particular scored low. Given that the universities have invested enormously in business and ICT systems the issues around data existing in silos; the perception that ICT is a bottleneck to organizational change; business process duplication; and a real lack of concrete information concerning whether the institution gets good value from ICT investment suggests that the role of EA has not been sufficiently articulated. The figure below presents the specific findings for this dimension.

In the UK, JISC has been promoting the value of EA and how it can be used to address some of the current policy changes. For example, the use of shared services across institutions in order to reduce costs is predicated on documented, current and viable enterprise architecture. In the case study organization, the dominant response to every question was "Not Known". Given that the majority of respondents were senior managers (departmental heads of academic divisions and services) this was a surprising result. If as JISC and others such as [21] have indicated that EA is important then addressing this concern is key if an organization is to improve its maturity for BIA. It also indicates that JISC should consider further how its focus on EA and the value arising from EA can be promoted in the HE sector.

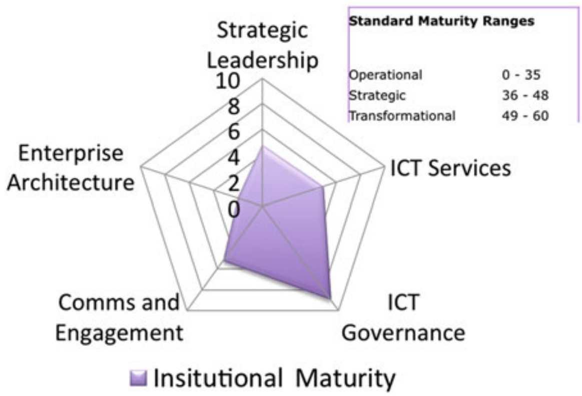


Fig. 2 Maturity level overview

The SICT toolkit positions ICT strategy as a key element of Governance. The responses to some of the questions in this area were interesting because of the disconnects apparent. Governance contributed strongly to the overall maturity level (see Fig. 2). We suspect that the relatively high score was due to conformance to institutional processes rather than a deeper understanding. For example, almost 60 % of respondents believed that projects were subject to detailed progress monitoring and reporting. Similarly, the availability of a documented ICT strategy (again weighted positively) was also noted by 60 % of respondents. In contrast, when respondents were asked how ICT strategy was related to other strategies, 62 % did not think or could not comment on how strategies were integrated. With respect to architectural integration as noted by Vargas Chevez [21] there was a lack of knowledge of the understanding of these issues with almost 70 % either unaware or disagreeing with the existing of architectural integration component in the ICT strategy.

3.2.4 SICT as an Instrument for Measuring Maturity

Our usage of the toolkit exposed to several areas for potential improvement. While SICT was approximately conformant to the derived general conceptual model presented in Fig. 1, it would have benefited from a clearer identification of attributes that contribute to maturity levels. This would enable greater cross validation of results, for example, by comparison with studies conducted by such instruments as Luftman's SAMM. The toolkit could also have benefited from some overarching questions on maturity level opinions for each of the dimensions. These could then have been tested and compared with the additional variables introduced during the analysis.

A key finding from an existing large study by Chan et al. [7] suggested that environment factors were important in affecting alignment of business and IT strategy. We note that while SICT does not consider this an important dimension, given the current external factors facing the UK HE sector are driving major change, we see this as a significant omission.

Maturity is not a stationary value, it evolves over time. For example, as EA gets embedded in an organisation, the maturity level will change. The SICT toolkit must begin to include elements that can support longitudinal studies so that changes in maturity levels can be measured over time. Enterprise Architecture takes time to be embedded and projects that can measure before and after maturity would take that into account.

4 Conclusion

This paper has presented the results and analysis of an exploratory study addressing strategic business and ICT alignment in higher education. Studies that have investigated non-profit organisations are relatively sparse hence exploratory studies where there are no obvious hypotheses can still make a useful contribution in setting out this landscape.

ICT is anticipated to play an important role in helping the tertiary sector address some of the key environment factors currently affecting the sector such as policy changes, demographics, and the changing nature of learning. A better understanding of BIA through studies such as that reported in this paper can help refine the nature of that role.

We also recognise that the limitations of a single case study are apparent even though the nature of the practice of governance in HE shows a relatively small variance [3]. Not withstanding this, results indicate that EA and integration requirements between business and IS planning are central to increasing the maturity levels for universities engaged in BIA but are not being adequately understood by the academy.

There are number of further research activities planned. We intend to refine the SICT toolkit to address some of the operational issues that were raised during this experiment. These include reducing the ambiguity between questions and making clearer the link between alignment attributes and the survey questions. We also recognise that alignment is a continuous activity and we plan to collect a second tranche of data at a later stage to see the evolution of the alignment maturity. Finally we are in discussions with directors of IT services in other academic institutions with a view to executing a refined version of the study to enable a more sector wide view of BIA.

References

1. Addison A (2010) Strategic ICT Toolkit. http://www.jisc.ac.uk/whatwedo/programmes/strategicmanagement/strategictechframework.aspx
2. Avison D, Jones J, Powell P, Wilson D (2004) Using and validating the Strategic Alignment Model. J Strateg Inf Syst 13(3):223–246
3. Barn B, Dexter H, Oussena S, Petch J (2007) Synthesis of multiple process models for creating a rich enterprise application reference model for service oriented architectures. IADIS Appl Comput J 10(1.89):5775

4. Bergeron F, Raymond L, Rivard S (2004) Ideal patterns of strategic alignment and business performance. Inf Manag 41(8):1003–1020
5. Blunch NJ (2008) Introduction to structural equation modelling using SPSS and AMOS. Sage, London
6. Chan YE, Reich BH (2007) IT alignment: what have we learned? J Inf Technol 22(4): 297–315
7. Chan YE, Sabherwal R, Thatcher JB (2006) Antecedents and outcomes of strategic is alignment: an empirical investigation. IEEE Trans Eng Manag 53(1):27–47
8. Duke J, Jordan A, Powell B (2008) A study for the JISC into the integration of technology into institutional strategies. http://www.jisc.ac.uk/publications/reports/2008/strategicdevelopment-finalreport.aspx
9. Gregor S, Hart D, Martin N (2007) Enterprise architectures: enablers of business strategy and is/it alignment in government. Inf Technol People 20(2):96–120
10. Henderson JC, Venkatraman N (1993) Strategic alignment: leveraging information technology for transforming organizations. IBM Syst J 32(1):4–16
11. Hoogervorst J (2004) Enterprise architecture: enabling integration, agility and change. Int J Coop Inf Syst 13(3):213–233
12. Huang J, Newell S, Poulson B, Galliers RD (2005) Deriving value from a commodity process: a case study of the strategic planning and management of a call center. In: ECIS 2005 proceedings, p 39
13. Khaiata M, Zualkernan IA (2009) A simple instrument to measure IT-business alignment maturity. Inf Syst Manag 26(2):138–152
14. Lindström Å, Johnson P, Johansson E, Ekstedt M, Simonsson M (2006) A survey on CIO concerns-do enterprise architecture frameworks support them? Inf Syst Front 8(2):81–90
15. Luftman J (2004) Assessing business-IT alignment maturity. In: Strategies for information technology governance. Idea Group Publishing, London, UK, p 99
16. McLean ER, Soden JV (1977) Strategic planning for MIS. Wiley, New York
17. Motjolopane I, Brown I (2004) Strategic business-IT alignment, and factors of influence: a case study in a public tertiary education institution. In: Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries. South African Institute for Computer Scientists and Information Technologists. Cape Town, South Africa, pp 147–156
18. Pereira CM, Sousa P (2005) Enterprise architecture: business and IT alignment. In: Proceedings of the 2005 ACM symposium on applied computing. ACM, New York, p 1345
19. Reich BH, Benbasat I (1996) Measuring the linkage between business and information technology objectives. MIS Q 20(1):55–81
20. Sabherwal R, Kirs P (1994) The alignment between organizational critical success factors and information technology capability in academic institutions. Decis Sci 25(2):301–330
21. Vargas Chevez N (2010) A unified strategic business and IT alignment model: a study in the public universities of Nicaragua. In: AMCIS. AIS Electronic Library, KTH
22. Zachman JA (1999) A framework for information systems architecture. IBM Syst J 38(2/3):276–292

Investigating Post-outsourcing Decisions: Using the Intellectual Capital View

Hamish T. Barney, Aybüke Aurum, Graham C. Low, and Kevin Wang

Abstract Over the last 30 years, IT outsourcing has become an enormous industry, and it continues to grow today. Over this period, outsourcing has matured and become widespread; this change has meant that for many managers, outsourcing questions have shifted from "Whether or not I should outsource?" to "What should I do when the outsourcing contract is about to expire/be terminated?" This study presents multiple case studies of the post-outsourcing decisions faced by two companies operating in Australia and begins the process of understanding this new class of outsourcing decisions. The study identifies the factors considered during post-outsourcing decisions and their interrelationship. The findings show that the factors affecting post-outsourcing decisions are of a combination of initial outsourcing decision factors and experience factors with the previous vendor. Several factors that were important in the initial outsourcing decision become much less relevant in the context of post-outsourcing decisions.

1 Introduction

The age of outsourcing megadeals began when Kodak signed a $1billion (USD) agreement with IBM in 1989 [1], which established IT outsourcing as a matter of strategic importance. Today, with the emergence of India and other outsourcing hubs, outsourcing has become widespread and the IT outsourcing industry has matured. Just as the number of outsourcing contracts has increased, so has the

H.T. Barney (✉) • A. Aurum • G.C. Low • K. Wang
School of Information Systems, Technology and Management,
University of New South Wales, Sydney, NSW, Australia
e-mail: hamish@hbarney.com; aybuke@unsw.edu.au; g.low@unsw.edu.au;
Kevin.Wang@student.unsw.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_5,

number of prematurely terminated outsourcing contracts [1]. Up to 51 % of all IT outsourcing contracts [2] are terminated prematurely and this figure appears set to continue rising. For instance, JPMorgan Chase prematurely ended a 7-year, $5billion (USD) contract with IBM to bring some of its IT functions back in-house [3]. Higher contract turnover highlights the increasing frequency with which companies are facing post-outsourcing decisions, which motivates this research.

As the proportion of businesses that outsource some or all of the IT has increased [2], the question that these businesses face has changed from "Should this function be outsourced?" to "What should I do when the current contract terminates?" A significant proportion of outsourcing contracts need to be renegotiated on a fairly routine basis [4]. It is difficult for prolonged outsourcing contracts to support the dynamic IT demands of corporations over time. Some firms are even choosing to bring certain IT functions back in-house after a contract has expired [3] because of the increased flexibility offered by in-house arrangements.

Several studies [3, 5] have highlighted the differences between post-outsourcing decisions and the initial "whether or not to outsource" decision. An assessment of the literature revealed that existing research has focused on describing factors affecting the initial sourcing decision and the factors affecting termination of outsourcing contracts. These factors are insufficient in the case of post-outsourcing decisions; issues like the experience with the previous outsourcing vendor also need be taken into account [5]. The objective of this study is to explore the factors which influence post-outsourcing decisions. This research is significant given that, to the best of our knowledge, there is no empirical study that specifically examines the post-outsourcing decisions using the intellectual capital view (where intellectual capital is defined as a combination of human, social, organisational and physical forms of capital). By contributing to the understanding of post-outsourcing decisions, through two in-depth case studies, this research will facilitate a better understanding of post-outsourcing activities in general, leading to more successful IT projects.

The remainder of this paper is structured as follows. A brief background to the study is presented in Sect. 2. Section 3 describes the two case studies, while the research methodology is presented in Sect. 4. In Sect. 5, the results of the case studies are presented. Section 6 provides a discussion. Finally, Sect. 7 presents the concluding remarks.

2 Background

We define post-outsourcing decision as "the decision about how an IT service will be provided when the current outsourcing contract is facing expiration or termination". Our assumption is that the business has time to make an informed post-outsourcing decision.

2.1 Implications of Terminating an Outsourcing Contract

When the existing contract is facing expiration, the business has three options: renegotiate/extend the current contract, backsource or switch vendors. In most contract termination cases, renegotiation is not an option because the client is not satisfied with the current vendor's performance and may not wish to continue the relationship [6]. The remaining options are not necessarily comfortable either. Backsourcing, for example, has three major drawbacks. Firstly, backsourcing is expensive, costing from 2 to 15 % more than outsourcing [5]. Secondly, backsourcing reflects negatively on previous management decisions [7]. Thirdly, backsourcing is a lengthy process and has a significant impact on the organisational structure, including the need to establish and integrate new knowledge, resources and capabilities in-house, as well as establish a new system of infrastructure and employees [8]. Similarly, the option to switch vendor also has several drawbacks. For instance, there are significant search costs associated with finding a suitable vendor and drawing up an arrangement, not to mention the need to dedicate more resources to ensuring that the new outsourcing partner is integrated into the business.

2.2 Post-outsourcing Decisions

Prior research has consistently identified cost [9–12] as a key driver of outsourcing. However, in other studies, factors such as power and politics, gaps in outsourcing expectations and external factors (e.g. vendor's strategic direction) have been shown to be just as important as cost [13–15]. While the factors that affect initial outsourcing decisions are also relevant to post-outsourcing decisions, there are additional factors that need to be considered in post-outsourcing decisions.

McLaughlin and Peppard [3] reviewed public records related to high-profile outsourcing deals to examine the motives behind backsourcing decisions. They found that post-outsourcing decisions are heavily influenced by the most recent outsourcing experience and changes to the technology and environment. Whitten and Leidner [5] examined the predictors of firms' choices to backsource or switch vendors. Their results showed that post-outsourcing decisions are highly influenced by the nature of the relationship and experience with the previous vendors. Neglecting these aspects of the post-outsourcing decision identified by prior research could be problematic. Decisions to terminate an outsourcing relationship and the subsequent post-outsourcing decision are not always driven by a negative experience (for instance, if a significantly more attractive option recently became available). There are various difficulties when assessing these changes suggested by [3] because these dynamic environmental factors need to be assessed in addition to reassessing the factors that influenced the initial outsourcing decision.

2.3 Intellectual Capital View

Intellectual capital (IC) is a derivative of the resource-based view. Intellectual capital is composed of four elements (human, social, organisational and physical capital) which are utilisable for competitive advantage [16, 17]. Human capital refers to the expertise and skills of the individuals. Social capital refers relational capital that is built up between individuals encompassing organisational culture and shared understanding that facilitates communication and collaboration. Organisational capital refers to the things that stay in the possession of the company when people go home at night including "institutionalised knowledge and codified experience residing within and utilised through databases patents, manuals, structures, systems and processes"; physical capital refers to tangible assets such as costs or equipment [16].

A literature review of IC in software development shows that the majority of studies cover investment issues in software product line practice and the impact of upfront investment on cumulative return in terms of money, effort and time [18].

Research suggests that since IT can enhance organisational memories, it should facilitate the institutionalisation of knowledge [16]. If IT investment in internal organisational resources can enhance the value of an organisation's intellectual capital, what are the impacts of outsourcing the function? What happens to knowledge when a function is outsourced and how IC can be retained and the creation of new IC can be supported in an outsourcing arrangement become important questions.

3 Case Study

The data was collected from two different companies where each company is regarded as one case study. These cases were chosen because these companies were both facing post-outsourcing decisions. Companies were selected from two distinct industry sectors to increase the external validity of the research, helping to identify the factors that are possibly common to post-outsourcing decisions in general.

Company A: Company A is one of Australia's most respected food manufacturing companies. It had a 4-year relationship with Vendor A. The outsourcing arrangement was a typical utility model where Vendor A1 was responsible for hosting and supporting both applications and infrastructure. However, a mix of issues including poor contract management and poor service delivery resulted in a dysfunctional outsourcing relationship. Toward the end of the contract, it was clear to both parties that the contract would not be renewed. Company A was then faced with a post-outsourcing decision: backsource or switch to another vendor. Cost and environmental reasons led to the rejection of backsourcing. Company A then spoke to several alternative vendors, including those vendors recommended by its parent company in the USA. Ultimately, the company chose to align itself with its parent company under the same international vendor (Vendor A2). Vendor A2 had a strong presence in Australia and had already worked with Company A on an ERP implementation. Vendor A2 was given control over the new contract and part of the migration process. So far, Company A is impressed with Vendor A2's performance.

Company B: Company B operates in the banking and financial services industry and in Australia has approximately 11.8 million customers and 39,000 staff worldwide. The organisation has a multi-brand model operating across many customer facing business units, including retail banking, institutional banking and wealth management. Company B merged with another major retail bank (Company B2) in 2008. It has undertaken several major IT projects to consolidate duplicated IT systems. The outsourcing contract for the management of Company B's credit card platform was nearing expiration after 10 years. Company B was facing a post-outsourcing decision. This was a good opportunity to consolidate the support for Company B and B2's credit card platforms. Company B initiated an invitation-only-based selection process. The bidders included the two incumbent vendors responsible for supporting Company B and B2's credit card platforms, respectively, and a third vendor who was servicing Company B in another area. Backsourcing was also considered. The two existing vendors offered the most attractive proposals, with Vendor B1 winning due to lower switching costs and risks.

4 Research Methodology

Given that post-outsourcing decisions have become increasingly common, this research examines the following research questions:

- RQ 1: What are the factors considered when making post-outsourcing decisions?
- RQ 2: What is the relative importance of the factors identified?

This paper aims to study post-outsourcing arrangements from the theoretical lens of the IC view [16] combined with Sullivan's [17] conceptualisation of "knowledge company". This helps us to identify "tangible" and "intangible" factors, which together provide a comprehensive set of factors considered during post-outsourcing decisions.

4.1 Research Method

To address the research questions, an exploratory case study, with semi-structured interviews followed by a questionnaire, was undertaken. The interview questions covered the participants' background, the decision process in post-outsourcing arrangements and the future of the current outsourcing relationship. The qualitative data that these interviews provided were used to address RQ1 and RQ2.

The questionnaire provides a means of assigning a priority to the putative post-outsourcing decisions factors to address RQ2. Relevant factors were compiled from the literature. Factors which have been shown to be important in the initial

Table 1 Participant distribution

	Company A	Company B	Total
Executives/managers (decision-makers)	CIO IT director	Programme director Programme head of IT Sourcing manager	5
Senior staff (decision implementers)	Service delivery manager Function lead Service desk lead	Senior process engineer Application team lead	5
Total	5	5	10

outsourcing decision [13, 15] and factors which affect outsourcing termination [19] were both used in the questionnaire. Additionally, factors related to the experience with the previous vendor [5], i.e. product/service quality, relationship quality and switching costs, were also included as possible factors in the questionnaire**.** After detailed examination of the literature, 37 factors related to outsourcing arrangements were considered. The full set of factors can be found in [20]. After pilot studies with academics and IT professionals, this list was reduced to 20 factors.

4.2 Data Collection and Analysis

Overall, ten stakeholders participated in this study (Table 1). The samples of the interview and questionnaire participants involved in this study cover almost the entire population of decision-makers in each case study. Two relevant stakeholder groups were managers who have a direct impact on the post-outsourcing contract negotiation and decision-making process and senior staff who have direct input to the post-outsourcing decision and are heavily involved during the implementation of the contract.

During each interview, the participants were given an overview of the research and the areas relevant to this study. The interview then commenced with a sequence of questions specified in the interview schedule. Interviews took approximately 45 min. The recorded interviews were transcribed and imported into NVivo 9 for coding. Thematic coding was applied where the scripts were grouped and coded based on the high-level factor groups specified by the IC view. The codes consisted of factors which the interviewees identified as affecting post-outsourcing. These factors are compared with the literature for discussion.

Following the interview, each participant was asked to complete a questionnaire (Table 2). The questionnaire asked participants to rank 20 factors derived from literature. These factors were grouped based on IC view. Participants were asked to allocate 1,000 points amongst each of the factors based on their perceived importance during decision-making and to identify any factors missing from the list.

Table 2 Factors ranking across two companies

		Company A			Company B		
	Factors Affecting Manager's Post-Outsourcing Decisions.	Mgr	Stf	Ring	Mgr	Stf	Rng
HC	Vendor staff expertise	16	9	7	8	9	1
	Increase workforce flexibility	8	18	10	4	13	9
	Risk of low vendor staff qualification and commitment	5	6	1	13	10	3
	Risk of damaging internal staff morale due to perceived job insecurity	16	19	3	18	13	5
	Risk of losing valuable staff	16	19	3	16	17	1
SC	Risk of disrupting internal social networks	16	17	1	18	19	1
	Risk of communication / collaboration problems with the vendor	8	9	1	9	16	7
	Risk of vendor non-compliance and opportunistic behaviour	1	4	3	15	10	5
	Imitation of internal or external stakeholder's outsourcing success	13	13	0	16	2	14
OC	Lack of organisational knowledge in the outsourced area	13	13	0	9	13	4
	Outsourcing away non-core functions to the vendor	7	15	8	9	20	11
	Risk of losing key internal knowledge and processes to the vendor	11	15	4	13	18	5
PC	Cost savings in staff or technology	3	1	2	1	2	1
	Access to vendor's advanced technology	8	3	5	7	5	2
	Risk of hidden costs in the contract	15	5	10	9	6	3
POR	Quality of previous vendor's staff (HC)	4	6	2	4	4	0
	Quality of relationship with the previous vendor (SC)	2	1	1	2	7	5
	Degree to which vendor knowledge and expertise have been internalised (OC)	16	9	7	18	10	8
	Quality of previous vendor's products (PC)	6	8	2	2	8	6
	Switching costs involved in changing the outsourcing arrangement (SF)	12	12	0	4	1	3

HC – Human Capital,**SC** – Social Capital, **OC** – Organisational Capital, **PC** – Physical Capital, **POR** – Previous Outsourcing Relationship, **SF**– Switching Factors, **Mgr**– Manager, **Stf**– Senior Staff, **Rng**– Difference between Manager and Staff Ranks

= Top 5 Factors = Bottom 3 Factors Underline = Top 3 Rank Differences

HC human capital, *SC* social capital, *OC* organisational capital, *PC* physical capital, *POR* previous outsourcing relationship, *SF* switching factors, *Mgr* manager, *Stf* senior staff, *Rng* difference between manager and staff ranks

Grey shading = top five factors, lined shading = bottom three factors, underline = top three rank differences

5 Results and Discussion

The following sections provide the results of interviews and questionnaire for RQ1 and RQ2.

5.1 Post-outsourcing Factors

In analysing RQ1, from the transcribed interviews, factors are coded, analysed and described based on the elements of IC view, i.e. human, social, organisational and physical capital.

Human capital: The interview results show that one of the main concerns was about the skill set of the vendor organisation demonstrating the importance of this issue. Human capital played an important role in Company A's decision to end their relationship with Vendor as they were *lacking in terms of some of the skill set*. Similarly, human capital was one of the primary reasons Company B felt confident that Vendor B1 would be able to service the new combined credit card system at lower cost and risk. Vendor B1's staff intimate knowledge about how the credit card system worked was one of their biggest assets in their bid; choosing another vendor would require the new vendor's staff to acquire this human capital afresh. Those interviewed were critical of the vendor's sometime hyperbolic claims of access to a deep pool of diverse and deep human capital as being more theoretical than actual. Those interviewed said often these skilled and knowledgeable resources were often not readily available.

Social capital: The interviews revealed new social capital-related factors: "social commitment" on the client side and organisational cultural fit between client and vendor. As the service delivery manager in Company A describes, "If you make a company feel like part of your company they become a stakeholder in your company. If you make a company feel like they are servicing your company, you get best effort". It is evident that in addition to cultural compatibility, the willingness to develop a social relationship is equally important. In the case of Company A, the fact that neither side was willing to commit contributed to the ultimate failure of the outsourcing contract. This issue was critical in Company A's post-outsourcing decision: "We make those folks feel like they really work for us, and as a consequence we had an enormous stability" (CIO, Company B). The findings from Company A also showed that the lack of trust between Vendor A and Company A strongly contributed to the negative experiences. A lack of social capital between vendor and client considerably reduces the likelihood of extending an existing outsourcing.

The second case provided further evidence that social capital is an important factor. In contrast to Company A, Company B's relationship with their previous vendor (Vendor B1) was positive. This led to the invitation-only tender process where only vendors that already had a positive relationship with Company B were considered. The quality of their relationship led to the final decision to renegotiate the relationship with Vendor B. When the final round bids of the two vendors were considered, the programme director commented that “we didn’t know how to manage Vendor B2 as well as we knew how to manage Vendor B. We had some good relationships with Vendor B1 whereas on the Vendor B2 side they were still trying to learn how to work”.

Organisational capital: The degree to which vendor processes and knowledge is transferable and able to be internalised is an important concern in post-outsourcing decisions. Previous experience with the existing vendor makes this factor a very transparent indicator of vendor performance, which is likely to impact the outcome of the post-outsourcing decision.

The interview data suggested that both clients expected their vendors to retain their knowledge to enhance services to the client. For instance, the Functional Lead (Company A) highlighted it is important that “the [previous] management of the outsourcing vendor was committed to an effective knowledge retention…” of the client so that “if somebody was going to leave that they were prepared for that”. This remark also demonstrates the relationship between human and organisational capital. Organisational processes that lead to knowledge sharing ensure human capital continuity if staff leave. However, if staff turnover is too high, continuously training new staff incurs additional costs and reduces the quality of the service provided.

The risk of losing internal processes and knowledge to the vendor was also another concern which was addressed by the senior process engineer in Company B: “I think the risk to Company B would be so big if they choose a different vendor. Because we had given them so much”. The process engineer’s comments highlight potential risks factors considered when the initial outsourcing decision was made have come to pass once the post-outsourcing decision is being considered. Previous experience may limit the post-outsourcing choices available.

Physical capital: The cost of the previous arrangement was used as a baseline for evaluating post-outsourcing options for Company A. The CIO of the Company said, “Now, why we were confident that it was a smart, economic decision… we had some benchmarks for what we were paying Vendor A1”. In post-outsourcing, the cost of the previous arrangement may be used as a basis for comparison; the next arrangement is expected to be superior to the current one. Company B’s programme director identified three dimensions to this factor: “The cost dimension there are three: implementation cost, vendor investment, how much investment the vendor was prepared to make and then a return which was the hard core business case”.

For Company B, in addition to the previous vendor’s products being *well defined* and having *stood the test of time pretty well* (programme head of IT), several organisational specific preferences seemed to be caused by this factor: “[T]he preference for putting in new function rather than upgrading the base product…” (programme head of IT) for instance. These contextually specific preferences need to be accounted for in post-outsourcing decisions because they may influence the choice vendor. The programme head of IT commented: “…if that other product had been richer in function, I think it may have changed our decision”.

Neither switching risk nor switching cost was explicitly considered by interviewees from Company A. This may be because Vendor A2 (the new vendor) was responsible for bearing the risks associated with this transition. Alternatively, there had already been a decision to change vendors, so switching costs played a much more important role in Company B's decision. The alternatives under consideration were very similar on other dimensions; consequently switching costs became a factor of differentiation.

This finding has implications for post-outsourcing because renegotiation as an option will always yield lower switching costs and risks. The most attractive option will be determined by "whether you can capture enough of [the benefits from an alternative arrangement] to make the cost of the implementation to be worth the trouble" (programme director).

5.2 Relative Ranking of the Factors

In analysing RQ2, the participants' understanding of the relative importance of the decision factors is examined. The factors in Table 2 were grouped according to the rubrics of human, social, organisational and physical capital. Additionally, factors which are associated with previous experience are grouped under a separate category. Table 2 also shows the rankings of each factor across two different stakeholder groups and across the two companies. The various shading and column abbreviation are explained in the legend. The top five factors are greyed and bottom three factors are marked (as indicated in the legend). The third column "RNG" is the range between the rankings (i.e. difference between staff rank and manager rank): the larger the range, the higher the discrepancy in opinions. The five biggest discrepancies between managers and staff for both companies are underlined.

These findings suggest that physical factors and previous experience are the most important factors across the two companies. Table 2 provides evidence that although the two companies span different industries, there is some alignment between the top five factors. Physical capital items, namely, cost savings and access to the vendor's advanced technology, are the top factors. This is concordant with the majority of outsourcing literature [15]. The organisation's past experience with outsourcing was also demonstrated to be important with quality of previous vendor's staff and the quality of relationship with the previous vendor ranked highly. This accords with the finding of [5], who found that past outsourcing experience strongly influences the post-outsourcing decision. Interestingly, it appears that when reflecting on the previous outsourcing relationships during a post-outsourcing decision, the intangible and intellectual aspects of the relationships, i.e. social and human aspects, appear to be more important instead of the physical aspects of the relationship (see Table 2).

6 Discussion

One of the key assumptions of this research is that three high-level options for post-outsourcing decision exist: renegotiate with current vendor, switch vendors or backsource. The results showed that for the companies we studied, backsourcing is unlikely; an in-house solution is perceived as being too costly. These findings are obviously preliminary given the limited sample of two companies, and further research is necessary and is underway to confirm these findings.

The findings showed that some initial outsourcing decisions were also reconsidered in post-outsourcing decisions, e.g. cost saving, particularly in the group of physical capital. One of the key differences found between post-outsourcing and the initial outsourcing decision is that when post-outsourcing decisions are made, some of the risks or potential benefits that were considered in the initial outsourcing decision have already been realised (both positive and negative). If the company has a positive experience with their previous vendor, then the relationship is an asset to the company which is lost if the firm chooses to "switch vendors". If it is a negative relationship, i.e. the previous relationship is a liability and a risk hazard, which is also "lost" if they choose to switch vendors. This also has a strong influence on post-outsourcing decision when considering alternative solutions, because in addition to the loss of the "relationship" asset, there is also the new cost associated with reacquiring the lost knowledge, a process which is both expensive and risky.

Although common patterns have been found across the two cases selected in this study, there was strong evidence in the results to support the uniqueness of context and its impact on the post-outsourcing decision. It was evident that the nature of the industry has an effect on the role of IT and consequently the perception of the IT outsourcing contracts. For instance, IT is perceived as a facilitating function (in Company A), and it is likely that during the post-outsourcing decision process is based on "getting the service at the lowest price". In contrast, it was evident in Company B that the "flexibility" of the product/service offering and "partnership with IT vendors" was an important concern.

The specific nature of the function or area may also have an effect on the factors which affects post-outsourcing decisions. The two cases covered in this study are both IT service outsourcing with coverage over both application and infrastructure support. The similarity between the two outsourcing transitions in this study may explain the general alignment of factors considered in Table 2. As Willcocks et al. [21] have identified, the functional area of the outsourcing arrangement has a strong impact on what should be considered during outsourcing. For example, this study found a lack of evidence across both companies for the consideration of "disruption to internal social networks" which was considered in outsourcing literature. This is most likely because this type of outsourcing requires minimal direct client vendor engagement (e.g. minimal vendor staff on site); hence the impact on the internal social network is also minimal. Consequently, the functional-level specificity is also an important consideration during post-outsourcing decisions.

These results are preliminary, given the limited size of the sample. Further research is required to confirm these results and determine the extent to which they are externally valid. Further research addressing this question is already underway. Further case studies and further surveys are being conducted looking at the relationship between intellectual capital and post-outsourcing sourcing decisions.

7 Conclusion

This research examined the relatively unexplored field of post-outsourcing decisions using multiple case studies. The goal was to explore the factors that are considered during post-outsourcing decision-making and better understand the relative importance of these factors.

The evidence shows that cost savings is the main driver in post-outsourcing, and previous experience with vendor and managers' own experience are just as important as cost savings. Backsourcing is perceived as an expensive option because of the high cost of regaining the knowledge required to provide the function in-house and is also complex, timely and risky and only chosen under very special circumstances. The choice between switching and renegotiation is more associated with whether or not:

- The current vendor is providing a satisfactory level of service at a reasonable price
- The relationship with the current vendor is positive
- The alternative vendors are providing high incentives to switch

The findings also show that an existing relationship with a vendor can be an asset, a liability or a risk. The managerial implications of this mean when making post-outsourcing decisions managers need to:

- Treat the "relationship" between the vendor and client as an asset and take a more prominent place in management thinking and decision-making
- Be aware of the risks involved in moving away from the existing relationship if the contract is heavily integrated into the business' operations

These factors may affect the duration, scale or simply the viability of certain options considered. Furthermore, previous experience with a vendor could be used as a baseline in the decision process, where every aspect of the current and potential solution is contrasted to see whether there was potential for improvement.

This study has verified the uniqueness of post-outsourcing decisions. Given the limitations with the sample size, there is an opportunity for future research to replicate this study across various cases and across different industries. The future study will also include developing a set of generic criteria for the IT industry along with a methodology of assessing and comparing a vendor's ability and suitability to deliver services as part of an outsourcing arrangement.

References

1. Dibbern J, Goles T, Hirschheim R, Jayatilaka B (2004) Information systems outsourcing: a survey and analysis of the literature. ACM SIGMIS Database 35:6–102
2. Cluster D (2005) Global IT outsourcing study. Diamond Cluster International, Chicago, IL
3. McLaughlin D, Peppard J (2006) IT backsourcing: from ‘make or buy’ to ‘bringing IT back in-house’. European Conference on Information Systems (ECIS), Goteborg, Sweden
4. Young A (2000) The future of outsourcing: thriving or surviving? Dataquest Service Trends
5. Whitten D, Leidner D (2006) Bringing IT back: an analysis of the decision to backsource or switch vendors. Decision Sci 37(4):605–621
6. Barney HT, Low GC, Aurum A (2009) The morning after: what happens when outsourcing relationship ends. In: 17th international information systems development conference (ISD2008) Paphos, Cyprus, 25–27 Aug 2009
7. Wong SF (2008) Understanding IT backsourcing decision. Pacific Asia Conference on Information Systems (PACIS). Suzhou, China
8. Bhagwatwar A, Hackney R, Desouza KC (2011) Considerations for information systems “backsourcing”: a framework for knowledge re-integration. Inform Syst Manage 28:165–173
9. Akoda J, Comyn-Wattiau I (2006) Developing a framework for analyzing IS/IT backsourcing. Information systems and collaboration: state of the art and perspectives. AIM, Luxembourg
10. Arnett KP, Jones MC (1994) Firms that choose outsourcing: a profile. Inform Manage 26:179–188
11. McFarlan FW, Nolan RL (1995) How to manage an IT outsourcing alliance. Sloan Manage Rev 36:9–23
12. Fontes R (2000) The outsource option. Folio: The Magazine for Magazine Management 112–113
13. Gonzalez R, Gasco J, Llopis J (2010) Information systems outsourcing reasons and risks: a new assessment. Ind Manage Data Syst 110:284–303
14. Kremic T, Tukel OI, Rom WO (2006) Outsourcing decision support: a survey of benefits, risks, and decision factors. SCM 11:467–482
15. Berg B, Stylianou AC (2009) Factors considered when outsourcing an IS system: an empirical examination of the impacts of organizational size, strategy and the object of a decision. Eur J Inform Syst 18:235–248
16. Youndt MA, Subramaniam M, Snell SA (2004) Intellectual capital profiles: an examination of investments and returns. J Manage Stud 41:335–361
17. Sullivan PH (1999) Profiting from intellectual capital. J Knowledge Manage 3:132–143
18. Barney S, Wohlin C, Aurum A (2009) Balancing software product investment In: 3rd International symposium on empirical software engineering and measurement, ESEM'09, Florida USA, 15–16 Oct 2009
19. Wong SF, Jaya P (2008) Drivers of IT backsourcing decision. Communications of the IBIMA, 2(14):102–108
20. Wang K (2011) Investigating post-outsourcing decisions: using the intellectual capital view. Honours thesis, School of Information Systems, Technology and Management, University of New South Wales, Sydney, 2052, Australia
21. Willcocks L, Hindle J, Feeny D, Lacity M (2004) IT and business process outsourcing: the knowledge potential. Inform Syst Manage 21:7–15

Improved Initial Class Diagrams with ATSA:OO

Robert B.K. Brown and Angela M.E. Piper

Abstract Despite the popularity of Object-Oriented programming and design, their analysis phase is difficult to learn and relies heavily on the analyst's intuition and experience. Even industry practitioners can produce initial class diagrams requiring time-consuming refinements in the design phase. Business clients can find it difficult to follow its diverse techniques and constructs. The ATSA method offers analysis under a single theoretical framework that is specifically designed to be easily understood by both neophyte practitioners and business clients. However, as it follows the procedural paradigm, it is not suitable for direct OO application. This paper presents an adaptation of ATSA for the OO paradigm, which produces an initial class diagram, complete with suggested attributes and methods. It offers improved cohesion and comprehensibility over its OO counterpart.

1 Introduction

A previous paper [1] briefly presented the Activity Theoretic Systems Architecture (ATSA) methodology. Activity Theory (AT) is use-centric, based upon a hierarchy of doings that occur within any given human endeavor. This makes it more compatible with the mindset of business clients (who are more easily able to describe their domain in terms of tasks performed and *goal*s satisfied) and, furthermore,

R.B.K. Brown (✉)
School of Information Systems and Technology, University of Wollongong,
Wollongong, NSW 2522, Australia
e-mail: bobbrown@uow.edu.au

A.M.E. Piper
School of Computer Science and Software Engineering, University of Wollongong,
Wollongong, NSW 2522, Australia
e-mail: apiper@uow.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_6,

makes the resulting system less susceptible to changes (such as those arising from staffing reconfigurations).

ATSA yielded system specifications in the form of a data dictionary with transforms noted in pseudocode fragments, anticipating a procedural implementation. However, the majority of the steps within ATSA were agnostic of implementation paradigm. Thus, it was anticipated that ATSA could be adapted to suit the widely used Object-Oriented (OO) paradigm.

Object orientation provides encapsulation, polymorphism, and inheritance which facilitate maintainability, reuse, and modification/extension. These advantages have made OO approach immensely popular.

Of the three phases of the OO approach—Object-Oriented Analysis (OOA), Object-Oriented Design (OOD), and Object-Oriented Programming (OOP)—the majority of these advantages are found in the latter two phases, while the first phase remains problematic. Despite the notion that OOA is "based on concepts that we first learned in kindergarten" [4], it is notoriously difficult for all but the most experienced practitioners to apply [7,11,12].

As ATSA was designed with learnability in mind [2], an OO version of ATSA should be significantly easier for inexperienced practitioners to apply. Therefore, Object-Oriented ATSA (ATSA:OO) should provide an excellent pathway from early phase elicitation into OOD and OOP.

This paper briefly recaps the ATSA method, presents new class finding heuristics underlying ATSA:OO, and provides a brief worked example based on the coffee machine problem drawn from a typical OO teaching text. The example demonstrates that ATSA:OO can generate a workable initial design class diagram.

2 Initial Design: Model-0

In one of the earliest descriptions of the OO concept [11], two phases of design were described. An initial *abstract* design, known as a General Design, emerged from an architectural system-level consideration of the problem description, which would then be passed on for Detailed Design. This description effectively created OOA.

Most current OO approaches and examples present an initial design, arising from (often disappointingly cursory) OOA considerations, typically employing variations of the Use Case diagram technique. There are many (often subtle) variations of this OOA, using different terms for their resulting design. We adopt the general term model-0 to refer to the outcome of any OOA method. In each case, model-0 serves as the basis for ongoing iterative design revision.

Our intention is for ATSA:OO to sit in the place of OOA, offering Activity Theoretic analysis techniques in place of the many Use Case techniques, and therefore, we expect ATSA:OO to generate a model-0 which is at least as good a starting point for further design refinement as that generated by OOA.

We will detail the generation of a model-0 from a typical textbook problem and compare it to a corresponding model-0 from a typical OOAD approach, independently conducted by a practitioner from the software industry.

3 The Coffee Machine Problem

We will use a problem from an OO teaching text. Identifying cohesive classes in this problem, at the correct level of abstraction, has challenged OO students for over a dozen years ([10], p. 129). It describes a simple drip-feed coffee brewing machine with its main physical components and operations as follows:

"The Mark IV special makes up to 12 cups of coffee at a time. The user places a filter in the filter holder, fills the filter with coffee grounds, and slides the filter holder into its receptacle. The user then pours up to 12 cups of water into the water strainer and presses the Brew button. The water is heated until boiling. The pressure of the evolving steam forces the water to be sprayed over the coffee grounds, and coffee drips through the filter into the pot. The pot is kept warm for extended periods by a warmer plate, which only turns on if there is coffee in the pot. If the pot is removed from the warmer plate while water is being sprayed over the grounds, the flow of water is stopped so that brewed coffee does not spill on the warmer plate. The following hardware needs to be monitored or controlled.

- The heating element for the boiler. It can be turned on or off
- The heating element for the warmer plate. It can be turned on or off
- The sensor for the warmer plate. It has three states: warmerEmpty, potEmpty, and potNotEmpty
- A sensor for the boiler, which determines if there is water present or not. It has two states: boilerEmpty or boilerNotEmtpy
- The Brew button. This is a momentary button that starts the brewing cycle. It has an indicator that lights up when the brewing cycle is over and the coffee is ready
- A pressure-relief valve that opens to reduce the pressure in the boiler. The drop in pressure stops the flow of water to the filter. It can be opened or closed" ([9], p. 60)

We briefly report on an analysis of this problem using ATSA:OO, treating the problem description text (above) as the information elicited from a client. The initial two phases of the method run as previously described for the procedural ATSA. For brevity we omit that elicitation and the construction steps of the activity network but offer a brief recap of the ATSA method as previously presented.

4 Activity Theoretic Systems Architecture (ATSA)

ATSA employs a single coherent theory throughout the Systems Analysis and Design (SA&D) process, Activity Theory (AT). It is well beyond the scope of this paper to properly introduce AT, so the interested reader is invited to consult earlier publications [6,8,13].

The construction of ATSA was motivated by a concern that neophyte practitioners lacked the experience and tacit skills required under other methods and approaches and a desire to address the persistently poor success rates reported for IT projects [5].

To establish the basis of the extension offered in this paper, a brief recap of the three main phases of the ATSA method is presented below. A more complete description of the method can be found in Brown and Piper [1].

4.1 ATSA Elicitation

The analyst identifies a list of *roles*, each being some human (who typically occupies some defined "*position*") acting in a specific mindset, engaged in task-specific work, and typically using task-specific tools. Any one position may fill several roles, but there may also be more than one position capable of, or authorized to, acting in a given role.

As the final use-centric system will be described to facilitate doings, ATSA considers *its* actors to be roles not positions. ATSA designs for a *community* of such actors (being roles performing activities) and designs both an overall facilitating system and appropriate *activity*-specific tools for each. Positions, therefore, cease to be of immediate concern to the analyst.

The analyst identifies Candidate Instruments (CIs) from existing files, folders, forms, records, registers, lists, and databases. These are data-like *things* (tangible or otherwise) which are required for roles to perform their work and satisfy their goals or which are made available for other roles as a result of their work.

4.2 ATSA Analysis

Having obtained the obvious business details, the analyst identifies task-specific goals. These goals are more readily obtained than the more abstract activity-level motives (to disambiguate an OO object from an AT object, we will use the term motive for the latter). When conflated with the single roles that work to satisfy each goal and also the list of all CIs both required and produced in the course of that work, these are labeled Goal-Driven Actions (GDAs).

As they will connect together in transactional chains, GDAs may be envisaged as the nodes of a directed graph, with CI transactions as arcs. For convenience, this may be rendered as a GDA adjacency matrix (GDAAM).

Goals, however, only exist at the action layer of AT. Activities are driven by more abstract and somewhat strategic motives which ATSA envisaged as *coherent sets of consistent goals*.

To assist in identifying activities, GDAs may be broken into component pieces and conflated following AT precepts. The analyst decomposes each GDA into a number of Single Instrument Nodes (SINs) equal to the number of CIs it receives and outputs. Each SIN inherits its parent GDA's goal and constraints. AT requires that each activity has just one role as the doer and a single motive (coherent set of consistent goals), so SINs are sorted first by role and then into groupings of consistent goals. Some subjective judgment on the part of the analyst is indicated here, but it is informed by an understanding garnered during elicitation and is at, worst, no broader an application of experience than that called for in any other method or approach (experience suggests it will be less). Under ATSA, the clients may still actively participate and assist as required, as all ATSA models, including the final activity network, are consistently expressed in comprehensible terms.

Detailed piecewise deconstruction of GDAs into SINs goes some way toward revealing duplications, inefficiencies, and ambiguities; however, it has been found that with experience, clustering of GDAs to activities directly may be possible. The SIN sorting technique might in fact be used topically, as a check. For learners, SIN sorting serves as “training wheels.”

A growing familiarity with their client’s business processes can tempt analysts to institute changes prematurely. Instead, they must refrain and record any such “good ideas” for consideration in the next phase.

Activities identified and conflated from the coherent grouping of GDAs (or their SIN fragments) will each have numerous CIs, both in and out, often connecting them together. The resultant activity network may be illustrated as a directed graph and/or represented as an adjacency matrix known as the Combined Activity Table (CAT).

4.3 ATSA ReDesign

In consultation with the stakeholder as needed, the analyst seeks to reconfigure and rearrange the client’s processes to enhance efficiency and to relieve the roles of burdensome lower-order doings (which have simple drivers such as conditions or simple goals).

Any “good ideas” noted during preceding phases may now be considered as options for changes. Further suggestions for rationalization of the network can be found using Node Reduction Heuristics (NRHs) which were described in detail elsewhere [3].

The complete activity network from the analysis phase will highlight the interactional consequences of any changes and thus prompt more complete and carefully considered decisions.

In its procedural mode, ATSA then envisages a central system which mediates *automatable* activities, holds instrument values, and conducts transforms upon them as required. The specification produced takes the form of a data dictionary and a series of transformation descriptions with constraints and conditions recorded (wherever possible) in structured English (pseudocode).

5 ATSA:OO

ATSA:OO follows the first two phases of ATSA exactly as described above. It also encourages redesign and refinement in the same way as ATSA, drawing upon “good ideas” and the NRHs. To conform to the OO paradigm, ATSA:OO does not collapse

Table 1 Elements of ATSA compared to additional and replacement ATSA:OO elements

ATSA	ATSA:OO
Positions, roles, CIs	
Goal identification	
GDAs SINs	
Conflation of activities	
Initial activity network	
NRHs, good ideas	
Client input, refinement	
New activity network, CAT	
Automate activities	Apply class finding heuristics
Bend instrument paths to system	Identify classes, attributes, and methods
Convert temporal rules to deontic	Model-0 class diagram
Data dictionary	
Pseudocode transforms	

automatable activities into an amorphous assortment of data instruments and transforms. ATSA:OO retains all redesigned activities and transactions in their networked form, which remains comprehensible to the client. It then seeks to identify coherent classes and their associations from this modified network. Table 1 shows the new elements of ATSA:OO in comparative sequence.

Since an activity represents a coherent unit of work, conducted by a role, to satisfy a coherent set of consistent goals, it is a good candidate to be the basis of a class. OO requires that a class has both methods (work done) and attributes (essentially a list of its variables), so we require that any class identified from within an activity network must consist of at least one activity and at least one instrument.

Classes will tend to be more complex than simple pairings of singleton activities and instruments. Figure 1 below shows the complete activity network for the coffeemaker problem. It also indicates identified classes by using grey zones encasing each of them. We offer four rules to guide the clustering of activities and instruments into classes. These are described below.

5.1 New Class Finding Heuristics

Since ATSA has already identified data-like and function-like constructs, namely, CIs and activities, the primary problem with identifying classes from the activity network is determining which instruments should be clustered with which activities. To address this problem we devised the four heuristics.

At each application of these heuristics, we first attempt to form one or more clusters (proto-classes) under the principle of that heuristic. Where this is not possible (for want of at least one available activity and available instrument), we extend one

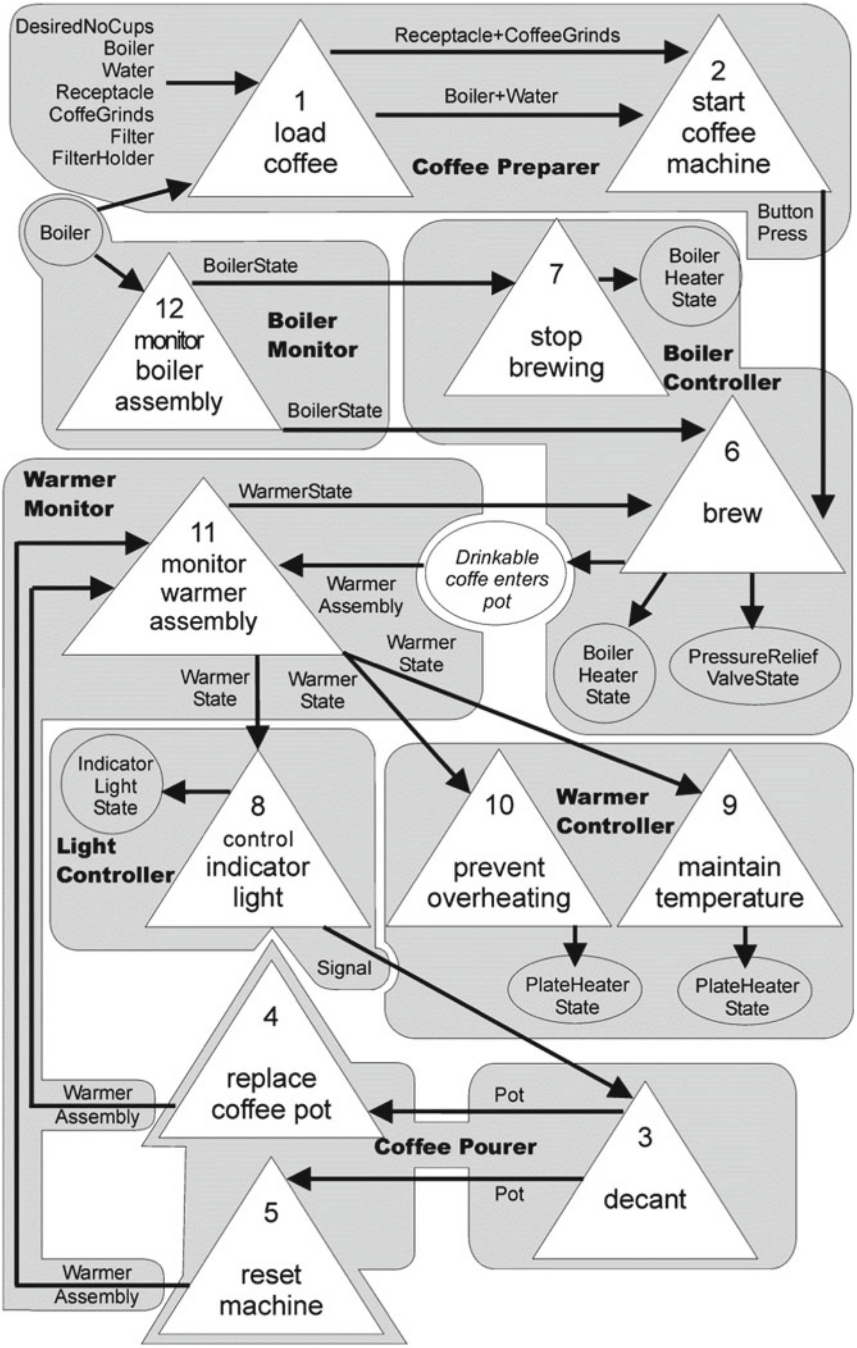


Fig. 1 Activity network for the coffeemaker with classes identified

or more of the existing proto-classes. Priority is given to proto-classes suggested by preceding heuristics, over those suggested by later heuristics. That is, later heuristics cannot de-cluster proto-classes formed under preceding heuristics.

Rule 1

Cluster "source" activities with their output instruments.

If an activity is responsible for outputting a given instrument to multiple other activities, we cluster the activity with that instrument. When many entities obtain data from a single source, good design practice says that the data should reside with the single source rather than with the many recipients.

In Fig. 1, activity 11 is a source activity which outputs multiple copies of the WarmerState instrument to different activities. Rule 1 therefore clusters activity 11 with the WarmerState instrument. This is sufficient to form a proto-class.

Similarly, activity 12 is clustered with the BoilerState instrument and activity 3 with the Pot instrument, each forming their own proto-class.

Rule 2

Cluster "sink" activities with their input instrument(s).

If an activity receives copies of an instrument from multiple other activities, we cluster the receiving activity with that instrument. This rule recognizes that rather than having the data reside with one or other of these multiple sources, a single copy of the data should reside with the recent entity, which should permit those source entities to set the value of that data.

In Fig. 1, activity 11 is the recipient of the Warmer Assembly instrument from multiple activities. Rule 2 would therefore cluster activity 11 with the WarmerAssembly instrument; however, activity 11 has already been clustered into a proto-class by an application of Rule 1 and is thus not available. The instrument WarmerAssembly is insufficient to form a proto-class on its own (lacking an activity), and thus, it is clustered with the existing proto-class (which already contains activity 11 and the WarmerState instrument).

Rule 3

Cluster activities that input the same or similar instruments and that also output the same or similar instruments, with those instruments.

This Rule seeks to cluster according to a commonality of process (based on both data and function). Two or more entities which act upon variations of the same input data, producing variations of the same output data, are variations of essentially the same function type and therefore should be clustered together with their input and output instruments.

In Fig. 1, activities 6 and 7 both receive variations of the BoilerState and produce various changes to BoilerHeaterState. BoilerState has already been clustered; however, BoilerHeaterState is still available. Thus, it is still possible form a proto-class consisting of activities 6 and 7 and BoilerHeaterState.

Similarly, activities 9 and 10 are clustered with PlateHeaterState, though WarmerState was unavailable.

Activities 4 and 5, however, have no available instruments for clustering as Pot and WarmerAssembly have each been clustered under the preceding heuristics. They are thus unable to form a proto-class by themselves and must therefore be assigned to either the proto-class containing activity 11 and WarmerAssembly or the one containing activity 3 and Pot. In this situation, we are guided by roles identified in the ATSA analysis phase. Activities 4 and 5 are conducted by the same role as activity 3, so we decide to cluster them together.

Rule 4

Consider unclustered activities and their associated instruments.

After the application of the preceding three heuristics, a number of isolated activities and/or instruments may remain unclustered. For these, we rely on the roles and motives identified under ATSA and the intuition of the analyst.

In Fig. 1, activities 1 and 2 are clustered with their outstanding input and output instruments due to a commonality of role and motivation identified under ATSA.

The only remaining unclustered entities are activity 8 and the IndicatorLightState and Signal instruments. These cluster into a proto-class. As an extremely simple class, this may later be absorbed into another class, but this consideration would arise under OOD.

5.2 *Identify Classes, Attributes, and Methods*

At the completion of the class finding heuristics, we have a number of proto-classes which can now be converted into classes. Class names are drawn from the roles and goals of the component activities, in the form of a new role name. For example, under initial analysis, activity 11's goal was to monitor the state of the WarmerAssembly, performed by a PressureSensor role. The class containing activity 11 and the WarmerAssembly and WarmerState instruments is given the name Warmer Monitor.

Attributes of classes are simply the instruments belonging to the proto-class. The methods are drawn from either the component activities (as in the case of the Coffee Pourer class) or their constituent GDAs (as in the case of the Light Controller class).

5.3 *Model-0 Class Diagram*

With the classes defined, all that remains is to determine the associations between the classes, and these can be directly inferred from the instrument transactions which cross the proto-class boundaries in the activity network. For example, the ButtonPress transacted between activities 2 and 6 indicates that the Coffee Preparer

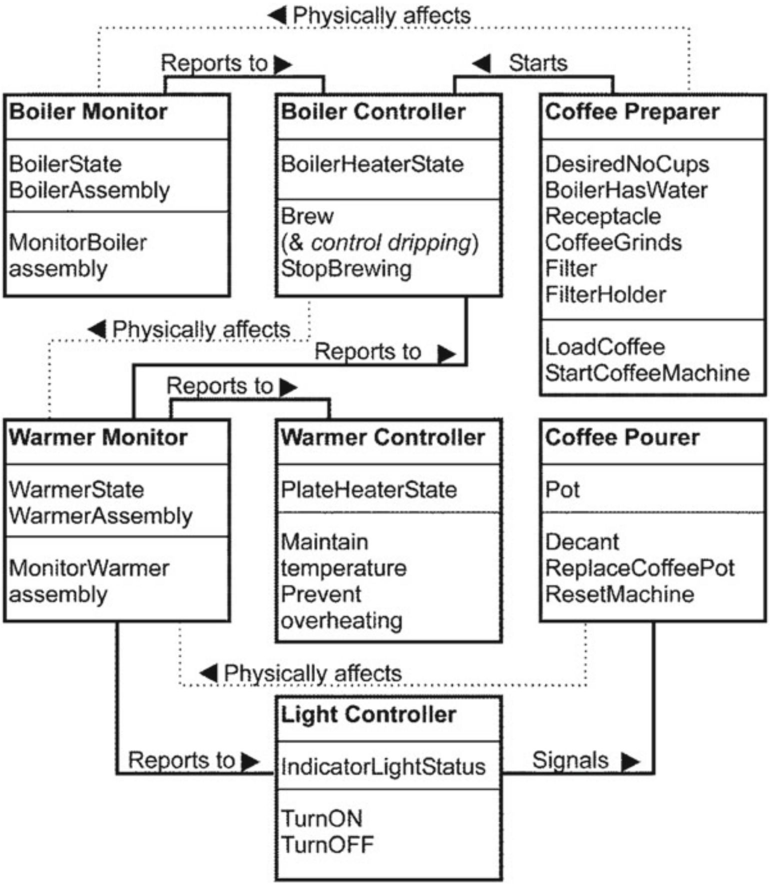


Fig. 2 Model-0 of the coffeemaker offered by ATSA:OO

starts the Boiler Controller, while the WarmerState transacted between activities 11 and 6 suggests the Warmer Monitor reports to the Warmer Controller (assuming a subscription model). Our resultant model-0 is shown in Fig. 2.

6 Comparison of ATSA:OO with OOA

A worked solution to the coffeemaker problem was offered by Weirich [14]. Figure 3 shows his model-0, resulting from OOA Use Case techniques. We can see that the model-0 from ATSA:OO in Fig. 2 is significantly more cohesive.

Figure 3 has very poor cohesion; it has a central "God" class, a mistake common to neophyte practitioners. It would require considerable additional effort to reform into workable cohesive classes.

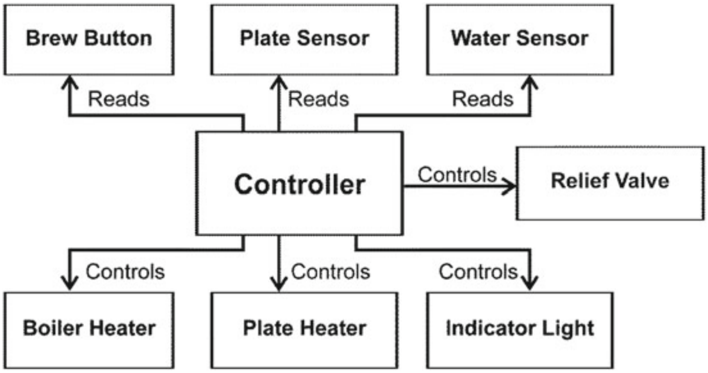


Fig. 3 Model-0 of the coffeemaker offered by traditional OOAD (from Ref. [14])

Figure 2 does not have this problem. Fully decomposing to SINs then reclustering generates distinct activities. ATSA:OO identifies the smallest feasible classes resulting from those activities, so it is highly unlikely to produce a "God" class.

Furthermore, because all activities are determined through clustering by role and goal, the resulting suggested class methods are functionally cohesive. As the ATSA:OO heuristics cluster activities with their associated instruments, each class is likely to have high informational cohesion (and therefore reduced coupling).

Additionally, the OOA model-0 only has class names whereas the ATSA:OO model-0 has suggestions for possible attributes and methods.

Both the OOA and the ATSA:OO model-0 show class associations. However, neither depicts composition or inheritance, as these more abstract relationships are expected to be identified during the OOD phase.

7 Conclusion and Future Work

ATSA:OO is an adaption of the ATSA method to suit the widely adopted OO paradigm. It is intended to replace the OOA phase, with a method that is easier for neophyte practitioners to understand and use and has a strong theoretical basis.

If even an experienced industry practitioner, employing Use Case analytical techniques under OOA, can produce a model-0 with a controlling "God" class and lacking both attributes and methods; then a neophyte practitioner can hardly be expected to use OOA with confidence. ATSA:OO's application of the precepts of AT yields classes with greater cohesion, ensuring that neophyte practitioners can have confidence in a stronger model-0, requiring less effort in the design phase.

There is substantially less reliance upon the intuition of the analyst, as ATSA:OO's model-0 arises from a straightforward application of simple steps. Each of these steps is sufficiently tightly prescribed to prevent seesawing between potential models in the analysis phase.

ATSA allows the client to comprehend the design process all the way through to the activity network. ATSA:OO continues this by clustering the activity network directly, allowing client comprehension all the way to the class diagram.

This paper considered the feasibility of ATSA:OO using a very simple example. Further examination of this adapted method will consider more complex design problems. The clustering rules of both ATSA and ATSA:OO appear to result in highly cohesive classes; quantitative evaluation of the cohesion found in code built under ATSA:OO analysis is required to measure this. Similarly, quantitative and experiential tests with groups of neophyte practitioners will measure the ease with which the method can be learned and applied.

References

1. Brown RBK, Piper IC (2011) What users do: SA&D with the ATSA method. 20th International conference on information systems development (ISD2011). Edinburgh, Scotland. Accessed 24–28 Aug 2011
2. Brown RBK (2010) The ATSA method for specifying both system and user interface requirements, PhD thesis, University of Wollongong, Australia
3. Brown RBK, Hyland P, Piper IC (2006) Specifying system requirements using the 5S method ACIS 2006 proceedings. Paper 100. http://aisel.aisnet.org/acis2006/100
4. Coad P, Yourdon E (1991) Object-oriented analysis, 2nd edn. Prentice-Hall, Englewood Cliffs, NJ
5. Crear J (2009) CIO Standish Group. http://www1.standishgroup.com/newsroom/chaos_2009.php. Accessed 24 Mar 2010
6. Engström Y (1987) Learning by expanding: an activity-theoretical approach to developmental research. Orienta-Konsultit Oy, Helsinki, Finland
7. Holmboe C (2004) A Wittgenstein approach to the learning of OO-modeling. Comput Sci Educ 14(4):277–296
8. Leont'ev AN (1978) Activity, consciousness, and personality. Prentice Hall, Englewood Cliffs, NJ
9. Martin RC (1995) Designing object oriented C++ applications using the Booch method, 1st edn. Prentice Hall, Englewood Cliffs, NJ
10. Martin RC (2003) UML for Java™ programmers. Prentice Hall, Englewood Cliffs, NJ
11. Stevens WP, Myers GJ, Constantine LL (1974) Structured design. IBM Syst J 13(2):115–139
12. Svetinovic D, Berry DM, Godfrey M (2005) Concept identification in object-oriented domain analysis: why some students just don't get it. 13th IEEE international conference on requirements engineering (RE'05)
13. Vygotsky LS (1978) Mind in society. Harvard University Press, Cambridge, MA
14. Weirich J (2004) OOAD design problem: the coffee maker. www.csun.edu/~renzo/cs680/coffeebrewerdesign.pdf. Accessed 20 Feb 2012

Information Systems for the Social Well-Being of Senior Australians

Lois Burgess, Helen Hasan, and Carole Alcock

Abstract In most developed countries, the ageing population is placing a severe strain on health systems and national budgets. In meeting this challenge, e-health initiatives seek medically focused ICT solutions for improved health services that enable senior citizens to remain living at home longer. A literature review and interviews with healthcare providers reveal that significant factors affecting the well-being of the elderly are isolation and loneliness. In this paper we report on the first stages of a study on the digital literacy of seniors and the potential development of social media to meet the capabilities of people as they age. Focus groups with senior citizen computer club members tell us that older folk are becoming more digitally literate. As social media becomes more usable and acceptable, we propose that online communities for senior citizens may have significant health benefits. As this is a multifaceted emergent phenomenon, we apply concepts from Complexity Theory to our analysis and to the design of future research on this topic using an action research methodology.

1 Introduction

In Australia, as in most of the developed world, the ageing population is creating challenges for the adequate provision of affordable healthcare for increasing numbers of senior citizens at home, in self-managed units or in residential care. The ageing population is placing severe strain on the health system and the nation's budget. E-health is perceived as a means to provide a high level of online medical service while reducing the economic burden and enabling the elderly to remain at home longer. There are also enormous benefits to be gained through e-health for those

L. Burgess (✉) • H. Hasan • C. Alcock
University of Wollongong Australia, Wollongong, NSW, Australia
e-mail: lburgess@uow.edu.au; hasan@uow.edu.au; caroleal@uow.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_7,

living in regional and remote areas. Significant effort has been put into research and development of technical solutions to medical issues including monitoring devices and Internet-based consultations. Less research has been conducted into how to enhance the emotional and social needs of the ageing population.

Our research is motivated by literature reporting that loneliness and isolation are among the main problems encountered by folk living well into their 1980s and 1990s. Those still in their home receive basic medical and support services through either e-health systems or professional visits, but their lack of mobility restricts their ability to interact socially. Those who move into independent self-managed units or full-residential institutions are often dislocated from family and friends.

Although they did not grow up with digital technologies, those now in their 1960s and 1970s are increasingly more computer and Internet savvy. Many have been active users at work or elsewhere, while others in retirement are catching up, using email, social media and online services to connect with family and friends and access information. It is not clear, however, how this will equip them for the later stages of life when they will be increasing more reliant on the Internet for their social interaction but have reduced physical and mental capability.

The exploratory study reported here seeks answers to two related questions. To what extent can the elderly use information systems to address issues of loneliness and isolation? How can socio-technical systems be established using emerging Web 2.0 applications to meet the disparate needs of isolated elderly users? By means of literature review, interviews with aged care providers and a focus group with members of a seniors computing club research explore the ways in which senior citizens currently participate in online communities, their capabilities for doing so and the potential benefits of involvement in online communities for isolated senior citizens. As this is an emergent phenomenon, we explain how we apply concepts from Complexity Theory to our analysis of this exploratory phase of the research. We conclude the paper with the design of future research on this topic using an action research methodology.

2 Literature Review

2.1 Isolation and Loneliness

Despite the global interconnectedness provided by the Internet, isolation and loneliness are an increasing social problem. Norrie [26] reports that isolation and loneliness pose an increasing threat to the health of Australians, many of whom are cut off from friends and locals by ill-conceived urban design. According to research cited by the report between 1984 and 2005, the percentage of Australians who said they did not have any trusted friends rose from 4 to 6 %, while the percentage who felt they could not call on locals for help increased from 11 to 13 %. The figures are expected to worsen with the rise in the number of single-person households.

The report concludes that "it's now recognised that loneliness is up there with high blood pressure, lack of exercise, obesity, and smoking as an indicator of shortened life expectancy".

Isolation appears to be a universal problem in ageing populations [14]. One UK digital media champion, Martha Fox, expressed concern that "over three million people aged over 65 don't see another person in a week" and believed that technology could help address this problem [8]. This is supported in a study on the use of online communities by older people, which found a positive association between a sense of belonging to an online community, sense of community and link to well-being [32]. Coughlan [11] also describes the particular health risks for the elderly in England where more than half of people over the age of 75 are living alone and about one in 10 says he or she suffers from "intense" loneliness. A lack of social interaction can make old people more vulnerable to depression and to problems such as excessive drinking, poor diet and a reduction in exercise. A study of 1,300 people aged 65 and over [17] found that 35 % of participants were lonely, with 9 % describing it as painful and 6 % as intrusive. Thirty-four percent had a nonintegrated social network with well-being, depressed mood and hopelessness, all independently associated with both loneliness and nonintegrated social networks. While younger people routinely make use of social media to stay connected, it is often assumed that the elderly are not inclined to do so, particularly those who have never used computers. However, we suggest that this is changing.

2.2 *Digital Literacy of the Elderly*

There has been a strong focus on teaching seniors basic Internet skills and there is considerable literature relating to this. The AHRC [4] Report emphasises education, training and provision of appropriate technology to achieve effective Internet access. Parry [29] states that teaching mobile web literacy is as important as teaching basic literacy. Several research papers [19, 33, 34] focus on training seniors in Internet use. Internet literacy programmes for older users are regarded as an important first step, but more is required.

Recent Australian research into Internet use by female seniors [28] pointed to the need for access to training, technical assistance and information as important for this group. Palmer [28] refers to earlier studies of elderly participants suggesting that lower levels of education, lower income levels and workforce participation might be factors contributing to their lower usage of Internet-based services. The most common use was for email. Lack of knowledge was perceived as a major barrier. Some had had bad experiences and felt disillusioned. Lack of services and not knowing where to find help were a concern. Users had a generally positive view of the Internet's impact on well-being, while non-users thought it would have a negative effect. A concern expressed in the report's conclusion was that, while, in this study, non-users experienced little inconvenience at present, they were concerned that lack of access might be a problem in the future [28]. Training aimed at improving digital literacy skills was among the report's recommendations as was consultation with seniors on e-government services.

In a small Australian study, Mellor et al. [23] considered a group of twenty residents in a nursing home, who reported a generally positive experience from the use of computers. As there were discrepancies in the findings, it was suggested that the impact on well-being was more complex than could be inferred from these results. This clearly warrants further study among a wider group.

Access to information services by the elderly is another area of importance. One report [4] expressed the view that digital technologies offered the potential for providing greater access to both government and business information services. Improved access to education and services at locations and in formats that otherwise might have been difficult to access was also stressed. Concern was raised regarding equity of access, with some older Australians not enjoying the standards that others take for granted. The report highlighted the potential for both older Australians and those with disabilities to extend their access to information services of every kind due to the inherent nature of digital technologies.

Designing systems with a variety of interfaces that older people will use seems a logical objective, although some researchers have suggested that age is less a discriminating factor than is commonly thought. In a study by Chung et al. [10], factors affecting perceptions of online community participation were investigated using the Technology Acceptance Model (TAM). One finding was that "the relationships among perceived ease of use, perceived usefulness and intention to participate in online communities do not change with age". Ease of use was no more important to older users than it was to younger users. These findings have implications for developers. Older users were concerned about privacy issues and this might be a factor influencing their participation in social media sites. Ease of navigation and content relevant to the target audience regardless of age were important. Perceived quality of sites and web design are, it seems, important to older and younger users alike.

2.3 E-Health, Seniors and Social Media

When considering the needs of older Australians, much focus has been directed towards e-health to assist at-home, isolated seniors, particularly the over 1980s. This is not surprising given the potential of developments in this area. In his reports on Australia's Digital Economy, Paul Budde [6, 7] describes e-health as potentially one of the "key killer apps linking e-health developments to the National Broadband Network", two significant areas being early diagnosis and posttreatment patient monitoring provided at home.

It is clear from the Australian demographic that e-health is important. In the recent government report "Australia 2050: future challenges" [2], the emphasis is on "an ageing and growing population". Consequent on this, a significant rise is predicted in aged care spending, related to the increasing number of people over 85, likely to be in residential care or with disabilities requiring special care. E-health applications may help to reduce the need for residential care. The report states that social media may hold some promise in this regard.

Social networking is seen as useful in combating loneliness and isolation: “Ultimately, virtual communities can combat social isolation by offering a new channel for older Australians to form support networks, maintain contact with friends and family, and participate in different parts of the community” [3]. A recent study suggests that social media collaborations in relation to specific diseases (in this case, Crohn’s disease among teens) can be extended to many of the diseases of older age groups: diabetes, heart disease, psoriasis and some cancers [18].

Reports on use of the Internet and social media in the USA in 2010 [1, 21, 30] indicate that the average usage of Internet social media among users over 50 had nearly doubled in the preceding year, from 22 to 42 % of those surveyed. Among those using social networking sites, the findings recorded 47 % of 50–64-year-olds and 26 % of those 65 years and older. Numbers using MySpace, Facebook and Linkedin had grown by 88 % among 50–64-year-olds and 100 % among older users (65 and over), whereas growth among younger users (18–29) (already representing 86 % of all users) was only 13 %. Email was still the main medium of communication among older users.

The literature reports conflicting findings on the usefulness of social media in overcoming isolation, loneliness and depression. In their 2006 UK study, Dickinson and Gregor [12] suggested that “computer use had no demonstrated impact on well-being in older adults”. Participants in a study conducted by Erickson in 2011 [14] reported mixed opinions on the value of Facebook as a “community”, with most seeing it as a medium to passively view what others were doing. A more recent Canadian study of seniors over 60 [15], however, suggested that the Internet promoted “higher perceptions of self-efficacy” among seniors using the Internet regularly over those who did not. Similar findings were reported in a US study [16, 31], which reported positive results, with 20 % reduction in depression, among elderly American Internet users. This study looked at survey responses from 7,000 American retirees, aged over 55 years. It concluded that there were significant health outcomes to be had from expanding Internet adoption in this age group, with potential economic savings as well. Etchemendy et al. [13] claim that ICT can be helpful in reducing negative moods and isolation and thus can be efficacious and effective resources for improving the quality of life in elderly. According to Norval et al. [27], research suggests that communication platforms that complement existing telecare systems could be designed to provide significant emotional support for older adults. However, it is clear that this is still an under-researched area where further work is needed [14].

The study that we have found to most closely relate to our work is one by Ballantyne et al. [5] who describe a qualitative pilot project that implemented an Internet social networking intervention with six older people aged 69–85 years recruited from a community aged care programme. The researchers evaluated the effect it had on older people’s experience of temporal loneliness. The findings from this project demonstrated that the utilisation of a social networking site has the potential to reduce loneliness in older people. Four major themes emerged from their inductive analysis of the data which were the participants' experience of loneliness, technology as an enabler, providing a supportive environment and connectivity. We are encouraged not only by the positive results of the study but also by the inductive approach taken, which allowed for emergence of issues and outcomes.

3 Theoretical Underpinning

Our literature review demonstrates the complex and multidisciplinary nature of our topic. Most e-health research follows a traditional scientific method of enquiry that is well ordered as it should be. Most websites for seniors are currently created top-down where government or civil "experts" train, advise or organise older people who are assumed to have low digital skills and knowledge. In contrast, what appears to be needed to overcome loneliness is to engage seniors in meaningful online communities and activities that are created bottom-up by the senior citizens themselves with assistance, facilitation and encouragement where needed. We thus turn to concepts from Complexity Theory such as self-direction, self-organisation and emergence.

Complexity Theory is concerned with the behaviour over time and space of complex systems. Complexity science deals with complex adaptive systems that are fluidly changing collections of distributed interacting components that react both to their environments and to one another. Complex systems have a history; they evolve and their past is co-responsible for their present behaviour. Elements in the system can be ignorant of the behaviour of the system as a whole responding only to what is available to it locally.

Within complex contexts principles of emergence, co-evolution, pattern formation, self-direction and self-organisation are critical for understanding. In complex environments, effective change and growth should not be mandated but encouraged and facilitated through the introduction of appropriate attractors, incentives, diversity and the setting of flexible and porous boundaries. Promising emergent forms and patterns of behaviour should be rewarded and supported even when these are not what were originally anticipated [20].

The literature suggests that the different capabilities, experiences and needs of the elderly will require a range of different technical and social solutions to the problems of isolation and loneliness. In the exploratory research presented here, we provided evidence of the complexity of this problem and the suitability of evoking Complexity Theory to indicate that simple imposed solutions are unlikely to be readily taken up by those in need of social connections. We will therefore look for solutions to emerge through a complex co-evolution of human and technical capability driven by those involved, namely, the elderly, their families and carers. The principles of Complexity Theory will enable us to better understand this process.

4 Research Approach

Our research began with the review of literature summarised above and sources from a number of different fields: isolation and loneliness, contributions to the social well-being of isolated senior citizens, IT and the elderly, social media, Internet access and usability. There is little intersection between research in these different areas, and we bring them together to inform our work through the lens of Complexity Theory which we have also described above.

Our exploratory research has been conducted in the following stages:

– A study of websites and online facilities created for and used by Senior Australians
– Informal interviews with carers and service providers in the Aged Care Sector to determine from their experience the social needs of the elderly and potential benefits of online social interaction among senior citizens
– Focus groups with senior members of a local computer club on their use of the Internet and attitudes to social media

We describe the conduct and results of this research below. The setting for this study is urban and regional Australia, which, as a Western developed country, has a high level of IT and Internet use and the challenges of an ageing population. This is an opportune time to conduct a study in this area as a national broadband network is currently being rolled out to provide a high-quality infrastructure to support new social multiplatform media. Our discussion uses the findings of this preliminary exploratory study to develop a research design for a wider study involving interventions in the form of the establishment, facilitation and assessment of online communities driven by the needs of groups of senior citizens.

4.1 Observations of Senior Australians Online

The first stage of the study involves an examination of the provision of ICT for seniors and its use and acceptance in Australia. Our exploratory study of existing approaches to online sites for seniors indicated that most were currently created top-down where government or civil “experts” train, advise or organise older people who are assumed to have low digital skills and knowledge.

The Australian Government provides a website, *seniors.gov.au*, for senior citizens and associated activities. This site reflects a typical bureaucratic top-down approach that sees the government as having the responsibility to inform and educate its citizens with limited community engagement [9]. Much of the seniors’ website is aimed at 50–70-year-olds and assumes that most older people have a limited level of digital literacy. In Australia there are 160 public seniors’ computer clubs which have a similar view and a focus on training, mentoring, technical support and provision of advice on security, privacy and awareness of IT pitfalls (http://www.ascca.org.au/). A typical club description is as follows:

> Brisbane Seniors Online Inc is a not-for-profit organisation made up of volunteers who assist Brisbane's seniors to become competent with computers and to use them online. Most mentoring is done on a one-to-one basis in people's homes if they so choose. Since BSOL was formed in November 2000, our volunteer mentors have taught, or are teaching, some three thousand people over the age of 50 to become computer-literate—to access the Internet, send and receive emails and use word processing.

The Australian Seniors Computer Clubs Association (ASCCA)'s Google group (http://groups.google.com.au/group/cyberascca) states that the main problem seniors face is the question of "how do I"? They offer an online help desk where members can post their questions and the experts of the Google group, along with other members, can offer suggestions on how to fix it. The University of the 3rd Age (U3A) takes a similar simplistic view offering short online courses and many other useful resources for older people, especially those who are geographically, physically or socially isolated. The U3A Online website provides up-to-date contacts for all Australian and New Zealand U3As as well as facilities for their members to exchange ideas, resources and information about regional U3A events. The information, including news items and links to many interesting websites, will also prove useful to other groups of older people.

With the move towards social networking, the ASCCA are investigating the use of the social networking website www.finerday.com which is more senior friendly than Facebook. The Australian Government seniors' website is also beginning to try some online community-building initiatives although most are not yet active. The Australian Seniors site says:

> In addition to providing a huge range of factual information relevant to older Australians, *seniors.gov.au* provides a wide range of exciting community-building features. These features enable you to interact with other people with similar interests and ensure that this online resource reflects the opinions and addresses the issues of direct relevance to Australians over 50.

Features planned in the development are "Living History" (stories of where people were and what they were doing during a significant event in Australian history), a "Q&A Forum", a "Meeting Place" (where topics of discussions could include such things as travel, genealogy and entertainment) and even "Online Games". Although these features are more creative, there is still a top-down approach where the government controls, directs and manages the agenda.

In an attempt to find emergent communities of seniors, we found Australian Seniors on Linkedin with 17 members and some recent discussion and an AustralianSeniors page on Facebook that had 43 likes but very little content. These seemed to be driven by a few individuals but not yet with a critical mass to be called a community. This suggests that while online communities of seniors may follow similar patterns of other such groups, there is a need to investigate where they may differ and have more specific requirements and challenges.

4.2 Advice from Those Working in the Aged Care Sector

As part of a set of interviews conducted during a separate research project, carers and service providers in the Aged Care Sector were asked for their comments and advice on our proposed project. Their responses confirmed the following:

– That the emphasis on current e-health research was on technical and medical issues
– That loneliness was a major problem as people aged and had a significantly affect on their health and well-being, often leading to depression
– That IT and the Internet could be used in a solution to the isolation that often increased as people aged but that there would have to be a variety of systems needed to suit people in different circumstances and conditions

Many of those interviewed belonged to a Dementia Support Network and indicated that dementia patients would be a particular challenge. However, they also noted that as people age a whole range of special needs develop which may impact on their ability to use a standard computer. This provides evidence for the complexity of the problem.

5 The Digital Literacy and Attitudes of Senior Citizens

In order to gauge the potential for senior citizens to use social media and assess their attitudes to doing this, we held a discussion on this topic with members of a Seniors Computing Club at their invitation. In keeping with our intention to take an exploratory approach to this phase of the research, we followed the nondirective, "affinity group discussion" approach of Hugh McKay, a renowned social commentator in Australia. This method [22] uses a naturally existing social group (in our case members of the computing club) meeting together in their natural habitat (the club's monthly meeting), to engage in informal and unstructured conversation about the topic of the research. An inductive analysis of the audio recording of this conversation involved the identification of themes and individual perceptions by the researchers and revealed the following:

Several of the members, mostly those who had used computers for years and were respected as experts by other members, were quite antagonistic towards the use of social media. They saw applications such as Facebook and Twitter as trivial exercise in social banter. As one said "I can't see the sense of it, my time is too valuable to waste on that" and "if my daughter wants to share a picture with me she should just attach it to an email". Others, however, began to describe how they used Facebook and Skype to keep in touch with family. Another called himself "a stirrer" and saw how blogging and online discussions would be attractive for him. Another advocated using Internet tools to put his life story online for his descendants to read once he is gone. One very elderly gentleman then described his extensive use of social media for all sorts of activities that he combines with his volunteer work and interests. Right at the end one elderly lady describe her experiences with isolation and said that if online communities could help, she would be all for it. Lots of the general discussion described how they would come across different applications on the Internet and see how they could be useful for something they were doing. This demonstrated the emergent nature of how things happen in this regard and support our intention to follow a “bottom-up” approach to the interventions we plan for the next phase of our research.

6 Discussion of Next Steps

As indicated above our research seeks answers to questions on the potential use of information systems in meeting the social needs of isolated elderly users. The results of the literature review and exploratory study described above, combined with our understanding of emergent processes from Complexity Theory, confirmed our commitment to the following propositions in answering those questions:

- That overcoming loneliness and isolation experienced in old age would have positive benefits for the national health system
- That one component of the e-health programme could be the use of the web to connect isolated elderly and overcome the negative effects of loneliness
- That this approach would need to take a bottom-up approach to engage citizens in the process
- That a diverse range of socio-technical systems and community activities would emerge as the programme progressed
- That Complexity Theory constructs can underpin research in this area

According to Kurtz and Snowden [20] and Mitleton-Kelly [24], the processes of emergence and co-evolution as understood by Complexity Theory cannot be planned but can be encouraged through the imposition of “attractors” with suitable “boundaries”. Precise outcomes cannot be known in advance but positive outcome are likely if the attractors and boundaries are chosen well. We will thus follow an action research methodology for the next phase of this research where interventions will be carried out to provide opportunities, advice, incentives and resources (i.e. attractors) among groups of Senior Australians to form online communities bounded by the technologies that suit them and are available. A “bottom-up” approach will be followed to engage participants at all stages and the emergent outcomes will be studied, evaluated and reported.

The creation of new communities is best understood by the principles of complexity as diversity, creativity, self-organisation and unanticipated outcomes are key elements in the cocreation of an innovative environment. The distinguishing characteristic of complex co-evolving systems is their ability to create new order. In human communities this may take the form of new ways of communicating, acting, thinking and relating or even the creation of a different culture or a new organisational form [24, 25].

In our study the application of Complexity Theory can guide our analysis of the emergence and facilitation of self-directed online communities of senior citizens as well as a participatory emergent approach to evolution of the diversity of suitable tools and their interfaces. Members of such communities require a level of digital information literacy beyond simple computing skills to the ability to *access*, *interpret, connect, manipulate, visualise* and *create* digital content. The computer itself has become almost ubiquitous so that the traditional concepts of usability and acceptance give way to those of real engagement and emersion in a merged interconnected physical and virtual world.

We will therefore apply concepts of Complexity Theory to our research design of interventions involving online communities of senior citizens and relate these to their subjective well-being which Wiesmann and Hannich [32] have developed as an indicator for successful ageing. These authors adopted the perspective that a sense of coherence plays a key role for psychological adaptation. The findings corroborate the idea that the sense of coherence (i.e. integrated and holistic) creates, or maintains, a form of psychological integrity as represented by subjective well-being. Our research will take this idea into the digital online environment.

7 Conclusion

The literature review and early findings of this research suggest that the use of social media by senior citizens may be beneficial to their well-being, overcoming isolation and loneliness as they become more housebound. The creation of online communities that would engage the elderly is however unlikely to succeed if imposed on them, and so we turn to concepts from Complexity Theory such as emergence to guide proposed research in this area.

References

1. Anon (2010) Older adults' use of social media nearly doubles. Telecomm Rep 76(18):34
2. AGD (2010) Australia to 2050: future challenges. Attorney-General's Department Commonwealth of Australia, Canberra
3. ADHA (2011) Older Australians and the internet: bridging the digital divide. National Seniors Productive Ageing Centre (NSPAC), Australia Department of Health and Ageing, Perth
4. AHRC (2000) Accessibility of electronic commerce and new services and information technologies for older Australians and people with a disability. Report of the Australian Human Rights Commission, Canberra
5. Ballantyne A, Trenwith L, Zubrinich S, Corlis M (2010) "I feel less lonely": what older people say about participating in a social networking website. Qual Ageing Older Adults 11(3):25–35
6. Budde P (2009) Australia: the national broadband network. PressReleasePoint. Posted 4 Nov 2009 by BuddeComm. http://www.pressreleasepoint.com/print/391710
7. Budde P (2010) Australia: the national broadband network. BuddeBlog. Jul 2010. http://www.buddeblog.com.au/frompaulsdesk/australia-the-national-broadband-network
8. Campbell R (2010) Giving it all she's got. Northern Echo. [Darlington (UK)] 23 Oct
9. Chadwick A (2011) Explaining the failure of an online citizen engagement initiative: the role of internal institutional variables. J IT Polit 8:21–40

10. Chung JE, Park N et al (2010) Age differences in perceptions of online community participation among non-users: an extension of the technology acceptance model. Comput Human Behav 26(6):1674–1684
11. Coughlan S (2011) Loneliness is “hidden killer” of elderly http://www.bbc.co.uk/news/education-12324231
12. Dickinson A, Gregor P (2006) Computer use has no demonstrated impact on the well-being of older adults. Int J Human Comput Stud 64:744–753
13. Etchemendy E, Baños RM, Botella C, Castilla D, Alcañiz M, Rasal P, Farfallin L (2011) An e-health platform for the elderly population: the butler system. Comput Educ 56(1):275–279
14. Erickson L (2011) Social media, social capital and seniors: the impact of facebook on bonding and bridging social capital of individuals over 65. Americas Conference on Information Systems, Detroit, USA
15. Erickson J, Johnson GM (2011) Internet use and psychological wellness during late adulthood. Can J Aging 30(2):197–209
16. Ford G, Ford S (2009) Internet use and depression among the elderly. Phoenix Center Policy Paper, Number 38. Oct 2009
17. Golden J, Conroy R, Bruce I, Denihan A, Greene E, Kirby M, Lawlor B (2009) Loneliness, social support networks, mood and wellbeing in community-dwelling elderly. Int J Geriatr Psychiatr 24/7:694–700
18. Interlandi J (2011) Facebook-like portal helps teens with Crohn’s Collaborate on medical research. Sci Am 305(3):66–69
19. Kehoe J, Bednall ETC et al (2009) Training adult novices to use computers: effects of different types of illustrations. Comput Human Behav 25(2):275–283
20. Kurtz C, Snowden D (2003) The new dynamics of strategy: sense-making in a complex and complicated world. IBM Syst J 42(3):462–483
21. Madden M (2010) Older adults and social media: social networking use among those ages 50 or older nearly doubled over the past year. Pew Internet and American Life Project: Pew Research Center. http://pewinternet.org/Reports/2010/Older-Adults-and-Social-Media.aspx
22. McKay H (2011) The unfocussed group discussion technique. Presentation at the IIBSoR Retreat. 6 Jul 2011
23. Mellor D, Firth L, Moore K (2008) Can the internet improve the well-being of the elderly? Ageing Int 32:25–42
24. Mitleton-Kelly E (2005) A Complexity approach to co-creating an innovative environment. World Futures 62/3:223–239
25. Mitleton-Kelly E (2011) A complexity theory approach to sustainability: a longitudinal study in two London NHS hospitals. Learn Organ Int J 18(45):53
26. Norrie J (2012) Loneliness on the rise as our cities atomise. http://theconversation.edu.au/loneliness-on-the-rise-as-our-cities-atomise-6068
27. Norval C, Arnott J, Hine N, Hanson V (2011) Purposeful social media as support platform: communication frameworks for older adults requiring care. ATTACH, Dublin
28. Palmer S (2011) Where do I start? Female seniors and the internet, Council on the aging (WA). Australian Communications Consumer Action Network, Sydney
29. Parry D (2011) Mobile perspectives on: teaching: mobile literacy. EDUCAUSE Rev 46(2):14–18
30. Pew Internet (2010) Spring Change Assessment Survey 2010. Princeton Survey Research Associates International for Pew Research Center. 29 Apr–30 May 2010
31. Phoenix Center (2009) Press release: internet cuts depression among senior citizens. http://www.phoenix-center.org. Accessed 15 Oct 2009
32. Sum S, Mathews R, Pourghasem M, Hughes I (2009) Internet use as a predictor of sense of community in older people. CyberPsychol Behav 12(2):235–239
33. Xie B, Jaeger P (2008) Computer training programs for older adults at the public library. Public Libr 2008:52–59
34. Xie B, Bugg J (2009) Public library computer training for older adults to access high-quality internet health information. Libr Inform Sci Res 31:155–162

A Framework for Evaluating the Impact of High-Bandwidth Internet Provision and Use on Digital Literacy and Social Life Outcomes in Australia

S. Dane, M. Fahey, C. Mason, R. van der Zwan, J. Tucker, D. Bradford, and C. Griffith

Abstract In this paper we present a framework for evaluating the social impact of high-bandwidth Internet provision and use in Australia. High-bandwidth Internet will be provided through a national broadband network to be rolled out gradually and which began in 2011. The framework is based around four key aspects: (1) identifying provision of the national broadband network as an intervention, (2) specification of important outcomes, (3) understanding the behavioural link between intervention and outcomes and (4) conduct of high-quality population-based empirical research. The framework is sufficiently flexible that it can be adapted and applied to various regions of Australia and, depending on the appropriate focus, to the conduct of different types of studies. It is hoped that, by focusing attention on the human behavioural link between high-bandwidth Internet provision and individual outcomes in the general community, we will be able to identify ways of promoting social inclusion and other benefits through appropriate use of the national broadband network.

Abbreviations

ACBI	Australian Centre for Broadband Innovation
CSIRO	Commonwealth Scientific and Industrial Research Organisation
GIS	Geographic information system
G-NAF	Geocoded National Address File
HBI	High-bandwidth Internet

S. Dane (✉) • M. Fahey • C. Mason • D. Bradford • C. Griffith
Commonwealth Scientific and Industrial Research Organisation (CSIRO), P.O. Box 883, Kenmore, QLD 4069, Australia
e-mail: sharon.dane@csiro.au

R. van der Zwan • J. Tucker
Southern Cross University, Coffs Harbour, NSW, Australia

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_8,

NBN	National broadband network
RISIR	Regional Initiative for Social Innovation and Research
TPB	Theory of Planned Behaviour

1 Background

The Australian government announced in 2009 that a national broadband network (NBN) would be established to make high-bandwidth Internet (HBI) available to all Australians. A wholly government-owned company, NBN Co, was created to deliver the roll-out of the network and aims to provide the NBN to 93 % of the Australian population through fibre optic cable with the remainder to connect by fixed wireless or satellite (http://nbnco.com.au).

The Commonwealth Scientific and Industrial Research Organisation (CSIRO) is Australia's national science agency whose mission includes conducting research for the benefit of Australian society, industry and the environment. CSIRO has established the Australian Centre for Broadband Innovation (ACBI) as a collaborative research initiative to develop and demonstrate socially and economically useful applications and services that can use next-generation broadband networks such as the NBN. One of ACBI's main objectives is to develop an understanding of communities' needs and capabilities with respect to services and applications that are enabled by the NBN. To this end, CSIRO and ACBI have allocated approximately $US 3 million over three years to undertake this work. The initiative will draw on expertise from social and economic research areas within CSIRO such as mathematical and statistical sciences, human services evaluation, population health research, energy demand management and economic planning.

CSIRO is also partnering with other research organisations such as the Regional Initiative for Social Innovation and Research (RISIR) Centre at Southern Cross University in Coffs Harbour, to ensure that relevant locally based research is undertaken.

This paper proposes a general framework for evaluating the impact of the NBN on digital literacy and social and well-being outcomes. Section two makes explicit the key components of the framework and states the major objectives of the initiative. Section three details our approach to measurement of connection bandwidth, digital literacy and other life outcomes and outlines the study designs for surveying individuals in our first target populations and planned statistical analyses. The final section discusses aspects requiring further work and considers extensions to other studies and other regions of Australia.

2 General Framework to Evaluate the Impact of the NBN

In developing a general framework, we broadly followed the NONIE guidelines that detail key steps in designing and conducting an impact evaluation [16]. These steps are considered in turn below.

2.1 Identifying the Intervention

The intervention of interest is the NBN. The NBN is to be provided in a staged roll-out that began in May 2011. It is expected to be complete for about one third of the country (approximately 3.5 million households) by 2015 (http://nbnco.com.au). Although universal coverage is planned, individuals may choose either not to connect to the Internet or to remain connected through low-bandwidth information transfer rates. Bandwidth is dependent on factors such as type of NBN connection (fibre, wireless, satellite), distance to Internet service provider (in the case of non-fibre connections) and client-side hardware.

There are two ways in which the impact of the NBN could be evaluated as an *observational* intervention. The first is *provision* of the NBN per se. For example, knowing that the NBN has been provided to one community and not (yet) to another makes it possible to compare the two communities, *on average*, in terms of social and/or life outcomes. The second is *connection bandwidth* measured as a transfer rate (Mb/s) of information. In this case knowing an individual's connection bandwidth makes it possible to estimate the *rate* at which outcomes such as "digital literacy" (*see* Sect. 2.2 below) change per unit increase in bandwidth. Greater variation among connection bandwidths is expected after the introduction of the NBN.

2.2 Specification of Outcomes Valued by Stakeholders

Our team initiated an evaluation of broadband outcomes with consultation among interested parties and stakeholders at the national and regional levels and by a literature review to determine what is currently known and to identify important gaps in knowledge. These investigations highlighted the capability for individuals and communities to make use of NBN-enabled services and applications and to transfer skills from one Internet-usage domain to another. We will refer to this cluster of capabilities as *digital literacy*.

In order to compare changes in the adoption and diffusion of Internet usage among Australian residents before and after the introduction of the NBN, it was considered crucial that standardised measures examining a range of Internet behaviours be incorporated across all survey applications. Critical areas identified were Internet usage for information seeking, buying and using services, communication and social networking, and entertainment and leisure.

In addition to measuring behaviour change for the above core set of Internet domains across all regions, interested parties have highlighted the need to incorporate customised survey items to focus on Internet behaviour in areas of interest to a particular community. For example, our research partners at RISIR in Coffs Harbour have identified three important components to digital literacy in their region. These include the use of the Internet for (1) financial services, (2) local government services and (3) social/business networking. Through the utilisation of a behavioural model (*see* Sect. 2.3), this tailored approach can identify not only the changes in Internet behaviour for specific areas of regional interest but also the underlying determinants of that change.

Consultation has revealed also the importance of identifying the impact of digital literacy (i.e. the adoption and diffusion of Internet usage) on broader life outcomes including health, education, community engagement, standard of living and general life satisfaction. We expect the concept of digital literacy to have general relevance and applicability to these life domains, even if its underlying components vary by community.

2.3 Understanding the Behavioural Link Between Intervention and Outcome

We propose adopting a model for human behaviour to facilitate an understanding of the impact of the NBN on outcomes of interest and their determinants.

Over the last two decades, a rich body of research has emerged on consumer adoption and diffusion of technology innovations, resulting in the publication of a number of reviews and meta-analyses (e.g. [4, 12, 15, 22, 24, 25]). ICT adoption researchers have advocated the use of "Integrated Models", which generally use the Theory of Planned Behaviour's [1] core constructs as a foundation (e.g. [6, 11, 24]).

The Theory of Planned Behaviour (TPB) emphasises that people's behaviour are driven by their attitudes, social influences (i.e. norms) and their sense of control in relation to the behaviour. The predictive validity of TPB is well established [3]. The model has been employed to investigate human behaviour in a wide range of domains, e.g. water conservation, smoking cessation, blood donation, adoption of cleaner vehicles, household recycling and Internet and communication technology adoption. Most of the research applying the TPB to draw a link between the Internet and related behaviour has been conducted in the workplace and has been cross-sectional in nature. Little, therefore, is known about links between provision of the Internet and related behaviours and outcomes in residential settings [7, 12]. There is also a dearth of research studying these effects over time and the effects of interventions [25].

Figure 1 depicts a model for human behaviour that integrates the TPB with other theoretical constructs found to be effective predictors of the frequency and diversity of Internet usage, such as "perceived usefulness" and "perceived ease of use" from the Technology Acceptance Model [9] and "relative advantage" from Innovation Diffusion Theory [20]. In keeping with recent integrated behavioural models (e.g. [13, 18]), it takes into account both cognitive and emotional determinants of attitudes. This theoretical framework is used to identify which factors (attitudes, norms or perceived control in the second column from the left in Fig. 1) drive or inhibit behavioural intentions, along with the critical beliefs that underpin these factors (first column from the left in Fig. 1). Once identified, the relevant beliefs could, for

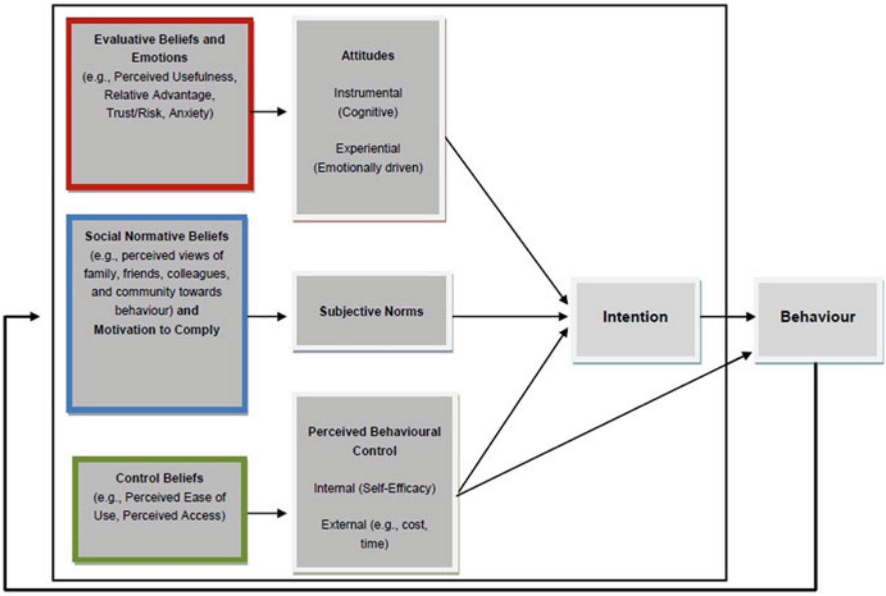


Fig. 1 An integrated model for the determinants of human behaviour

example, be measured over time to examine their relationship with behaviour change across subgroups of individuals and/or locations. Moreover, this approach facilitates the targeting of core beliefs in intervention strategies (e.g. education, training, marketing) to affect change.

2.4 *Qualitative and Quantitative Empirical Research Studies to Answer Specific Research Objectives*

To draw the link between the NBN and NBN-enabled applications and broader life outcomes, a critical step is to collect high-quality, reliable and valid data on individuals, households and communities. In new fields of enquiry, research is often needed to know exactly what to study, to develop survey instruments and to validate them. Following this preliminary work, substantive studies must be designed to answer the major research objectives and must be feasible given the available resources and time frame.

As previously noted, an understanding of the impacts of broadband requires longitudinal and experimental studies in household and/or community settings [6, 12]. We plan to focus on population-based research by recruiting representative samples from two regional Australian cities: Coffs Harbour and Port Macquarie. Roll-out of the NBN has begun in Coffs Harbour, although connections will not be possible before 2013. The roll-out has not begun in Port Macquarie and it is not among the early release areas.

3 Measurement of Broadband Connectivity, Digital Literacy (Internet-Related Behaviour) and Life Outcomes

Empirical conclusions depend heavily on the measurement of the intervention and the outcomes described in section two. Therefore, these aspects are described in more detail here.

3.1 Measuring Broadband Connectivity

It is difficult to directly observe individual broadband connectivity because (1) there is no standard definition of "broadband Internet" in terms of bandwidth transfer rate (Mb/s), (2) individuals may not know their transfer rate, and (3) for privacy reasons bandwidth transfer rate will be difficult to obtain and/or verify from ISPs. We therefore plan to investigate two approaches to measuring broadband connectivity. The first is to use a web-based broadband speed tester to directly measure download and upload speeds.

The second approach is to consider high-bandwidth connectivity as an unobserved category, e.g. an individual is (likely to be) connected to the NBN or is (likely) not connected. A set of simple, but relevant questionnaire items can be used as data from which to estimate the probability of belonging to each connection category. Questionnaire items need to relate to aspects of Internet use that are easily understood, e.g. Do you use the Internet at home? Do you have a wireless connection at home? Do you download films from the Internet at home? Do you play online games at home? A statistical model is used to estimate connection category probabilities [17].

If the second approach is validated against the first, it could prove useful in settings where it is not possible to directly measure the bandwidth available to an individual, e.g. owing to a participant opting for the paper version of a survey, when a network connection is not available, or if the web-based speed tester is down for technical reasons.

3.1.1 Validation Study for Measuring Broadband Connectivity

This study will be designed to validate our direct and indirect approaches to the measurement of broadband connectivity. For the direct approach, an ad hoc broadband speed tester will be engineered in our laboratories and tested for accuracy on hardware and software platforms that are likely to be encountered by users in the general population. Both the direct and indirect survey approach will then be trialled on a sample of residents selected from an NBN release area that has already been switched on to HBI.

In this study it will be important to include a large proportion of individuals who have chosen (or who are highly likely to have chosen) to make use of HBI, and to contrast them with a comparable sample from the same area who are not highly likely to have chosen to make use of HBI. Given that initial uptake of NBN services is low (http://nbnco.com.au), representative sampling of the population would lead to selection of very few HBI users and would be an ineffective strategy. As an alternative, NBN users could be randomly selected from users subscribing to an NBN-enabled application available in an early NBN release area. A control group could then be randomly selected from the same area. This type of study is sometimes referred to as outcome-dependent sampling and has efficiency advantages (in cost and precision) over random sampling study designs when the outcome is rare [5].

For the broadband speed tester, statistical analysis will focus on the willingness and ability of users to correctly access HBI and run the selected application. For the indirect survey approach, analysis of data would focus on (1) the interpretation of the indirect observed connection categories and (2) the accuracy of the survey approach in identifying connection categories by comparison with the “gold standard” broadband speed tester.

3.2 Measuring Digital Literacy and Its Determinants

Digital literacy is by definition a latent phenomenon which is not directly observable. Our proposal is to observe this phenomenon indirectly, through a set of questionnaire items concerned with Internet-related behaviour (e.g. frequency and diversity of Internet usage in various domains). As discussed previously, we propose to use one set of measures that will be kept constant for different regions in Australia, allowing for regional comparisons in terms of the impact of the NBN. A second set of measures will be used to focus on identifying change in Internet-usage behaviour for specific areas of interest to a given community. For this customised approach we propose to employ the integrated behavioural model using the core constructs of the TPB and, thereby, examine the underlying determinants of the Internet-related behaviour (e.g. evaluative beliefs). These will be conceived similarly as unobserved phenomena that are to be indirectly estimated. Structural equation models will be used for estimating latent constructs [23].

3.2.1 Telephone Interviews and Pilot Study

Before constructing a TPB questionnaire, it is imperative to first conduct a qualitative elicitation study to indentify people’s salient behavioural, normative and control beliefs (see left column Fig. 1) regarding engaging in the behaviour under consideration [2]. This information is then used to effectively word the items to be used in a quantitative survey.

Transcripts of interviews need to be analysed by at least two researchers using qualitative data management software (e.g. NVivo) to facilitate flexible and dynamic coding of text data. The goal is to explore the data for themes emerging from the text and to code, sort and query the data. An audit trail for each theme coded will be created with continual access to the original data for cross validation of themes.

Once the questionnaire items are devised, a quantitative pilot study should be conducted to test if the items are easily understood and relevant to the target population. This requires responses from a small but diverse community sample.

3.2.2 Validation Study

Major findings of quantitative studies critically depend on well-validated measurements. Therefore, and following any survey item modifications based on the outcome of the pilot study, the next objective is to conduct a validation study to assess the measurement properties of the survey instrument. A factor analysis of the data will be done to assess the reliability and validity of the items and hypothesised latent constructs. These constructs relate to the integrated behavioural model (utilising the TPB core constructs) as depicted in Fig. 1.

Participants will be selected to be representative of the general population (*see* Sect. 3.1.1). Sample size requirements will be based on estimating latent digital literacy constructs with sufficiently good precision. Sample size for factor analysis depends on the number of survey items, the number of factors and the communalities. Allowing for nonresponse of 10–15 %, then to detect true population factor structure to a high degree, the estimation of four factors based on six survey items per factor would require a sample size of 220 subjects assuming communalities ranged between 0.2 and 0.4 [19].

3.2.3 Cross-Sectional Benchmark Survey

Once the survey measurements have been validated as described above, the next goal is to conduct the survey proper. This survey would be administered prior to the introduction of the NBN in various regions and, therefore, serve as a benchmark of Internet-related behaviour to compare with follow-up studies. Ideally, a survey of this nature should be conducted with a representative sample of the population. In theory, representativeness can be ensured only by using some form of probability sampling, e.g. random selection. We plan to achieve representativeness by sampling participants as follows. First, residential addresses will be randomly selected using a geographic information system (GIS) and a Geocoded National Address File (G-NAF). Second, individuals will be randomly selected from households (e.g. by asking that the person usually resident in each household having the next birthday participate in the study).

To achieve a high response rate, we plan to use several strategies, including a media campaign to raise awareness of the study, provision of options for survey participants (e.g. online or hardcopy, delivered directly to households), incentives for participation and training of data collectors to better inform perspective participants. Sample size requirements will be planned using the data collected for the earlier validation study and based on estimating association between digital literacy and broader life outcomes (see below) with sufficiently good precision.

3.2.4 Measuring Broader Life Outcomes

It is also important to identify the potential impact of Internet-related behaviour on a range of life outcomes. The benchmark survey described above will, therefore, include various established outcome measures such as those relating to emotional well-being [26], community engagement and respectful treatment [8], environmental mastery—i.e. perceived control over one's life [21], skills and opportunities—and general satisfaction with life [10].

3.3 Randomised Control Trial

Employing the integrated behavioural model in conjunction with life outcomes will also allow us to examine the effectiveness of different educational strategies aimed at improving digital literacy within various communities. The aim would be to compare general NBN training with education initiatives tailored to the needs of the local community. In such a trial, conducted following the roll-out of the NBN, participants would be assigned to groups that receive one of the three different types of treatment. The government has produced a number of generic materials to help users make the most of the high-speed network. That material relies on self-directed learning. Access to that material, which includes implicitly any other materials users may find, forms treatment 1. Treatment 2 will be comprised of a set of generic training modules delivered to users individually or as small groups. The content of the generic modules will be pitched to illustrate both the capacity of the NBN and strategies for using the NBN across a range of activities (information seeking, buying and using services, communication and social networking, entertainment and leisure). Treatment 3 will provide for small user groups more tailored content: The units in this treatment will take into account the needs of the user group, the community and environment and will utilise wherever possible peer-to-peer trainers. Content for those groups will be tailored, in the first instance, to older users with no previous experience of the Internet; to those living in social housing with special needs or those receiving social benefits (long-term unemployed and so on); to those with some experience of the Internet but who would not normally use the Internet for all service, health, business and banking; and to high-end users (four groups). Those four groups dictate the participants to be allocated to the other treatments.

To that end, participants meeting the trial inclusion criteria will be allocated at random and from across the Coffs Harbour fibre service area maps to an initial treatment group. The impact of the training protocol will be measured as described above and follow-up assessments will be carried out at 3 months and then at 12 months after treatment.

4 Future Extensions and Discussion

We have described here some principles of impact evaluation and how these will be applied in our planned investigations of NBN impacts on the lives of people in several Australian regions. Our broader objectives include the promotion of NBN benefits and the identification (and reduction) of negative impacts. An example of the latter is a possible widening of the social exclusion gap and greater social inequality in some communities following provision of the NBN. Some of the foreseeable benefits of the NBN have been described in detail by Hayes [14]. We hope that by focusing attention on the behavioural links between high-bandwidth Internet provision and individual outcomes in the general community, we will be able to identify ways of promoting social inclusion through the use of the NBN. We intend to achieve these ends by conducting high-quality population-based research with the participation of regional institutes and input from local experts.

Our evaluation framework can be adapted to suit various regions in Australia and address different areas of interest in terms of Internet-related behaviour. We intend to pursue this extension of the present work in collaboration with research partners in universities and other institutions, so as to expand expertise and application. We hope also to extend and generalise our evaluation framework to meet the needs of policymakers at the national level, as proposed in a recent paper based on experience with the introduction of HBI in the United Kingdom [27].

References

1. Ajzen I (1991) The theory of planned behaviour. Organ Behav Hum Dec 50:179–211
2. Ajzen I (2006) Constructing a TPB questionnaire: conceptual and methodological considerations. http://people.umass.edu/aizen/pdf/tpb.measurement.pdf. Accessed 12 Jul 2012
3. Armitage CJ, Conner M (2001) Efficacy of the theory of planned behaviour: a meta-analytic review. Br J Soc Psychol 40:471–499
4. Bagozzi RP (2007) The legacy of the technology acceptance model and a proposal for a paradigm shift. J Assoc Inf Syst 8:244–254
5. Chen HY (2003) A note on the prospective analysis of outcome-dependent samples. J Roy Statist Soc Ser B 65:575–584
6. Choudrie J, Dwivedi YK (2004) Towards a conceptual model of broadband diffusion. J Comput Inform Tech 12:323–338
7. Choudrie J, Dwivedi YK (2006) Investigating factors influencing adoption of broadband in the household. J Comput Inform Syst 46:25–34

8. Dane SD, Stenner K, Hobman EV (2012) Predicting subjective well-being: material versus psycho-social needs. Report EP 124179. Commonwealth Scientific and Industrial Research Organisation (CSIRO), Australia
9. Davis FD, Bagozzi RP, Warshaw PR (1989) User acceptance of computer technology: a comparison of two theoretical models. Manage Sci 35:982–1002
10. Diener E, Emmons R, Larsen J, Griffin S (1985) The satisfaction with life scale. J Pers Assess 49:71–75
11. Dwivedi YK, Choudrie J, Brinkman WP (2006) Developing a survey instrument to examine consumer adoption of broadband. Ind Manag Data Syst 106:700–718
12. Dwivedi YK, Williams MD, Lal B, Mustafee N (2010) An analysis of literature on consumer adoption and diffusion of information system/information technology/information and communication technology. Int J Electron Gov Res 6:58–73
13. Fishbein M, Cappella JN (2006) The role of theory in developing effective health communications. J Commun. doi:10.1111/j.1460-2466.2006.00280
14. Hayes R (2011) Valuing broadband benefits. Institute for a broadband-enabled society. University of Melbourne, Melbourne
15. Jeyaraj A, Rottman JW, Lacity MC (2006) A review of the predictors, linkages, and biases in IT innovation adoption research. J Inf Technol. doi:10.1057/palgrave.jit.2000056
16. Leeuw F, Vaessen J (2009) Impact evaluations and development: NONIE guidance on impact evaluation. World Bank, Washington, DC
17. McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York
18. Montano DE, Kasprzyk D (2008) Theory of reasoned action, theory of planned behaviour, and the integrated behavioural model. In: Glanz K, Rimmer BK, Viswanath K (eds) Health behavior and health education. Theory, research, and practice, 4th edn. Jossey-Bass, San Franciso, CA
19. Mundfrom DJ, Shaw DG, Lu Ke T (2005) Minimum sample size recommendations for conducting factor analyses. Int J Test 5:159–168
20. Rogers EM (1995) Diffusions of innovations. Free Press, New York
21. Ryff CD (1989) Happiness is everything, or is it? Explorations on the meaning of psychological well-being. J Pers Soc Psychol 57:1069–1081
22. Silva L (2007) Post-positivist review of technology acceptance model. J Assoc Inf Syst 8:255–266
23. Skrondal A, Rabe-Hesketh S (2004) Generalized latent variable modeling. Chapman and Hall, Boca Raton, FL
24. Venkatesh V, Morris MG, Davis GB, Davis FD (2003) User acceptance of information technology: toward a unified approach. Mis Quart 27:425–477
25. Venkatesh V, Davis FD, Morris MG (2007) Dead or alive? The development, trajectory and future of technology adoption research. J Assoc Inf Syst 8:267–286
26. Watson D, Clark LA, Tellegen A (1988) Development and validation of brief measures of positive and negative affect: the PANAS scales. J Pers Soc Psychol 54:1063–1070
27. Williams T (2011) Connecting communities: the impact of broadband on communities in the UK and its implications for Australia. Huawei Technologies Co. Ltd, Sydney

A Social-Creative-Cognitive (SCC) Model for Requirements Engineering

Linda Dawson

Abstract This paper describes a research project and the findings from a set of six cases which examine professional requirements engineering (RE) practice from the perspective of how analysts reach agreement on system requirements for information systems development (ISD). In these studies, it was found that the analysts reached agreement on requirements with clients through communication and negotiation based on both analysts' cognitive skills in problem-solving and creative skills in developing informal models that could be used in the negotiation of agreement and sign-off on requirements specifications. A theoretical model of social-creative-cognitive (SCC) aspects of requirements engineering is proposed encompassing the empirical findings.

1 Introduction

The social and collaborative aspects of the requirements engineering process have been well documented [7, 17, 18, 24, 26, 27, 36, 38, 41]. Viewing the requirements engineering process as a social process implies that if the product of the requirements engineering process is the specification document on which the design and implementation of the system is based, then this product has to be agreed upon by both parties. That is, the specification needs to be validated as correct or acceptable from both points of view—the formally modelled consultant's point of view and the client's informally modelled point of view.

This paper reports on part of a larger longitudinal project [9, 11] whose objective is to understand how systems are developed in practice and which roles, if any, experience, cognitive processes, social processes, methods, models, tools and

L. Dawson (✉)
Faculty of Informatics, University of Wollongong, Wollongong, NSW, Australia
e-mail: lindad@uow.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_9,

techniques play in the RE process. The research reported in this paper addresses part of the larger project on the basis of the following research question:

> How do professional developers reach agreement on requirements for system development?

Sections below present the findings of a multiple-case study addressing these questions based on qualitative data collection and analysis methods. Section 2 provides background in related work. The research approach adopted is presented in Sect. 3 along with descriptions of the six cases. The findings are presented in Sect. 4, and Sect. 5 presents a theoretical model which encompasses the findings and discusses the implications of the findings for practice and education and training.

2 Social Processes in Requirements Engineering

Although there is no single commonly accepted definition of requirements engineering [12, 24, 26, 38], the definition used in this paper is

> Requirements engineering is an iterative and collaborative process of elicitation, modelling, and validation of information system requirements which provides an agreed specification which is the basis for the design and implementation of that information system.

This definition places the requirements engineering process early in the system development cycle, although in many projects it may not necessarily be the very first phase. It may be that the requirements engineering process is triggered by preliminary investigations [1] or questionings of users and clients of the way activities are undertaken within their organisation [5]. This definition also contains explicit reference to the collaboration needed between clients and analysts [41] and the need for feedback via iteration in the process of specifying requirements.

There are various frameworks proposed for understanding RE practice [20, 24, 26, 31, 32].

Pohl [32] proposes a framework suggesting three dimensions of the requirements engineering process. These three dimensions can be outlined as follows:

- Specification dimension involves the development of the specification from the "opaque" to the specific.
- Representation dimension which deals with the methods for representing the specification and includes informal, semi-formal and formal languages.
- Agreement dimension describes the "common specification" or agreed specification which is based on the different viewpoints of the parties involved in developing the specification.

This view of requirements engineering is in harmony with the idea of information systems development being more than a purely technical undertaking. It incorporates the concept of an information system described by [24] as a "sociotechnical system," that is, a system "… that involve[s] computer-based components interacting with people and other technical system components in an organisational setting".

The agreement dimension of Pohl's framework is specifically explored in this research paper.

3 Research Approach

The objective of the project described in this paper was to understand how systems requirements are developed and agreed on in practice and which social or other aspects play a role in this agreement. The research questions are:

- How do professional developers and clients reach agreement on requirements for system development?
- Is this agreement a purely social agreement between the developer and the client?
- What other aspects contribute to this agreement?

The evolutionary case research approach [9] was used in this study based on recorded semi-structured interviews with individual requirements engineers. These interviews provided empirical data which is interpretive and descriptive rather than normative or quantitative. Interview questions focussed on exploring the three main processes of requirements engineering: elicitation, modelling and validation, e.g. Is elicitation explicitly undertaken and when does it start? When does it end? When does modelling begin? Do you think it is necessary to validate the specification once the models have been produced? Which (how many) models are produced during specification? Who are they produced for? Who uses them? Which models, if any, are shown to the user? Which models are used internally by the development team?

The evolutionary case approach is iterative and cyclic and is particularly suited to interview-based data collection. In each case cycle the researcher seeks to refine the current version of the theoretical model by:

- Looking for *reinforcement* of concepts already contained within the theoretical model or framework
- *Revelation*—identifying new areas for exploration and potential reinforcement
- Learning and *reflection* on data collected so far
- *Re-examining* previous transcripts to find any further reinforcement of an emerging theme

The researcher is active in the data collection. Leading questions are encouraged in order to facilitate reinforcement, and semi-structured, open-ended questions are used to facilitate revelation. Exploration of these revelations is incorporated into revised interview scripts which are used in the next case in the cycle. Reinforced concepts are retained in the evolving theoretical model. The process is ongoing but can be concluded when there has been enough reinforcement for a representative model of the research domain being investigated to stand alone or when theoretical saturation has been reached [14]. So, the outcome of the research method is a theory about the area being investigated which is initially grounded in the literature and then progressively grounded in data gained from investigating the application of system development methods in practice.

Table 1 Background information for each consultant

Case	Job title	Client	Project
1	Operations manager	Federal govt	Complex technical
2	Principal consultant	State govt	Web based transactions
3	Senior consultant	Telecommunications	Fault management system
4	Director and partner	Software developer	Insurance
5	Technical manager	Software developer	Stockbroking package
6	Principal consultant	Hospital	E-prescribing

4 The Case Studies

Although the larger longitudinal project currently comprises eight cases, the six cases reported in this study provided data specific to the agreement dimension of requirements engineering. Participants were recruited through industry. Some participants provided contacts for subsequent participants. There was no attempt to select participants based on specific background characteristics. Most participants used object-oriented approaches to ISD, one used an agile feature-driven approach and all participants were familiar with a range of methodologies. The contextual information for each consultant interviewed is summarised in Table 1.

5 Findings

Data analysis was cyclic and based on identifying (revealing) and confirming (reinforcing) themes from the interview transcripts using an illustrated narrative style as described by Miles and Huberman [30] and as used in Fitzgerald [15] and Urquhart [41]. Miles and Huberman [30] describe this as looking for "… key words, themes, and sequences to find the most characteristic accounts". Themes that are revealed and reinforced become part of the evolving theoretical model [9].

Case 1 involved a small confidential project which had to be completed quickly for a government department. The system was highly technical and involved complex calculations and predictions. The consulting organisation in this case used a commercial semi-object-oriented method (James [33]; James [34]) based on a template and the use of cards to describe requirements. The template is a booklet that provides guidelines for the tasks which need to be undertaken during the process of requirements specification. Every requirement that is documented is based on a requirements card describing various "characteristics". The cards are filled out in collaboration with the client/users during the requirements specification process. One characteristic associated with agreement is the "fit criteria" which is a user-defined test which ensures a requirement is a single functional unit which can be tested. Also associated with agreement is the customer satisfaction characteristic based on how *happy* the customer would be if the requirement was included and the customer dissatisfaction characteristic based on how *unhappy* the customer would be if the requirement was not included. This allows the consultant and client to prioritise any "wish list" the client might have.

> We write the cards with the client, then we go away and write the document. The cards are essentially self-documenting, but then we do a second level of checking [to ensure] that we haven't misinterpreted [anything] by actually stepping through it again. The only testing is the identification of the fit criteria.

So, in Case 1 agreement was reached based on the cards developed with the client. The cards represented an informal model of the system that was easily discussed by both parties.

Case 2 used object-oriented methods to develop in-house specification templates representing identified "common transactions" for multiple clients who could then do their own requirements engineering with the assistance of an IT liaison person. Booklets were sent to the client organisations containing instructions on how to use the template, the generic use case (diagram and script), the generic object model, interaction diagram, etc. The first task for the client was to work through the general use case flow diagram to see how well their transaction matched the common model. This test was called a "goodness of fit" test. Customising the template involved modifying the basic flow diagram (based on the "goodness of fit") and the object model, modifying the use case script by striking out (not removing) elements, so that someone could look across the page and see what had been changed.

So, in Case 2 agreement was based on the client customising a booklet-based template with assistance from an IT person. Only simplified use case diagrams and dialogues were shown to the users when describing requirements. Models based on formal notation were considered too complex for users to understand: "We tell them [the users] that the model is technical mumbo jumbo … you know I wouldn't show them a data model either … the closest I've gotten is working with this type of flow diagram (use case flow diagram)… they can follow that pretty well but they don't usually have the patience to really work through the interaction diagrams or the model."

The consultant in Case 3 used object-oriented methods to specify and build a fault management system for a telecommunications organisation. In this case agreement was reached using informal models such as use case scripts and a prototype to develop the requirements in consultation with "subject matter experts" nominated by the client. One of the main members of the expert group (the main business contact) was a network manager with about 25 years experience in transmission management. He knew all there was to know about the client's management of their transmission network. He was involved wherever possible and he played a user liaison role and a business expert role. "A lot of the requirements model was drawn by talking to these guys and verified as well through the development phase." Knowledge elicitation was done using interviews with users and the special expert user group. It was highly iterative to the degree where the subject matter expert would be calling in every couple of days. "It would have just been a conventional sort of thing, throw some prototype together … and that can be done very quickly. Get the guy in, sit down and work through our current prototype and that might have happened once a week for twenty weeks." Working on this project was one of the subject matter expert's main job responsibilities. He was freed up from some of his network management responsibilities to come and work with the system team. There was a separate acceptance test suite developed by the users, but that was not set up until well into the development stage approximately six months before acceptance testing was due to start. "All the way through we used our use case model to test things as we were developing … ".

The consultant in Case 4 had extensive experience in using object-oriented methods to specify and build actuarial and insurance systems. Agreement was reached using models shown to users based on ad hoc diagrams, rich pictures and screen simulations rather than class models or interaction diagrams, although the diagrams using formal notations were used within the team and in the design phase.

In Case 4, the analyst was explicit about using various ad hoc diagrams, pictures, PowerPoint simulations and use cases to reach agreement with the clients/users. "I mean if you draw a picture and that doesn't make any sense to them then you draw another one … [a] requirements specification has to be in terms that they understand." And " … and in every project I've ever worked on … there's been a few key pictures. The one I'm working on at the moment is the billing cycle—it's a wheel and its got the steps in the billing cycle on it and that's in everybody's head and everybody talks in those terms and it's just the key base thing—it's the conceptual core of the thing … I'm a great believer in ad hoc diagrams that give the picture that springs from your understanding of the problem and in a lot of OO work the process of development hinges on one or two of these pictures."

The consultant in Case 5 was a senior project manager for a software development organisation which creates custom-built systems for individual clients including generic packaged software systems for the stockbroking industry. The consultant was experienced in many methods, both object-oriented and non-object-oriented, for specifying and building business systems. The methodology was an in-house methodology based on UML notation but not the complete rational development method. Prototyping in the form of a GUI prototype for the users was used in the project. "We actually do a prototype and then work through the users with that and then gain sign off at that level."

So, in Case 5, agreement was reached using prototypes, screen simulations and animations with use case models used mainly at the validation phase. In this case the prototype was the most used tool for validation and agreement, and use cases were only used for exceptions or special cases "… as we were looking at the requirements document we had the prototype running and projected up on a big screen and we walked through the prototype in relation to the requirements".

The consultant in Case 6 was a principal consultant in a software development organisation which develops custom-built systems for large healthcare projects. This project was an implementation of the full ICT infrastructure for a new private hospital and also used the hospital as a test bed for the broader healthcare group, to trial new technologies and to roll out systems to the rest of the group. The consultant was experienced in many methods but was using an agile feature-driven approach (AFDD). In this case agreement was reached using text, rich pictures and modified models " … process modelling with data flows in it, but done in a way that is in a way that the business will understand it, using symbols that that business understands … rich text, rich pictures". This consultant also described another method used in a previous project, " … very much the user centric approach, we used the 'big floppy book', which was actually a whole heap of explicit, mock screen shots… [p]ut them together as a book and say this is how… your business process works, and they could actually flick through the screens".

In all the cases reported here, there were two types of models produced: informal models in the form of pictures, text and diagrams, and/or use cases, and prototypes, i.e. models that can be understood and explained without specific training, which were separate from formal models based on specific modelling notations such as entity-relationship modelling and UML modelling. The two types of models were used for different purposes. The informal models were used in the validation of the specification and achieving understanding and agreement with clients, and the formal models were used internally within the analysis team and passed on to the design phase of the development.

6 Discussion and Implications

The basis for the perceived need for both informal and formal models in requirements engineering as found in this study confirms that the requirements engineering process is fundamentally a social process involving two main groups: the users/clients and the professional consultants [7, 16, 24, 32, 41]. It is not claimed that the implications discussed here are new or exhaustive, rather that the findings from this research project strengthen the idea that requirements engineering is a social, creative and cognitive process [8, 17, 19, 38].

Based on these findings and previous research in the literature as discussed above, we propose a social-creative-cognitive model to encompass the relationships between these concepts. The model is constructed incrementally as follows.

Requirements engineering is a *social process*, and this social process requires understanding by all parties to reach agreement. Understanding requires communication skills, and agreement requires negotiation skills. The facilitation of understanding and agreement also requires *creative modelling skills* on the part on the analyst to produce understandable informal models. These models are developed during elicitation, refined during modelling and used for validation of requirements before sign-off or agreement to go ahead with design, and implementation is given by the client.

Further, creative informal modelling as demonstrated by the analysts in this study relied on *cognitive skills* including abstraction and mental modelling and problem-solving and reasoning skills particularly analogical reasoning skills on the part of the analyst as reported elsewhere in Dawson [10].

These concepts and their relationships are represented in the theoretical model shown in Fig. 1. This model contains features for defining a theoretical model as

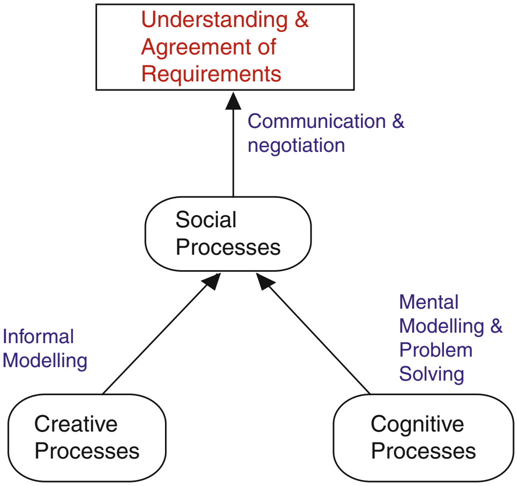


Fig. 1 A theoretical model of the contribution of social, creative and cognitive processes to requirements engineering

defined by Dubin [13] and Bacharach [2], i.e. the interactions or relations between defined units or concepts within a set of boundaries or constraints depicting a limited portion of the world.

In this model, the main social goal of successful requirements engineering is to achieve agreement and understanding about requirements between users/clients and the professional developer or development team. The achievement of this goal depends on three processes. The social process involves the users and the analysts in communication and negotiation which brings about the understanding and agreement. This social interaction is influenced by the professional input of the analyst in the role of problem-solver and modeller. Further, the analyst also has to express the solutions to the problems and the models arrived at in his/her mind in a concrete manner which facilitates the understanding and agreement. This creative process involves the development of informal models (such as diagrams, simulations, animations or textual explanations) that can be understood and discussed by the users and analysts in their social communications and negotiations. These implications are discussed further in the following two sections.

6.1 Requirements Engineering as a Social Process

The perception of the analysts in this study was that for agreement between analyst and client to take place, there needs to be two types of models: informal models for communicating the specification to the user for information and validation and formal models developed by the analyst team to pass on to the design and implementation team.

It has been generally recognised [6, 25, 42] that many of the errors that lead to costly maintenance and/or failure of information systems can be traced to omissions, inconsistencies and ambiguities in the initial requirements specification. If, as the findings of this research project confirm, the models used for validation of the specification with the clients are different to the models used in design and implementation, then this may indicate one of the areas where these inconsistencies, omissions and ambiguities might arise. Recognising and understanding this issue requires further research and provides a step towards building the right tools and techniques to assist the requirements engineering process.

6.2 Requirements Engineering as a Creative and Cognitive Process

As with many professional analysis and design activities involving creative reflection [22, 28, 35], requirements engineering can be considered to be a creative process particularly on the part of the analyst producing the requirements specification. The case studies showed evidence of recognition on the part of the practising professionals that they had to be able to model or represent what the users wanted in some diagrammatic form. The findings suggest all of the analysts who used formal notations such as UML or entity-relationship diagrams would not use diagrams based on these notations with the users or clients because they believed the users would not understand them. For all of the analysts reported in this study, this meant that they had to find (considerably diverse) creative solutions to the representation problem: simplified use cases, ad hoc diagrams, rich pictures, animations, PowerPoint simulations and text-based explanations. Each analyst had his/her own creative approach to informal user modelling. There is enough evidence provided in the case studies to imply that this creative approach to user modelling is common in professional requirements engineering practice. This suggests that the variety and use of such informal models should be systematically described in detail, which would be useful to professionals, educators and students.

Closely related to the creative aspects of requirements engineering are the cognitive aspects of requirements engineering. Requirements specification can be considered as a high-level cognitive process [10, 29, 37, 39]. As previously reported [10], in four of the six cases in this study, requirements specification involved mental modelling during the transformation from elicitation to concrete models for design and implementation. Overall, these four analysts believed that they were continually "modelling in the mind" during the elicitation process and that these mental models were further refined in the mind before they were communicated to others (users or fellow analysis team members) or before they were committed to paper. So, there is evidence that requirements modelling requires cognitive skills including abstraction and mental modelling together with problem-solving and reasoning skills particularly analogical reasoning skills on the part of the analyst. These cognitive skills are an essential foundation for the social interactions required between analyst and client for understanding and agreement to be reached.

6.3 Implications for Education and Training

Typical undergraduate courses in information systems or related disciplines involve some exposure of students to systems analysis methodologies, techniques and tools. Often, students are required to participate in a project where the principles of systems analysis can be applied to an example of an industrial or organisational style system development project. The challenge for academics designing these courses or writing textbooks to accompany these courses is often how to relate theory and project work to real professional practice. Successful programmes in these areas have provided projects with real clients, project team environments and/or other real world environments [3, 4, 21, 23, 40].

The findings from the case studies in this research project have several implications for education and training. Based on the findings presented here, courses seeking to provide realistic commercial and organisational project environments should include the following elements and ideas.

There are many tools and techniques for requirements analysis and specification, and as this research project has shown:

- In practice analysts often develop their own in-house methodologies based on diverse tools and techniques rather than adhere to a single prescribed or commercial methodology.
- Many professional analysts build their own personal methodology by trying out and adapting those techniques and tools that suit their way of thinking and their way of interacting with clients and the particular projects that they are working on.

There are many models for representing requirements, and as this research project has shown:

- Users may not understand formal notations like ER, DFD and UML diagrams.
- Some professional analysts develop informal models based on ad hoc diagrams, rich pictures, animations, PowerPoint simulations, text-based explanations and simple use cases for explaining requirements to users/clients.
- Informal models which are not based on formal notations like ER and UML are often the basis for agreement and sign-off for requirements specifications.

The implication for education and training from these ideas (and suggested by the study findings) is that for students to be able to undertake a major project they need to be encouraged to build their own personal toolkit after being exposed to as many tools and techniques as possible. This also implies that students should be encouraged to experiment with and develop some informal modelling techniques for communicating requirements to users/clients. There are many different modelling methods and notations available and that some are more appropriate for certain types of projects than others.

7 Conclusion

This paper has identified and discussed the implications of a set of case studies of professional requirements engineering practice regarding how professional requirements engineers reach agreement with clients on requirements specifications. The analysis of the findings is the basis of a theoretical model of the processes, concepts and relationships involved in the requirements engineering process with respect to the development and use of informal models for reaching understanding and agreement between users and analysts. The case study findings also suggest a need to examine education and training methods for requirements engineers and systems analysts with respect to developing diverse approaches to the use of models, methodologies, techniques and tools for elicitation, modelling and validation of user requirements.

References

1. Avison D, Fitzgerald G (2003) Information systems development: methodologies, techniques and tools, 3rd edn. McGraw-Hill, Maidenhead
2. Bacharach S (1988) Organizational theories: some criteria for evaluation. Acad Manag Rev 14(4):496–515
3. Brown RBK, Piper IC (2011) What users do: SA&D with the ATSA method. Paper presented at the 20th international conference on information systems development (ISD2011), Edinburgh, UK
4. Carbone A, Lynch K, Arnott D, Jamieson P (2000) Adopting a studio-based education approach into information technology. Paper presented at the 4th Australasian computing education conference (ACE2000), Melbourne
5. Checkland P, Scholes J (1990) Soft systems methodology in practice. Wiley, Chichester
6. Conboy K (2010) Project failure en masse: a study of loose budgetary control in ISD projects. Eur J Inform Syst 19(3):273–287
7. Coughlan J, Macredie RD (2002) Effective communication in requirements elicitation: a comparison of methodologies. Requirements Eng 7(2):47–60
8. Darke P, Shanks G (1996) Stakeholder viewpoints in requirements definition: a framework for understanding viewpoint development approaches. Requirements Eng 1:88–105
9. Dawson L (2008) Active exploration of emerging themes in a study of object-oriented requirements engineering: the "Evolutionary Case" approach. Electron J Bus Res Meth 6(1):29–42
10. Dawson L (2011) Cognitive processes in object-oriented requirements engineering practice: analogical reasoning and mental modelling. Paper presented at the 20th international conference on information systems development (ISD2011), Edinburgh, UK
11. Dawson L, Darke P (2002) The adoption and adaptation of object-oriented methodologies in requirements engineering practice. Paper presented at the 10th European conference on information systems, Gdansk, Poland
12. Dawson L, Swatman P (1999) The use of object-oriented models in requirements engineering: a field study. Paper presented at the 20th international conference on information systems, Charlotte, NC
13. Dubin R (1976) Theory building in applied areas. In: Dunnette M (ed) Handbook of industrial and organisational psychology. Rand McNally College Publications, Chicago, IL, pp 17–39

14. Eisenhardt KM (1989) Building theories from case study research. Academy of Management Review 14(4):532–550
15. Fitzgerald B (1997) The use of systems development methodologies in practice: a field study. Inform Syst J 7:201–212
16. Flynn DJ, Warhurst R (1994) An empirical study of the validation process within requirements determination. Inform Syst J 4:185–212
17. Galal G, McDonnell JT (1998) A qualitative view of requirements engineering. Paper presented at the 3rd Australian conference on requirements engineering, Geelong, Australia
18. Goguen JA (1994) Requirements engineering as the reconciliation of social and technical issues. In: Jirotka M, Goguen JA (eds) Requirements engineering: social and technical issues. Academic, London
19. Jackson P, Klobas J (2008) Building knowledge in projects: a practical application of social constructivism to information systems development. Int J Proj Manag 26(4):329–337
20. Kavakli E (2002) Goal-oriented requirements engineering: a unifying framework. Requirements Eng 6(4):237–251
21. Keen CD, Lockwood C, Lamp J (1998) A client-focused, team-of-teams approach to software development projects. Paper presented at the software engineering: education and practice, Dunedin, New Zealand
22. Khushalani A, Smith R, Howard S (1994) What happens when designers don't play by the rules: towards a model of opportunistic behaviour and design. Aust J Inform Syst 1(2):2–31
23. Lamp J, Lockwood C (2000) Creating realistic experience of an IS project: the team of teams approach. Paper presented at the 2000 IRMA international conference, Alaska
24. Loucopoulos P, Karakostas V (1995) Systems requirements engineering. McGraw-Hill, London
25. Lyytinen K, Robey D (1999) Learning failure in information systems development. Inform Syst J 9(2):85–101
26. Macaulay L (1996) Requirements engineering. Springer, London
27. Machado R, Borges M, Gomes J (2008) Supporting the system requirements elicitation through collaborative observations. In: Briggs R, Antunes P, de Vreede G-J, Read A (eds) Groupware: design, implementation and use, vol 5411. Springer, Berlin, pp 364–379
28. Mathiassen L, Purao S (2002) Educating reflective systems developers. Inform Syst J 12(2): 81–102
29. Mayer RE (1992) Thinking, problem solving, cognition, 2nd edn. W.H. Freeman and Company, New York
30. Miles MB, Huberman AM (1994) Qualitative data analysis: an expanded sourcebook, 2nd edn. Sage Publications Inc., Thousand Oaks, CA
31. Nuseibeh B, Kramer J, Finkelstein A (1994) A framework for expressing the relationships between multiple views in requirements specification. IEEE Trans Software Eng 20:760–771
32. Pohl K (1994) The three dimensions of requirements engineering: a framework and its applications. Inform Syst 19(3):243–258
33. Robertson J, Robertson S (1997) Volere requirements specification template, 4th edn. Atlantic Systems Guild, London
34. Robertson J, Robertson S (2010) Volere product summary. http://www.volere.co.uk/index.htm. Accessed 6 Jul 2012
35. Schön DA (1983) The reflective practitioner: how professionals think in action. Basic Books, New York
36. Siau K, Rossi M (2011) Evaluation techniques for systems analysis and design modelling methods—a review and comparative analysis. Inform Syst J 21(3):249–268
37. Siau K, Wang Y (2007) Cognitive evaluation of information modeling methods. Inform Software Technol 49(5):455–474
38. Sommerville I, Sawyer P (1997) Requirements engineering: a good practice guide. Wiley, Chichester
39. Sutcliffe AG, Maiden NAM (1992) Analysing the novice analyst: cognitive models in software engineering. Int J Man Mach Stud 36:719–740

40. Teles VM, de Oliveira CET (2003) Reviewing the curriculum of software engineering undergraduate courses to incorporate communication and interpersonal skills teaching. Paper presented at the software engineering education and training, 2003 (CSEE&T 2003). Proceedings of 16th conference on 20–22 Mar 2003
41. Urquhart C (1998) Analysts and clients in conversation: cases in early requirements gathering. Paper presented at the 19th international conference on information systems, Helsinki, Finland
42. Wastell DG (1999) Learning dysfunctions in information systems development: overcoming the social defenses with transitional objects. MIS Quart 23(4):581–600

Beyond Interface Design: Considering Learner Cognition When Designing E-Learning Systems

Mark Freeman, Abdallah Alasraj, and Paul Chandler

Abstract By developing e-learning systems with an understanding of users' cognitive load, rather than just focusing on traditional usability constructs, it is envisaged that better learning outcomes will occur. This conceptual paper presents a review of how an understanding of cognitive load can assist with the processes of developing e-learning systems that allow for increased learning outcomes. Through a comparative analysis of human–computer interaction (HCI) methods and cognitive load theory (CLT), a greater understanding of design principles can be gained. The paper focuses on the three main effects discussed in CLT literature—split-attention, redundancy, and element interactivity—and how a developer could use these methods to reduce cognitive load and improve learning outcomes.

1 Introduction

Consideration of a user's cognition when designing any system has great significance, particularly when focusing on e-learning systems. In the past, e-learning systems have been termed as a "wicked problem" due to their complexity [9]. It has been argued that web-based systems have a significant role to play in the facilitation of learning because hypertext structures have a likeness to the human brain's mapping of knowledge; this is from the perspective of Cognitive Load Theory (CLT) [29, 30]. Prior research by the authors (see [1]) has identified various ways that the

M. Freeman (✉) • A. Alasraj
School of Information Systems and Technology, University of Wollongong, Wollongong, NSW, Australia
e-mail: mfreeman@uow.edu.au; aafaa858@uowmail.com.au

P. Chandler
Faculty of Education, University of Wollongong, Wollongong, NSW, Australia
e-mail: paul_chandler@uow.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_10,

interfaces of procedural learning systems could be developed to aid in increasing learning outcomes through the application of CLT principles. As a human learns, he acquires new links and structures within his brain, and this leads to an increased level of information built on prior structures; this process is known as *acquisition*. Restructuring entails a learner organizing knowledge structures into groups, procedures, and schemas. Human knowledge thus exists in semantic memory that works as a network of interrelated concepts. Research has shown that two kinds of memory exist within the human brain: working memory (WM) and long-term memory (LTM). WM processes information and then sends it to LTM where the information is "permanently" stored and potentially retrieved at a later time for subsequent use.

By grounding the development of e-learning systems within a CLT context, and through effective use of the human–computer interaction (HCI) processes, the human brain will acquire and process information that exceeds the traditional understanding of the capacity of WM. This is achieved through better restructuring of previous knowledge. This is supported by the findings in Alasraj et al. [1], where the interfaces were developed with CLT principles in mind. This conceptual paper will present a review of the literature on how the human memory processes information. This will be followed by a discussion of three methods that can be used for better knowledge acquisition through e-learning systems: split-attention, redundancy, and element interactivity. By an e-learning developer demonstrating awareness of these methods through e-learning system design, learners are able to processes greater amounts of information and achieve greater learning outcomes. It is argued that by applying attention to aspects beyond the usual usability issues of a system's interface (e.g., efficiency, effectiveness, ease of use, utility, learnability) and focusing on the usability of the content being delivered, greater learning outcomes can be achieved by a learner and good usability would naturally occur by understanding how the human brain works.

1.1 ICT in Education

With the increasing use of the Internet (particularly the World Wide Web—www) for communication and information access, there has been pressure to better use ICT within learning contexts. The ability for ICT to support educational programs that are customized for the purpose of meeting the needs of the students [13] and the flexible delivery opportunities that ICTs can provide have increased the demand for high-quality ICT integration into education. E-learning is seen as one method available to assist in the delivery of information to learners. The success of e-learning technologies as part of a learning process has previously been established in the literature [14]. E-learning has the ability to offer flexibility in education [2]. However, learners need to engage with these technologies during their education to gain the true benefits of this solution [19]. This is a major issue contributing to the current situation where the full potential of e-learning systems has not been realized and why, for some, it is considered a "wicked problem." Traditional usability evaluations of these systems have resulted in a focus on superficial interface design, rather than on the modes of learning afforded by this new medium through which the learners can engage.

Traditionally, learner instruction has focused on content. In contrast, when the information is presented through e-learning systems, the focus has been on interface design. Typically, courses have been planned and written around textbooks. Contemporary trends now prefer programs and courses that encourage performance and competency. These programs are beginning to emphasize aptitude and abilities and focus on how this information would be used rather than what the information is about. The shift to performance- and competency-based curricula has been greatly encouraged and supported with an increase in ICT tools [26]. In addition, these new forms of instruction have continued with the introduction of new technologies, particularly the Internet and multimedia.

2 Cognitive Load Theory

Sweller [29] defines human cognitive architecture as the way cognitive structures are organized and interrelated. Information is stored and processed by WM and LTM so it can be retrieved for subsequent use. Information enters the human information processing system via a variety of channels associated with the senses, commonly referred to as sensory memory; at this stage, the information can be forgotten or initially processed into WM, the momentary storage of information that is processed to perform cognitive tasks. When information is in WM, it can be forgotten or elaborated and coded into LTM. WM was originally referred to as short-term memory. CLT bases WM on the ideas of Miller [17], where it has the capacity of about seven plus or minus two "chunks" of information and a duration of between 18 and 20 s [24]. The size of a chunk of information is dependent on the topic and the learner. LTM is the "permanent" memory storage within human memory. Previously learned and stored information is processed in LTM and used with tasks such as problem solving, but the contents of LTM cannot be used until they reenter WM through a process of retrieval [29]. With the use of effective HCI, the human mind has the potential to process large amounts of information that exceed WM capacity, which would usually reduce the learner's ability to gain the necessary knowledge from the system so it could be reused in new situations [16]. In order to best develop resourceful methods and programs, e-learning systems need to be modified to reflect the cognitive attributes of the learner.

CLT deals with the limitations of WM and its interaction with LTM, as a "permanent" memory store. Chandler and Sweller [5] have defined the total amount of mental activity that WM deals with simultaneously as *cognitive load*. The basic process of understanding new knowledge involves the reconstruction of existing knowledge (or schemas) to generate new higher-order knowledge (schemas).

For a novice learner, the process of acquiring or solving a new or complex task would involve processing the elements as units into a number of low-order schemas, which are then combined to form a higher-order schema [30]. Once a schema has been constructed for a complex task, all the related interactions are incorporated in that schema and it is then treated as a single "chunk" by a learner's WM, thus reducing WM load. There are three types of cognitive loads that interact with a learner's ability to learn new information: intrinsic, extraneous, and germane load.

2.1 Intrinsic Cognitive Load

CLT states that intrinsic cognitive load (ICL) is determined by the level of complexity of the learning materials. Clarke et al. ([7] p.9) defined ICL as "the mental work imposed by the complexity of the content." ICL imposed on WM is low when non-interacting elements are learned in isolation. For example, the ICL associated with reading spans across a huge continuum depending on the amount of knowledge stored in LTM. ICL and element interactivity decreases as reading becomes increasingly automated. A person who can confidently read would recognize a word such as "book"; it will be processed as a single chunk in WM. However, a person who does not recognize the word, but recognizes single characters, would process four chunks. This indicates that an element differs between learners. Hence, the element can be either a schema that is already learned or a unit of new information. The load on WM during the learning process is dependent upon the number of elements that has to be processed simultaneously, which depends on the extent of interactivity between the elements [28]. When developing e-learning systems, it is important to understand the level of a learner's prior knowledge to effectively manage the level of ICL that is associated with the task.

2.2 Extraneous Cognitive Load

Extraneous cognitive load (ECL) results from poorly designed instructional materials [32]. ECL is imposed by the actual instructional techniques, procedures, and materials used during instruction, for example, an overly complex e-learning system where the learner spends more mental effort understanding the system than learning the concepts. ECL can interfere with schema acquisition and automation and, hence, hinder the learning process. Sweller [27] argues that ECL is governed by the instructional process, which gives the developer the ability to vary the e-learning system developed to reduce ECL. Paas et al. [22] suggest that a reduction in ECL can be achieved by employing a more effective instructional design. It is very important not to expose WM to an unnecessary ECL while the learner is in the process of constructing and acquiring schemas. This should be taken into consideration when developing e-learning tools.

2.3 Germane Load

Germane load (GL or relevant load) is the remaining capacity present in WM; it can be redirected from ECL to schema acquisition [32]. GL is the process of constructing and storing schemas in LTM. Instructional design methods can augment GL, especially when learning with worked examples. The variability effect suggests that the variability of tasks will enlarge the cognitive load but meanwhile develop the efficiency of learning [23]. The process of constructing schemas also provides benefits through motivating learners to connect concrete worked examples with abstract knowledge when using e-learning tools.

2.4 The Use of CLT in Instructional Design

E-learning tools require skilled instructional developers for them to be effective. However, developers face challenges that are different from those usually experienced in classroom instruction [18]. E-learning puts greater responsibility on the instructor to make sure sufficient assistance is given to the learner in many different ways. Just as important, the learner learns about the educational content at the same time that they are learning about delivery technologies. This has the potential to impose greater cognitive load on learners, specifically those who have limited experience with technologies [8]. While simple e-learning settings may easily be designed, these designs may not have certain essential properties that are vital to achieve successful deep learning, for example, learner engagement and flexibility [11]. E-learning has the potential to increase learners' ECL as they have to deal with the technology before they can focus on the learning and has the potential to stall learning if the instructional design does not manage or account for increased cognitive demands. In order to be effective, e-learning designs should balance an interactive and interesting environment with manageable mental effort levels. Thus, it is necessary for developers not only to know and understand the causes of increased cognitive demands put on learners but also to measure their perceptions of tasks in e-learning in order to create efficiency from learner perspectives.

The cognitive load structure can be applied effectively to various forms of dynamic presentations within e-learning tools such as instructional games, simulations, and animations. For example, the continuous presentation of animations may be too challenging for a novice learner because of a high level of transitivity. A novice learner may be helped by replacing animations with a series of comparable static charts and diagrams. The impacts of computer animations and static diagrams on learners' levels of understanding and comprehension were investigated by Hegarty et al. [10], who argued that there was no evidence proving that animations resulted in higher levels of understanding than static diagrams. The understanding of diagrams, however, was improved by asking learners to forecast how the object from the static diagrams would behave and by giving them an oral explanation of the dynamic processes. Despite this finding, animated diagrams could still be more favorable for a more experienced learner who has attained an adequate knowledge base. Kalyuga [12] also argued that the relationship between the expertise level of the learner and the efficacy of static and animated illustrations indicated that the performance of less experienced learners was notably higher after studying static illustrations. However, more experienced and knowledgeable learners exhibited better performance after studying animations. As the expertise level of the learner increased, the animated instruction group's performance progressed to a higher level of learning as opposed to the static group's performance. Static graphics are less useful to more experienced learners since their existing dynamic knowledge structures have to be assimilated with redundant details shown in graphics. Increased cognitive resources may also be required for such processes, decreasing the effects of relative learning and raising WM demands. Interactive animations can offer suitable environments for learning; however, it is important to consider the learners' level of domain expertise when designing the e-learning environment.

Expertise levels may change as learners' experience in a certain task domain increases. Hence, the tailoring process must be active, taking into account learners' levels of expertise in real time as they steadily change throughout a learning session. As the expertise levels of the learners increase, e-learning systems designed around problem solving, based on games, or using less-guided probing environments may efficiently support the learning of advanced skills and knowledge in particular task domains.

The mental resources expended by the learner when learning and conducting various activities are restricted by the size of WM. This represents a key factor affecting the resourcefulness and efficacy of learning. As mentioned earlier, WM may experience an overload if more than a few chunks of information are processed at the same time, resulting in the inhibition of learning. The LTM, on the other hand, is not restricted in duration and capacity and significantly impacts on the operation of WM. LTM lets the learner deal with numerous interacting aspects of information in terms of bigger units or chunks in WM, which results in the reduction of cognitive load. The related cognitive attributes of the learner and the knowledge structures available may greatly affect the efficacy of many different methods of instruction. Hence, for it to be effective and resourceful, methods and learning programs that are ICT based should be modified to be apt to the cognitive attributes of the learner.

3 Human–Computer Interaction (HCI) and E-Learning

HCI is defined as a developed system that attempts to comprehend how people can interact with a computer from an organizational perspective [33]. The key to quality HCI lies in well-defined interaction design that mimics the human–computer engagement process. Hence, HCI seeks to understand how the human–computer interface can be achieved through high-quality design that not only performs and is usable but incorporates human elements such as the physical and social experiences of the interaction. In the design of interaction devices involving perceptual or motor issues, accessible expertise of ergonomics and human factors was implemented [25]. The type of computer users became wider in the mid-1980s, and the use of computers for educational purposes started to emerge, for example, interactive learning environments and training simulators. The various application environments for an individual or group were then taken into account: from work to education to entertainment and in locations such as the home or in a mobile context.

Sharp et al. [25] stated that consideration of the learning process should be considered a factor in HCI for two main reasons: novice users need to learn how to operate a computer system to achieve certain tasks and to support the acquisition of various knowledge and skills using educational software. A training-wheels method was proposed [3] for learning how to operate computer systems. Only the fundamental functions are made available for beginners, and the possible functions can be extended for users with more expertise. These concepts support the principles of intrinsic cognitive load (ICL); new information can be learned in isolation, and the design only focuses on system elements that are designed to achieve the learning tasks.

There are two major research branches that have been emphasized with regard to the user-centered design of educational software. One branch is the evaluation of educational applications [15, 35], and the other is user-centered development approaches for e-learning applications [15]. Tselios et al. [34] used traditional usability measures to rate two e-learning systems with two groups of learners tested on the two different learning systems. The system that was rated more usable obtained considerably better learning results. The concept of usability in the context of educational computing still has to be improved to suit pedagogical approaches and theories of learning. Tselios et al. [35] also argued that learning systems should not limit their focus to efficiency of task execution, as this may have a negative effect since it can hamper learning processes associated with exploration. Koohang and Du Plessis [15] identified that a cross-functional design team that combines both developers and educational experts would be beneficial for the integration of user-centered technology design and learner-centered instructional design.

Both HCI and CLT are based on the same fundamental theories of cognition developed in the 1970s and 1980s. Both of the theories have their focus on the reduction of unrelated cognitive load. Some CLT instructional design principles have the potential to be applied to e-learning system design in a similar way when compared to usability goals and principles, particularly the CLT principles of split-attention, the redundancy effect, and element interactivity.

3.1 Split-Attention

Split-attention refers to a learner's unnecessary splitting of attention when trying to understand instructional material across multiple sources, for instance instructional material consisting of a diagram with a textual explanation. Such instructional material requires a learner to attend to both sources of information causing the split-attention effect [31]. If the instructional material can only be understood by mentally integrating the multiple sources and the additional information does not assist in learning, then the portion of WM needed to integrate the multiple sources is allocated and hence hinders learning. Split-attention is observed when ICL is high and when it exceeds the capacity of WM when combined with ECL. Sweller et al. [31] compared students who were learning using conventional teaching material incorporating diagrams and separate textual explanations with students taught using material that physically integrated the diagrams and text into a single source of information. This outcome of the experiment revealed that the students studying from integrated material performed better than those using conventional material; this reduced ECL. Chandler and Sweller [6] observed split-attention in computer instruction where conventional instructions were compared with integrated format materials. These studies also found that learners performed better using the integrated format. Therefore, when developing e-learning systems it is important to consider the placement of information.

3.2 Redundancy Effect

Instructional material design, which aims to present information in a format with *both* diagrams and accompanying text, may not always be an effective procedure as some information may be unnecessary or redundant. Redundancy can increase cognitive load; this interferes with the learning process. The interference caused by unwanted information is known as the redundancy effect, where the same information is presented in different forms [5]. Chandler and Sweller [5] tested for a redundancy effect using instructional material. They compared groups studying from three different instructional formats: a conventional separate diagram and text with textual material below the diagram, an integrated diagram and text format with textual material in the diagram next to the relevant part, and a diagram only format with labels. The results indicated that learners required significantly less time to understand the information in the diagram only format and performed better in this test. This revealed the importance of removing unnecessary information and showed that integrating redundant text with a diagram can hinder learning. Processing the redundant information uses cognitive resources, imposing an unnecessary cognitive load.

3.3 Element Interactivity

Three types of cognitive load impact on WM over time [22]. The amount of information a learner must process over a period of time is certainly important, but the most important factor is the complexity of that information. According to Sweller and Chandler [30], instructional content is composed of component parts or "elements." These elements may be said to *interact* if there is a relationship between them, thus raising the complexity of the instruction. This phenomenon is described as "element interactivity."

It is easy to understand low element interactivity material [32], and it can be learned procedurally rather than simultaneously, without imposing ECL on WM. On the other hand, high element interactivity involves many elements interacting simultaneously and imposes a heavy load on WM. Sweller [28] demonstrated that a learner may learn when elements connect and interact with other elements; this understanding applies only when material to be processed is of high element interactivity. Sweller [27] suggested that element interactivity cannot be measured independent of the learner as the elements are affected by each individual's knowledge. As the novice increases his expertise, schemas are acquired, and the process of learning starts to become automated; elements that were previously required to be processed individually are now processed as a single element.

3.4 Integration of HCI Concepts and CLT Concepts

The concepts and principles of CLT and HCI have already been integrated to some extent by researchers. For example, Oviatt [21] suggested that the usability principle "making a system easy to use and learn" could reduce ECL, and "designing a usable learning environment" was proposed to be beneficial for reducing ECL and therefore improving the learning process. In the same way, Chalmers [4] described CLT principles for reducing ECL to be methods for enhancing the usability of educational software. In a study of learners using a spreadsheet application for mathematics, Clarke et al. [8] noted that learners' cognitive load was decreased when they were taught the instructions of spreadsheet applications prior to the mathematics, as opposed to being taught the spreadsheet and mathematics skills simultaneously. Van Nimwegen et al. [36] illustrated the use of CLT concepts, particularly the concept of GL, to clarify why the use of externalizing information on the interface to decrease cognitive load will have negative effects in some cases; findings were presented through a study of a conference booking system. Compared to systems without externalized support information, systems with more externalized information can lead to reduced thinking, planning, and knowledge acquisition.

The split-attention principle—i.e., when pieces of information are linked to each other, they are all required for a task [4]—has a striking similarly to the usability heuristic principle: "The user should not have to remember information from one part of the dialogue to another" [20]. Therefore, related information should be exhibited appropriately on the screen. The redundancy principle has links with the usability heuristic "every extra unit of information in a dialogue competes with the relevant units of information" [20]; thus, information redundancy on the screen should be avoided. Both CLT and HCI theories identify that individual learner characteristics play a significant role. Some other CLT principles that are designed to reduce cognitive load and to reduce ECL, such as the worked-example principle and the modality principle, seem to have no corresponding reference in HCI theories, probably because these principles are related to the learning processes and not instructional design.

Previous research has provided two conceptual models to clarify research into education software design. The first model demonstrates findings from prior research showing that cognitive load induced by using a software tool can be represented as part of ECL [21], and another component of ECL is created from the instructional design itself. Cognitive load can be reduced by either designing highly usable software or by training learners to use the software [21]. The component of cognitive load due to software use can be ignored if software is designed to be very easy to use or the software usage can be automated through training. The model also demonstrates the relationship of CLT principles with traditional HCI design principles by focusing on reducing what CLT refers to as ECL [6]. Conventional usability principles are insufficient to increase learning because there is still a need to ensure the use of GL in the learning process; however, this requires a greater understanding of educational principles and/or expertise by the e-learning tool developer.

4 Conclusion

Increased learning outcomes can be achieved through the application of CLT principles to the design of e-learning systems, creating systems with structures that reflect the construction of knowledge in the human brain. Prior research by Sweller [27] confirms that the users' cognition, which is influenced by the design of e-learning systems, is significant to knowledge acquisition. Despite the fact that split-attention occurs naturally, effectively designed e-learning systems provide increased learning outcomes as they provide learners with information that can be regulated. Therefore, understanding users' cognition when designing e-learning systems is of great significance for achieving increased learning outcomes.

This conceptual paper has discussed the increasing role of ICT in education and the role of e-learning in reducing the cognitive load faced by learners. It has presented a number of areas relating to the cognitive load of learners, beyond simple interface design, that should be considered in the design of e-learning tools. It is critical to understand the different ways in which cognitive load is affected by the use of e-learning tools. This paper also provided an empirical exploration into the relationship between cognitive load and e-learning strategies, as demonstrated by previous studies. Finally, this paper presented some of the similarities between HCI's focus on improving system design and CLT's methods to reduce cognitive load.

References

1. Alasraj A, Freeman MB, Chandler P (2011) Considering cognitive load theory within e-learning environments. Paper presented at the Pacific Asia conference on information systems (PACIS) 2011, Brisbane, Australia

2. Blin F, Munro M (2008) Why hasn't technology disrupted academics' teaching practices? Understanding resistance to change through the lens of activity theory. Comput Educ 50:475–490. doi:10.1016/j.compedu.2007.09.017
3. Carroll JM (1990) The Nurnberg funnel: designing minimalist instruction for practical computer skill. MIT Press, Cambridge
4. Chalmers PA (2003) The role of cognitive theory in human-computer interface. Comput Hum Behav 19:593–607. doi:10.1016/S0747-5632(02)00086-9
5. Chandler P, Sweller J (1991) Cognitive load theory and the format of instruction. Cognition Instruct 8:293–332. doi:10.1207/s1532690xci0804_2
6. Chandler P, Sweller J (1996) Cognitive load while learning to use a computer program. Appl Cognitive Psych 10:151–170. doi:10.1002/(SICI)1099-0720(199604)10:2
7. Clark RC, Nguyen F, Sweller J, Baddeley M (2006) Efficiency in learning: evidence based guidelines to manage cognitive load. Perform Improv 45:46–47. doi:10.1002/pfi.4930450920
8. Clarke T, Ayres P, Sweller J (2005) The impact of sequencing and prior knowledge on learning mathematics through spreadsheet applications. Educ Technol Res Dev 53:15–24. doi:10.1007/BF02504794
9. Hauck R (2008) Good practices for e-learning in higher education courses. Paper presented at the World conference on e-learning in corporate, government, healthcare, and higher education 2008, Las Vegas, NV
10. Hegarty M, Kriz S, Cate C (2003) The roles of mental animations and external animations in understanding mechanical systems. Cognition Instruct 21:209–249. doi:10.1207/s1532690xci2104_1
11. Herrington J, Reeves TC, Oliver R (2006) Authentic tasks online: a synergy among learner, task, and technology. Dist Educ 27:233–247. doi:10.1080/01587910600789639
12. Kalyuga S (2008) Relative effectiveness of animated and static diagrams: an effect of learner prior knowledge. Comput Hum Behav 24:852–861. doi:10.1016/j.chb.2007.02.018
13. Kennedy DM, McNaught C (1997) Design elements for interactive multimedia. Aust J Educ Tech 13:1–22
14. Kocur D, Koščˇ P (2009) E-Learning implementation in higher education. Acta Electrotech et Inform 9:20–26
15. Koohang A, Du Plessis J (2004) Architecting usability properties in the e-learning instructional design process. IJEL 3:38–44
16. Mayer RE, Moreno R (2003) Nine ways to reduce cognitive load in multimedia learning. Educ Psychol 38:43–52. doi:10.1207/S15326985EP3801_6
17. Miller GA (1956) The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychol Rev 63:81–97. doi:10.1037/h0043158
18. Morrison GR, Anglin GJ (2005) Research on cognitive load theory: application to e-learning. Educ Technol Res Dev 53:94–104. doi:10.1007/BF02504801
19. Muldoon N (2008) Self-direction and lifelong learning in the information age can PLEs help? Paper presented at the Lifelong learning: reflecting on successes and framing futures. Keynote and refereed papers from the 5th international lifelong learning conference, Rockhampton, 16–19 June 2008
20. Nielsen J (2005) 10 Heuristics for user interface design. http://www.useit.com/papers/heuristic/heuristic_list.html. Accessed 26 Mar 2012
21. Oviatt S (2006) Human-centered design meets cognitive load theory: designing interfaces that help people think. In: Multimedia 2006. 14th annual ACM international conference on multimedia, Santa Barbara. pp 871–880. doi:10.1145/1180639.1180831
22. Paas FG, Renkl A, Sweller J (2003) Cognitive load theory and instructional design: recent developments. Educ Psychol 38:1–4. doi:10.1207/S15326985EP3801_1
23. Paas FG, Van Merriënboer JJG (1994) Variability of worked examples and transfer of geometrical problem-solving skills: a cognitive-load approach. J Educ Psychol 86:122–133. doi:10.1037/0022-0663.86.1.122
24. Peterson L, Peterson MJ (1959) Short-term retention of individual verbal items. J Exp Psychol 58:193–198. doi:10.1037/h0049234

25. Sharp H, Rogers Y, Preece J (2007) Interaction design: beyond human–computer interaction, 2nd edn. Wiley, Chichester
26. Stephenson J (2001) Teaching & learning online: pedagogies for new technologies. Stylus, Sterling
27. Sweller J (1994) Cognitive load theory, learning difficulty, and instructional design. Learn Instr 4:295–312. doi:10.1016/0959-4752(94)90003-5
28. Sweller J (1999) Instructional design in technical areas. Australian Council for Educational Research Press, Camberwell
29. Sweller J (2003) Evolution of human cognitive architecture. Psychol Learn Motiv 43:215–266
30. Sweller J, Chandler P (1994) Why some material is difficult to learn. Cognition Instruct 12:185–233. doi:10.1207/s1532690xci1203_1
31. Sweller J, Chandler P, Tierney P, Cooper M (1990) Cognitive load as a factor in the structuring of technical material. J Exp Psychol Gen 119:176–192. doi:10.1037/0096-3445.119.2.176
32. Sweller J, Van Merrienboer JJG, Paas FG (1998) Cognitive architecture and instructional design. Educ Psychol Rev 10:251–296. doi:10.1.1.89.9802
33. Te'eni D, Carey J, Zhang P (2007) Human computer interaction: developing effective organizational information systems. Wiley, Hoboken
34. Tselios N, Avouris N, Dimitracopoulou A (2001) Evaluation of distance learning environments: impact of usability on student performance. Int J Educ Telecommun 7:355–378
35. Tselios N, Avouris N, Komis V (2008) The effective combination of hybrid usability methods in evaluating educational applications of ICT: issues and challenges. Educ Inf Technol 13:55–76. doi:10.1007/s10639-007-9045-5
36. Van Nimwegen C, Van Oostendorp H, Burgos D, Koper R (2006) Does an interface with less assistance provoke more thoughtful behavior? In: 7th international conference on learning sciences, Bloomington. pp 785–791

Methodological Framework for the Transition from Requirements to Software Architectures

Matthias Galster, Armin Eberlein, and Li Jiang

Abstract The quality of information systems heavily depends on software requirements engineering and software architecting. Both activities are the foundation for subsequent tasks, such as software development project planning, implementation, or maintenance. This paper introduces a methodological framework to support the transition from requirements to architectures. The framework consists of a process and different process modules (e.g., for systematic software architecture style selection). We also present a case study which shows higher modularity and lower coupling of architectures created using the proposed framework when compared to architectures created without the framework.

1 Introduction

Requirements engineering (RE) discovers the purpose of an information system by specifying its software requirements and manages these requirements throughout the software life cycle. RE includes systematic procedures to ensure understandable, relevant, and traceable requirements [24]. Software architectures on the other hand describe the structure of a software system and its fundamental design decisions. A good architecture helps implement all requirements, at an acceptable level of

M. Galster (✉)
University of Groningen, Groningen, The Netherlands
e-mail: mgalster@ieee.org

A. Eberlein
American University of Sharjah, Sharjah, UAE
e-mail: eberlein@ucalgary.ca

L. Jiang
University of Adelaide, Adelaide, SA, Australia
e-mail: li.jiang@adelaide.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_11,

quality and within budget. Software architectures encompass design decisions, which, if incorrect, can cause project failure. Much work on architectural description, analysis, and refinement can be found, whereas little work explicitly links RE and architectures [2]. Drawing a line between RE and architecting is not realistic as requirements and architectures impact each other [26]. Furthermore, architecting starts as soon as a good understanding of requirements exists. Later, architectures evolve. Thus, moving from RE to architectures is an iterative and interleaved process that tends to be informal and relies on extensive communication as key factor. Problems occur as software architectures often solve the wrong problem or are not understood by developers. The research presented in this paper investigates how to support the transition between software requirements (or RE) and architectures. We are interested in how architecting can be supported already during RE. Several reasons contribute to the need for a more systematic transition:

- The transition is often based on experience, intuition, and domain knowledge of software engineers. Thus, the quality of architectures depends on the skills of individuals [3]. Furthermore, informal transitions make it difficult to trace architectural decisions. This causes difficulties when making changes later.
- Architectures are the foundation for project management, implementation, maintenance, etc. and therefore should implement all requirements. An architecture is useless if we cannot use it during system implementation.
- RE aims at overcoming problems that occur because of incorrect or missing requirements, or the lack of communication of system features amongst stakeholders. Even if RE is carried out, the effort spent on RE becomes futile if we fail to communicate requirements throughout development. Poor architectures waste effort as we later have to make the implementation fit requirements.

The goal of this paper is to describe a methodological framework for the transition from software requirements to architectures. The framework uses four modules, each using different methods and techniques. These modules have been proposed as part of our previous research (see Sect. 3). This paper focuses on the integration of the modules in a framework and on the evaluation of the framework. Section 2 of this paper presents background and related work. Section 3 introduces our methodological framework for the transition between software requirements and architectures. In Sect. 4, we present a case study. Limitations of the proposed framework are discussed in Sect. 5 before we conclude in Sect. 6.

2 Related Work

In brief, previous work on the transition from requirements to architectures can be separated into four categories:

1. Approaches which expand RE or architecture methodologies (e.g., by annotating RE methodologies to also describe architectural concerns). Examples include problem frames [21] or use case maps [1].

2. Approaches that focus on the actual transition. Examples include object-oriented transitions [23] or weaving RE and architecting [18].
3. Mapping of requirements one at a time to architecture representations (e.g., partial behavior trees or partial component-based architectures [4], constructing component-based systems [17], or behavior tree modeling [13]).
4. Approaches with highest similarity to our work include the integration of the solution of requirements conflicts and architectural design [14], and the codevelopment of system requirements and functional architecture [19]. In contrast to these proposals, our work helps design systems from scratch. Also, our work allows architecture refinement through iterative architecture style selection. Furthermore, our work focuses less on consistency and negotiation of requirements and architecture but on the transition process in general with consistency management as an implicit activity while at the same time providing flexibility in the details of a transition.

Approaches for the transition from requirements to architectures fall between two extremes: (1) formalization, which requires tremendous learning effort and provides little flexibility [6], and (2) expert-based approaches, which do not follow systematic steps and are usually expensive and error prone. Moreover, many approaches do not trace design rationales. However, this is important for practitioners [26]. In summary, several issues need to be addressed:

- The transition from requirements to architectures needs to be considered already during RE.
- The input of the transition needs to be clarified through standardized requirements specifications.
- The output of the transition needs to be clarified through architecture specification templates.
- The impact of individual requirements on the architecture needs to be considered.

3 Proposed Methodological Framework in a Nutshell

The framework (Fig. 1) consists of a process and process modules. The implementation details of the individual modules were previous work, as referenced in this section. In this paper we show how they are integrated into a framework.

We separate modules from the process steps as modules allow more flexibility in the process: Modules can be replaced as long as the input and the output of models are compatible. For example, in Module 1, a requirements specification technique can be replaced with another one, as long as atomic requirements are produced. Even though Fig. 1 shows a sequential process, iterations are possible.

Module 1 specifies requirements for architecting. This is necessary as requirements can be described in different ways, e.g., in natural language, graphical models, or formal methods. It generates atomic requirements appropriate for architecting, independent of how the RE process used in a project specifies requirements.

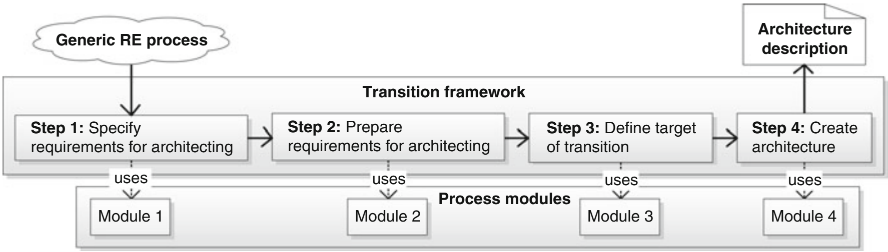


Fig. 1 Overview of proposed framework

Atomic requirements do not differentiate functional and nonfunctional requirements as all requirements are operationalized. As atomic requirements have a smaller scope, they are less likely to spread across the architecture and can be traced more easily. How to apply atomic requirements for architecting is explained in [8] and their use has been evaluated in [12]. In general, each atomic requirement should be written in natural language and should be described so that they cannot be functionally split any further and so that their testing only results in pass or fail. The starting point of Module 1 can be any phase in a generic RE process, e.g., elicitation or documentation. The output is a set of atomic requirements.

Module 2 classifies requirements based on their impact on the architecture. Often, global factors, e.g., quality attributes or technical product constraints, are factors that influence the design of architectures [26]. However, characteristics of individual requirements also impact the architecture process. Thus, systematic handling of this impact facilitates the design of architectures. We express characteristics of requirements as requirements attributes. We assign ten attributes to each requirement (effort; risk; complexity; is early design implied by requirement or not; dependencies between requirements; volatility; are hardware constraints implied by requirement or not; addressed architectural concerns, such as business logic, data, internal or external communication, or user interface; importance of requirement; familiarity of the development team with requirement). Values for these attributes are assessed by requirements engineers and domain experts. We then rank requirements based on the values for attributes effort, risk, complexity, volatility, importance, and familiarity assigned to each requirement. This ranking determines the order in which requirements are implemented. Also, requirements with similar attribute values are implemented in the same architecture element as they require the same treatment. For example, requirements that are volatile and highly complex are encapsulated to increase modifiability. If early design is implied by a requirement, then this prescribes design choices at a very early stage. Dependencies between requirements could indicate that connectors between architecture components are needed. Implied hardware constraints help assign functionality of the system to hardware. Finally, based on the architectural concern addressed by a requirement, we determine what primary architecture artifact it could relate to (processing component, data component, connector). For instance, if requirements with similar attribute values address business logic, these requirements can be assigned to a processing component. Details on how to rank requirements can be found in [7].

Module 3 addresses the issue that it is often not clear what the target of a transition is because architectures can be represented in different ways. For example, a transition could result in a set of UML diagrams or a textual description of the architecture. Also, stakeholders and concerns which should be addressed in the architecture have to be defined. The representation depends on the purpose of the architecture, resources available, and the expertise of developers involved. Thus, Module 3 provides a taxonomy to describe architecture representations in terms of how the architecture is specified (e.g., objectives, creator), concepts used to describe representations (e.g., architecture elements and relationships), and a notation. Examples for concrete perspectives include process perspectives or code perspectives. The taxonomy has to be instantiated for each project. For each project and perspective we have to decide on objectives, creators, architecture elements, etc. on a case to case basis. Details on the taxonomy can be found in [9].

Module 4 relates requirements and architecture elements. It helps derive architecture elements and how to integrate them in the architecture. As architecture style selection is often the first major architectural decision, this module includes an approach for systematic architecture style selection [11]. It collects input from different stakeholders about their architectural concerns and their requirements on basic architecture elements, e.g., data components, processing components, connectors, or subsystems (e.g., modifiability, support for large-scale data processing). A trade-off analysis suggests an architecture style based on the ratings of the stakeholders. Based on the selected architecture style, architecture element types are determined. For example, if the client-server style is selected, architecture element types are clients and servers. Note that style selection can happen at several abstraction levels. For example, a style can be selected for the whole system, and separate styles can be chosen for each part of the system (e.g., pipe-and-filter style in each client derived from the client-server style). Also, based on the architecture style, architecture elements are arranged in the architecture. Again, referring to the example of the client-server style, the control and data topology could be star or hierarchical; binding time could be write time, compile time, or runtime; and data and control flow in opposite directions. At this point, a high-level blueprint for the architecture can be created. Next, we perform a set of steps to create architecture perspectives as defined in Module 3:

- Determine processing components: Initial processing components are created from groups of similar requirements which address business logic or user interface-related concerns.
- Determine internals of processing components, dependent on architecture perspectives: For example, for a process perspective, initial processes are created based on the individual requirements assigned to each processing component.
- Determine data components: Initial data components are created from requirements which address data. As with processing components, groups of similar requirements would be used to form a data component.

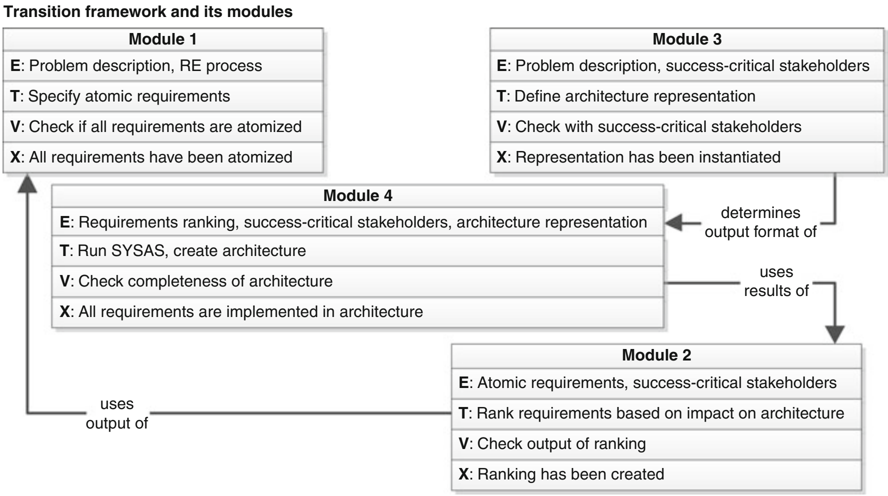


Fig. 2 Interaction of modules

- Create connectors: Connectors are created based on requirements which address internal or external communication. Furthermore, connectors are created based on dependencies between requirements that are assigned to different processing or data components.

Processing or data components can be merged or split. Besides graphical representations of the architecture, element catalogues list all architecture elements of a perspective, with name, description, and information, if instances of elements are created during runtime. Inputs for Module 4 are the outputs of Module 2 and the representation selected in Module 3. Outputs of Module 4 are sets of architecture elements, their relations and interactions, in different architecture perspectives.

The interactions of the four modules are shown in Fig. 2, using the ETVX paradigm [20]. Entry criteria (E) should be satisfied before using a module, i.e., entry criteria are items required for executing tasks (e.g., people, tools, artifacts). Tasks (T) describe what is to be accomplished by the module (e.g., processes, techniques). Validation procedures (V) verify the quality of items produced in a module. Exit criteria (X) should be satisfied by a module.

4 Case Study

To assess the methodological framework, we conducted a case study. A case study was chosen over surveys or experiments, because the transition from requirements to architectures has to be studied in its context. Furthermore, there are many variables in information systems design that cannot be controlled, such as the background of engineers involved in designing a system. We used a holistic (i.e., one unit of analysis) single-project study [15] that is exploratory in nature. The design of the case study followed guidelines suggested in [22].

4.1 Case Study Goal and Proposition

The goal of the study was to investigate if the proposed framework helps with the transition from software requirements to software architectures. Our unit of analysis was the transition process and the outcome of this transition, i.e., a software architecture. Based on our initial problem statement outlined in Sect. 1, the case study proposition is that using the proposed methodological framework has a positive impact on the development of software architectures compared to using ad hoc practices. We consider the proposition to be true if an architecture created using the framework is better than architectures created by software engineering students without the framework.

4.2 Case Study Design

The selected case is the design of a meeting scheduler system (MSS). The MSS was chosen because it is a real-world problem and exhibits the nature of real-world requirements for which no unique solution exists. Furthermore, it is a well-known example in the RE domain [5]. We use a comparative method to compare the "goodness" of the resulting architectures. Thus, data was collected from work products that were the result of performing the methodological framework. Also, we computed metrics for different architectures to compare the work results with architectures for the same system that were created without the framework. We used metrics proposed in [10], with metric values ranging between 0 and 1 (except for $MF(s)$ which ranges between 0 and the number of requirements implemented in an architecture):

- The modularity factor $MF(s)$ describes the degree to which system or subsystem s is composed of discrete architecture elements so that changes in one element have minimal impact on other elements. $MF(s)$ quantifies how requirements are distributed in s. We also use a normalized version of $MF(s)$.
- The internal coupling factor $CpFi(s)$ measures the degree of dependency between software modules of system or subsystem s. The internal coupling considers structural information, i.e., connections between modules, compared to functional relationships between requirements when calculating $MF(s)$. Thus, $CpFi(s)$ considers all possible pairwise sets of elements in s and checks whether the elements in the pair are related.

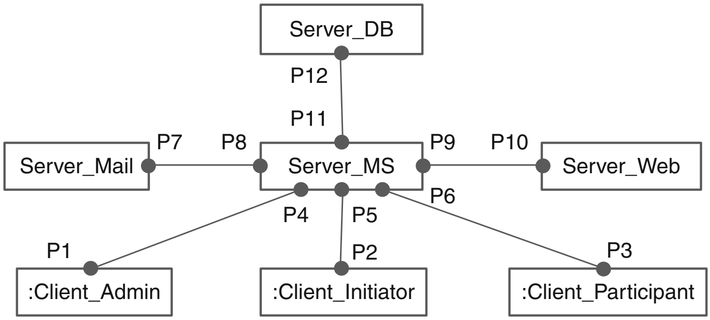


Fig. 3 Functional perspective of MSS

- The external coupling factor *CpFe*(*e*) differs from internal coupling as it describes the interdependence of individual architecture elements within other elements. Thus, *CpFe*(*e*) focuses on dependencies of element *e* with other elements instead of coupling within *e*.

4.3 Application of the Framework

Using Step 1 of the framework, we elicited 72 atomic requirements. During Step 2, we assigned attribute values to requirements and ranked requirements based on these attribute values. This means, we obtained information about when to implement requirements. In Step 3, we selected the functional perspective, the process perspective, as well as the physical perspective to describe the MSS architecture. In Step 4, a client-server style was chosen for the architecture. Requirements were mapped on these elements. This resulted in 44 architecture elements and six subsystems. Figure 3 shows a graphical representation of the functional perspective of the MSS created using the framework, with several server components and clients. The clients can have different instances (denoted by ":"). Also, server components can communicate with each other, i.e., may take the role of a client in certain operational modes. Based on the properties of the client-server style, a star topology was chosen. P1 to P12 denote ports of components to describe connectors between components. For example, the connector between P1 and P4 implements the management of users of the MSS. Data was collected from the architecture description by identifying the components, connectors, and the assignment of requirements to components and connectors.

4.4 Collection of Data from Students

We collected data from architectures for the MSS created by four student groups who completed the design of the architecture as part of a software engineering

Table 1 Data extracted from student projects

Item	Group 1	Group 2	Group 3	Group 4
# of source files (.java)	16	16	17	15
# of database files (text files)	4	2	2	2
# of layers (based on code structure and packages)	1	2	2	1
# of modules (based on directories)	1	4	4	1
# of architecture elements (components, i.e., classes)	20	18	19	17

Table 2 Metric values collected from architectures

Metric	Group 1	Group 2	Group 3	Group 4	FW-based
MF	4.50	4.50	4.24	4.80	2.02
NMF	0.94	0.94	0.94	0.93	0.72
CpFi	0.23	0.42	0.36	0.43	0.14
CpFe	0.11	0.20	0.18	0.21	0.34

course project at our university (Table 1). These architectures were created without the proposed framework. Students attending the course were in their fourth year of study and had already attended other software engineering courses. As students lacked industry experience, they can be considered as novice software engineers. Course projects were performed in groups of three students. To ensure that projects could be compared with regards to functionality, we performed an initial screening. We extracted the architectures of the course projects from Java source code because architecture and design documents changed when coding the system. To obtain the architecture from projects, we extracted class diagrams from the code, using jGRASP (www.jgrasp.org).

As external coupling is calculated for each architecture element and the architecture elements vary in each project, we determined the average external coupling for each project (Table 2). As we did not have information about where requirements had been implemented in the student projects, we could calculate modularity only for the whole system. We know that 72 atomic requirements had been implemented in the system. Also, when calculating modularity for student projects, we did not consider connectors as no functionality was implemented in connectors in the diagrams extracted in jGRASP. The metrics are shown in Table 2. "FW-based" denotes values for the architecture created using the proposed framework.

The average normalized modularity factor of all group projects is higher than the average modularity factor of individual subsystems of the architecture created with the framework (0.72). The same is true for *MF*. The average normalized modularity factor of group projects approaches 1 which could point to pathological systems. Also, the overall system modularity factor (not shown in Table 2) is slightly higher for the architecture created with the framework (0.95). Thus, the architecture created based on the framework exhibits similar or even better modularity. The average internal coupling of all group projects is higher than the internal coupling factors at system level of the architecture created using the framework (0.14). However, the average external coupling factor of all group projects is lower than in the

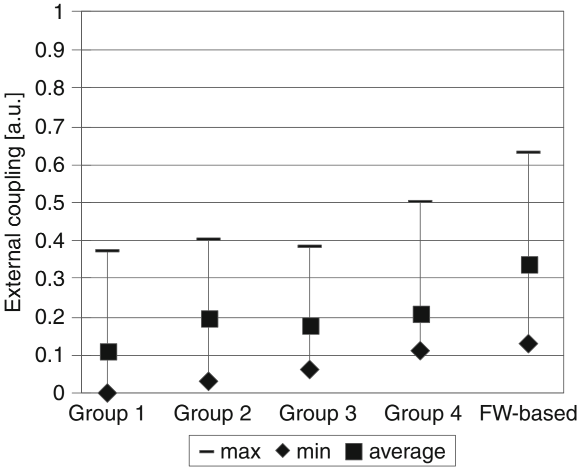


Fig. 4 Comparison of external couplings

architecture created with the framework (0.34). To further investigate the external coupling, Fig. 4 compares the average, minimum, and maximum *CpFe*.

From this we see that student projects have very low minimum coupling which might indicate pathological parts of the systems. On the other hand the high average external coupling in the framework-based architecture is caused by a component with high external coupling (<0.6). This component represents an outlier with regard to external coupling and thus would be subject to further analysis during an architecture evaluation stage. Thus, the architecture created with the framework appears to be better in internal coupling and at least as good with regard to external coupling. In summary, referring to the case study proposition, the architecture created with the proposed framework shows equal or better coupling and modularity than alternative architectures created without the framework.

4.5 *Threats to Validity of Case Study*

External validity refers to the ability to generalize the results: We used a case study well known in the RE domain and with a reasonable size. We compared case results with each other. Due to the required background in software engineering and architecture, only twelve participants were involved. The use of projects implemented by students should not diminish the results of this study. Even though we acknowledge the threat in generalizing the results to experienced software engineers, there is no indication that the results could not be generalized with regard to, at least, novice software engineers in industry [25]. Students are relatively close to the population of practitioners [16]. Yet, we cannot claim that we completely mitigated the risk of using students. Internal validity refers to the design of the study: We used students. In a different situation, stakeholders might be more mature and thus be able to more properly select architecture styles or to do attribute value assessment. Also, limitations of the case study are related to simplifications we made in the underlying model. These threats to the internal validity refer to the validity of the methodological framework itself and are described below. Overall, the presented case study is considered a preliminary evaluation and more empirical studies need to be done.

5 Limitations of Methodological Framework

We argue that investing effort proactively is more helpful and less risky than acting reactively if problems in the development process occur later. As with most approaches in software engineering, the quality of the methodological framework relies on the accuracy and availability of the input data. Some of the input data (e.g., proper requirements, developer needs for style selection) might be difficult to get, especially if an organization does not maintain such data or lacks RE processes. Thus, it is likely that development companies with a higher process maturity would find the proposed methodological framework more suitable than companies with lower process maturity. Scalability indicates the ability of the system to handle large amounts of data or to be readily expandable without major structural changes. The current version of the framework requires effort from humans. Thus, in the case of large-scale applications, the amount of effort is expected to increase. Subjectivity occurs with any topic in software engineering and in particular in the context of software architectures. This subjectivity occurs when performing activities, such as requirements attribute value assessment when ranking requirements, or when selecting styles. There could be several sources of subjectivity, such as different preferences or experience levels of requirements engineers and architects or different interpretations by architects and requirements engineers.

6 Conclusions

We presented a framework to support the transition from software requirements to architectures. The case study showed that architectures created with the framework can have at least similar, and in many cases better modularity and coupling than architectures created without using the framework. Also, the framework provides a means for explicitly capturing architecture design decisions and their rationale, and establishes traceability between software development artifacts.

One open research issue is related to the role of humans in the transition from requirements to architectures: How much automation is possible and how much creativity is necessary and feasible in a transition? It might be possible to reduce human intervention. However, it might be very difficult to fully automate the generation of software architectures. Another open issue is the capturing and sharing of experience and knowledge for future use. Knowledge reuse is most promising when developing software systems that are similar. However, reusing knowledge in completely different domains might be challenging. The question is how much reuse is reasonable to reduce effort. One possible direction for future work is the integration of reference architectures in Step 3 of the framework.

References

1. Amyot D, Mussbacher G (2001) Bridging the requirements/design gap in dynamic systems with use case maps (UCMs). In: 23rd International conference on software engineering, Toronto, Canada, pp 743–744
2. Avgeriou P, Grundy J, Hall JG, Lago P, Mistrik I (2011) Relating software requirements and architectures. Springer, Berlin
3. Buschmann F (2009) Introducing the pragmatic architect. IEEE Software 26(5):10–11
4. Dromey RG (2003) From requirements to design: formalizing the key steps. In: First international conference on software engineering and formal methods, Brisbane, Australia. IEEE Computer Society, pp 2–11
5. Feather MS, Fickas S, Finkelstein A, van Lamsweerde A (1997) Requirements and specification exemplars. Auto Software Eng 4(4):419–438
6. Finney K (1996) Mathematical notation in formal specification: too difficult for the masses? IEEE T Software Eng 22(2):158–159
7. Galster M, Eberlein A (2011) Facilitating software architecting by ranking requirements based on their impact on the architecture process. In: 18th IEEE international conference and workshops on the engineering of computer-based systems (ECBS), Las Vegas, NV, IEEE Computer Society, pp 232–240
8. Galster M, Eberlein A, Moussavi M (2007) Atomic requirements for software architecting. In: International conference on software engineering and applications (SEA'07), Cambridge, MA, Acta Press, pp 143–148
9. Galster M, Eberlein A, Moussavi M (2008) Defining the output of a software architecture process. In: International conference on software engineering and applications (SEA'08), Orlando, FL, Acta Press, pp 64–69
10. Galster M, Eberlein A, Moussavi M (2008) Early assessment of software architecture qualities. In: 2nd IEEE international conference on research challenges in information science, Marrakech, Morocco, IEEE Computer Society, pp 81–86
11. Galster M, Eberlein A, Moussavi M (2010) Systematic selection of software architecture styles. IET Software 4(5):349–360
12. Galster M, Eberlein A, Moussavi M (2010) Textual software requirements specifications in the context of software architecting. In: 22nd International conference on software engineering and knowledge engineering, Redwood City, CA, KSI, pp 42–47
13. Gonzalez-Perez C, Henderson-Sellers B, Dromey RG (2005) A metamodel for the behavior trees modelling technique. In: Third international conference on information technology and applications, Sydney, Australia, IEEE Conputer Society, pp 35–39
14. Herrmann A, Paech B, Plaza D (2006) ICRAD: an integrated process for the solution of requirements conflicts and architectural design. Int J Softw Eng Know 16(6):917–950
15. Kitchenham B, Pickard L, Pfleeger SL (1995) Case studies for method and tool evaluation. IEEE Software 12(4):52–62
16. Kitchenham BA, Pfleeger SL, Pickard LM, Jones PW, Hoaglin DC, Emam KE, Rosenberg J (2002) Preliminary guidelines for empirical research in software engineering. IEEE T Software Eng 28(2):721–734
17. Lau K-K, Nordin A, Rana T, Taweel F (2010) Constructing component-based systems directly from requirements using incremental composition. In: 36th EUROMICRO conference on software engineering and advanced applications, Lille, France, IEEE Computer Society, pp 85–93

18. Nuseibeh B (2001) Weaving together requirements and architectures. IEEE Software 34(3): 115–117
19. Pohl K, Sikora E (2007) COSMOD-RE: supporting the co-design of requirements and architectural artifacts. In: 15th IEEE international conference on requirements engineering, Delhi, India, IEEE Computer Society, pp 258–261
20. Radice RA, Roth NK, O'Hara AC, Ciarfella WA (1985) A programming process architecture. IBM Syst J 24(2):79–90
21. Rapanotti L, Hall JG, Jackson M, Nuseibeh B (2004) Architecture-driven problem decomposition. In: 12th International requirements engineering conference, Kyoto, Japan, IEEE Computer Society, pp 80–89
22. Runeson P, Hoest M (2009) Guidelines for conducting and reporting case study research in software engineering. Empir Softw Eng 14(2):131–164
23. Sangwan R, Neill C, Bass M, Houda ZE (2008) Integrating a software architecture-centric method into object-oriented analysis and design. J Syst Softw 81(5):727–746
24. Sommerville I, Sawyer P (1997) Requirements engineering – a good practice guide. Wiley, New York, NY
25. Svahnberg M, Aurum A, Wohlin C (2008) Using students as subjects – an empirical evaluation. In: 2nd International symposium on empirical software engineering and management, Kaiserslautern, Germany, ACM, pp 288–290
26. van Heesch U, Avgeriou P (2011) Mature architecting – a survey about the reasoning process of professional architects. In: 9th Working IEEE/IFIP conference on software architecture, Boulder, CO, IEEE Computer Society, pp 260–269

Integrating Enterprise Architecture and IT Service Management

Nelson Gama, Pedro Sousa, and Miguel Mira da Silva

Abstract The increased adoption of IT by all organizations has led to more complex IT infrastructures, enforcing the need for guidelines that will allow the alignment and management between an organization's architectures. To answer this need, different frameworks in the IT Governance area have been proposed, namely the widely adopted IT Service Management framework (ITIL) as well as different frameworks following Enterprise Architecture principles. In a time when rationalization is necessary and convergence and aggregation are motto, maintaining these two approaches, which may in fact overlap in some points, is an inefficient waste of resources. In literature review and framework research, we noticed few scientific references regarding integration, which increases the theme's relevance. Considering that both these domains are complementary, this paper pro-poses the integration between ITIL and Enterprise Architecture principles, with Services as the integration key point.

1 Introduction

Information technology (IT) plays a fundamental role in organizations. The more important the role of IT, the more complex the IT infrastructure and the harder it is to manage. The growing demand on IT leads to the improvement of key concepts related to IT governance, in particular the ones connected to IT alignment with strategic objectives and cost reduction initiatives.

For many years now, different efforts have been made related to IT governance; however, results are far from what was expected, and the gap between IT and the results expected by the organizations' objectives leads to an increasing interest in alignment efforts and related frameworks. From these initiatives, two main

N. Gama (✉) • P. Sousa • M.M. da Silva
Dep. Eng. Informática, Instituto Superior Técnico, Av. Rovisco Pais,1, Lisbon, Portugal
e-mail: nelsongama@ist.utl.pt; pedro.sousa@link.pt

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_12,

approaches have had major relevance: Enterprise Architecture (EA) and IT service management (ITSM). EA summarizes the relevant components in an organization, how they are related, and how they fit and work together in different architectures and with different views [1, 2]. ITSM is a reference model with an integrated approach to effectively and efficiently deliver IT services, providing a better IT alignment with organizations' needs [3–5], with cost and risk reduction [6]. From ITSM area we have frameworks related to IT governance, like ITIL, COBIT, and CMMI for services, just to name a few. From these, ITIL has become ITSM standard, currently the most widely accepted framework for managing IT services in the world [6–9]. As a result, this study will pay closer attention to EA and ITIL.

Today, there is no fully complete framework that can be used as a comprehensive off-the-shelf IT governance framework to ensure the alignment between service management and the organization's concepts and artifacts. In fact, different frameworks are often used as complementary and, most of the times, simultaneously too. Beyond the difficulties associated with the governance of both initiatives, parallel EA and ITIL projects imply a duplication of investments and costs. Indeed, even with shared infrastructures, we cannot avoid a duplication of data repositories, procedures, and human resources or maintain different efforts (ITIL and EA) aligned. Although some have tried to merge these two approaches by identifying several benefits from the relationship and integration of ITIL and EA, the results are far from satisfactory [7, 10–12].

This paper provides a research in two widely used frameworks to manage and align IT with business, allowing guidelines to adopt a single initiative that involves the merger and alignment between EA and ITIL approaches. A merged approach would avoid the duplication of efforts, cost, and time through a unique path. Thus, this paper provides a proposal to adopt a common guidance. Through ITIL principles, we suggest using the services delivered as the key point of integration between EA and ITSM.

2 Research Methodology

The methodology applied in the course of this paper is Design Science Research Methodology (DSRM). We selected this methodology because we wanted to develop and validate a proposal to solve our problem but had no initial validated theory [13, 14]. DSRM is a typical problem-solving paradigm, addressing research through the development and evaluation of designed solutions in order to meet identified needs over interactive steps, enabling us to understand the nature of causes and design solutions [13, 14].

We follow DSRM according to a process model with sequence activities [15, 16]; each sequence corresponds to the following sections: Section 3 covers aims and objectives like awareness and recognition of a problem from a literature review, hence giving us the theoretical background and the topics to be addressed. Section 4 clarifies the main problem and our research questions, offering a tentative idea of how the issues might be addressed and defining the objectives for a solution. Section 5 presents the design of a proposal as an attempt to face the previously described problem. Finally, we present a "Demonstration" that solves an instantiation of the problem and is followed by the "Evaluation," which compares the results with the research questions. We end by proposing themes for further work.

3 Related Work

3.1 Enterprise Architecture

Enterprise Architecture (EA) is a coherent set of principles and the basis of knowledge and representation of the organization itself. It has become the methodology that best enables the planning and development of systems and IT aligned with business [1, 2, 17]. EA involves the design of different architectures and specifies the orchestration among them, being used to manage and align assets, people, operations, and projects to support business goals and strategies [18, 19].

Different needs, scopes, and authors have suggested distinct representations and architectural frameworks (e.g., ArchiMate, TOGAF, and DoDAF just to name a few), decompositions or domains, having in common principles like a holistic representation of organizations, relationships between artifacts and architectures, and independence and connection among layered architectures.

The decomposed representation of organizational layers (as seen in EA) usually shares the following architectures [1, 11, 17, 18, 20, 21]: business, process, application, information, and technology. The architecture alignment takes on particular relevance, and EA integrates each of these into a cohesive framework so as to obtain a coherent "blueprint" of the organization [22].

Besides EA principles, the implementation of a manageable and configurable EA requires the use of appropriate tools, including a vital data repository that backs up the development, storage, presentation, and enhancement of the organization's architecture representations [23]. The EA repository has many similarities with the one of a Configuration Management System (CMS/CMDB) from ITIL.

Even though EA principles make a clear reference to the correlation between architectures, they do not specify how to control and manage the different artifacts within them in a day-to-day basis.

3.2 ITIL

The acronym ITIL refers to Information Technology Infrastructure Library, a collection of five books with the best practices related to the effective and efficient management of IT: Service Strategy, Service Design, Service Transition, Service Operation, and Continual Service Improvement.

The current version of ITIL covers the major weaknesses identified in the previous versions, specifically being too focused on technology. Now, the focus is on the overall service life cycle management, allowing a wider coverage of IT governance to address the "business alignment aspect" [5, 11, 24].

ITIL framework is underpinned by the Configuration Management System (CMS), which is defined as a set of tools and databases (CMDBs) for collecting, storing, managing, and presenting data about all configuration items (CI) and the relationships among them that influence the business [24]. Everything can be recorded as a CI such as hardware, applications, and interfaces but also information about incidents, known errors, changes, people, manuals, and service-level agreements (SLAs). Conceptually, a CMS supports ITIL's management processes described for service management, but it is also a shared center of decisions and the global "as-is" of organizations' systems. A CMS has a lot in common with EA principles. However, ITIL framework makes no reference of how we can develop these architecture definitions. We can use ITIL in EA to design the service management of an organization because ITIL is a reference model for IT governance. However, ITIL neither provides a complete coverage for all layers within EA nor does it specify implementation details [25].

3.3 SOA and Business–IT Alignment

Service-oriented architecture (SOA) is a paradigm oriented to provide business agility in order to respond quickly to changes in business by re-architecting business processes, creating new ones, or architecting existent IT as services exposed in business services [26]. The loose coupling characteristic of SOA's agility works as a basis for achieving architectural alignment [27]. In fact, service composition and orchestration are an advantage of SOA paradigm [27]. Each service, in SOA, has different granularity and may support business processes, singly or with services choreography. Moreover, the meaning of service is different in each referential.

There are many different definitions of SOA [28]; some are more related with Web services, applications' functions, and software development [26, 29], while others consider SOA an architectural approach that emphasizes service concept and service consumers as a basis to structure the entire organization [30]. Therefore, we adopted the definition from ITIL [24] and Noran [30], in which a "service" means the creation and delivery of value to users by easily fulfilling their needs without having specific costs and risks, which are made up from a combination of people, processes, and technology. A service, in a business sense, is then defined as a provider–client interaction that creates and captures value [27]. Here, users know what services they want but do not know how they are delivered [27].

3.4 Relationship Between ITIL and EA

ITIL and EA have rarely been studied together, and no relevant results about the relationship (interactions) between these two approaches have been produced. Much of the previous work was focused on previous versions of ITIL, only covering service delivery and support [11]. Therefore, this subject remains without significant development, which increases our work's interest and relevance.

ITIL books promote the connection between ITIL and EA in Service Design [24] and recognize that architecture components should cover all areas of technology. However, the architectural conception is not considered in deployment and the distinction between EA and IT architecture is not clear; service architecture is in fact IT that we will define as IT services [24].

Braun and Winter [10] proposed an EA expansion to integrate ITIL (V2 at the moment) and SOA. In their proposal, EA is considered a key concept, but ITIL is only regarded as IT operations. EA provides an overview of the IT architecture, while ITIL was assigned as an essential part of the management processes to service delivery. Aligned with the IT services, SOA concept is also integrated into EA at the application architecture level. ITIL and SOA were integrated into EA as a framework to deliver IT services, focusing this research of integration only on delivered IT services.

Almost aligned with the previous study, Thorn [12] addresses the relation between ITIL and TOGAF but with a different focus. EA is regarded as a fundamental concept for organizational engineering, focusing on EA development, in which ITIL is included as a framework to an operation model for IT delivered services. In his research, Thorn argues that both frameworks can be used together by mapping the two approaches. TOGAF covers the development of EA and is involved in the product's conception life cycle, whereas ITIL ensures the delivery and management of IT services to users and consumers [12]. Despite the recognized need of different teams and tools, the two frameworks complement each other, since TOGAF needs an EA repository, while ITIL requires a CMDB.

A more recent research promoted by Nabiollahi [11] provides a service-based framework for EA to meet the ITSM requirements of ITIL V3, suggesting that EA should be extended to involve service architecture layer from ITIL Service Design [24]. The development of an architecture model for IT services is proposed, making it a service layer for EA. However, it does not clarify how to do it or the relationships among architectures.

Another research concerned with a more generic and technology-independent view on IT services was developed by Correia [7]. In this research, ITIL supports services from an operational perspective through a CMDB, while EA repository is used to store the architectures, sharing a common data model and the same ontology. Correia's research then suggests a common ontology, a meta-model, and the sharing of IT services, specifically the formal representation of framework concepts and their relationships, both repositories existing at the same time.

4 Research Questions

EA principles aim to represent organizations as a system, relating multiple architectures to their artifacts and components. The widespread scope of ITIL involves all organizational architectures, but it does not describe how to deploy it. Currently ITIL and EA teams work in different parts of the organization with little opportunity to share expertise. Initiatives to address the alignment between IT and organizational issues have lead to an overlap of the work developed. The efforts spent in managing organizational data in separate repositories, from different initiatives, EA and ITIL, might become unmanageable. Therefore, both approaches should be merged.

Our research question "How to integrate ITIL and EA initiatives in a single body, avoiding efforts and resources duplication, but keeping all the benefits" will be answered by the following secondary research questions:

- Which is the key point between the two approaches and how can it be defined?
- How to merge both approaches, considering the effort and magnitude needed to build and maintain coherent information and, especially, their relationship?
- Since the integration of all processes and artifacts makes it necessary to have a tremendous amount of additional development and maintenance effort, how to keep the merged information up-to-date?

The answer to our questions derives more from the process side of the solution than any other. The problem we are trying to address is organizational engineering and the solution encompasses a conceptual model, independent of tools or adopted frameworks.

5 Proposal

The alignment approach among SOA, EA layers, and ITIL was already depicted by Chen [27] and Braun [10]. However, both researches (based on ITIL v2) suggest a relationship among approaches and not their integration.

Therefore, we propose to merge both ITIL and EA initiatives in a single body restricting resources and efforts. The suggested solution allows the mapping and visualization of the organization's actual state, top-down and bottom-up. In EA parlance, this is equivalent to the "as-is" model and allows, from ITIL principles, service delivery through all architectures. The solution we offer encompasses all the EA principles with referred architectures (architecture layer in Table 1) and the relationship among them, following ITIL service management processes.

We started by identifying a set of concepts, keeping the ones common to EA and ITIL, with strong relation to main concepts. From all of them, we also identified the interfaces, keeping the loose coupling characteristic from SOA paradigm. To do this, from the relationships among SOA elements in EA frameworks [31] and among

Table 1 Aggregation of concepts

Architecture layers [1, 11, 17, 18, 20, 21]	Core EA artifacts (based on [21])	SOA elements [31]	ITIL artifacts and management processes (according ITIL [24])
Business	Business goals and objectives	Business service	Demand
	Business roles	Actor	Service portfolio
		Business interface	
		SLA and contract	
Processes	Business processes	Service conditions	Service catalogue
	Business functions	Product	Service level
		Service interface	Capacity
		Measure	IT service continuity
		Service conditions	Availability
		Service description	Information security
		Service policy	
Information	Conceptual/semantic data model	Application service	Service asset
	Data and logical data model	Application interface	Configuration
	Data management process models	IS service	Release and deploy
	Data entity/business function matrix	Service interface	Change
	Various data related views	Service description	CMS/CMDB
		Service policy	Secure library
Application	Programs		
	Applications		
	Software components		
Technology infrastructure	Hardware models	Infrastructure service	Application
	Communications models	Infrastructure interface	Technical
	Processing models	Platform service	IT operations
	Other technology models	Service interface	Operational
		Service description	
		Service policy	

the core artifacts of EA with cross-layer views in different frameworks [21], we introduced the main ITIL artifacts and management processes. The results are shown in Table 1.

To clarify the relationship among concepts, we used a model as a graphical representation, allowing people to recognize the relationship among concepts in different architectures or views. Models are effective artifacts that support understanding and communication [5, 32]. Concept maps are, by definition, a graphical representation to capture, represent, structure, and share tacit knowledge, specifically concepts and the relationships between them [32, 33]. Beyond a knowledge representation, a concept map is as an evaluation tool due to the development of high levels of cognitive performance [34].

EA represents an organization from a strategic output to technological infrastructure, through layered architectures. Therefore, one of the very first definitions should be about the Product/Service delivered, the organization's output.

The Product/Service provided ought to be aligned with strategic orientations and integrated with defined goals. In turn, strategy is influenced by the services we want to offer to users, whose opinion is developed by what we deliver [6]. An effective service orientation is about providing what users need. Nevertheless, IT only has value to the business if it delivers the expected services. Therefore, we linked all activities with business objectives from Product/Service. Due to strategy definition, the Product/Services can be described as the focal points of business activities, shaped according to strategic requirements since users are only able to understand and pay for what we deliver them (the users' view of the service). Therefore, services have to be defined and measured from the users' perspective. From a business point of view, their identification is even a prerequisite to clarify the services critical goal. A service definition should promote a direct strategic effect, which can be measured [35]. From a technical point of view, services are translated into basic services, with elementary functionality [35], in IT services. An IT service identifies what is required to support a service, so it is not essential to know users' needs in detail, because they are already translated into Product/Services. From a different perspective, we clarify what IT services are needed to provide the defined services. However, it is crucial to determine how these directly affect the performance of services.

In order to be clear, we will determine which activities should be developed to support IT services, namely, the activity sequence—our processes. By activity identification, we will determine the provider involved, applications used, information CRUD, and supporting technological infrastructure. The next step is to integrate all identified concepts, defining the correspondent architectures. We used a layered approach identifying and linking architectures and elements. Figure 1 illustrates the conceptual map of integration between concepts from which we define and establish the relation between elements and architectures.

We continued the work, establishing the relationships as we went along, and stored the data into our footprints, providing different views as visualization of the relationships.

Each concept provides services to the related concepts with SOA elements, namely, service description, service policy, and service interface. We did not represent all elements in Fig. 1 to improve understanding.

The service-level agreement (SLA) is established by the strategy and defines the services, which are translated into the service catalogue.

The defined correlation among elements and architectures is established through services using SOA principles, allowing the implementation of ITIL processes, and mapping the integration between the two frameworks (ITIL and EA). The ITIL service life cycle concept, described in ITIL books, is developed as follows: Service Strategy addresses where, why, and what services should be done; Service Design defines how to meet strategic definitions, translating Product/Services into services; Service Transition is connected to the services deployment into operations and related processes; operational day-to-day activities are treated by the Service

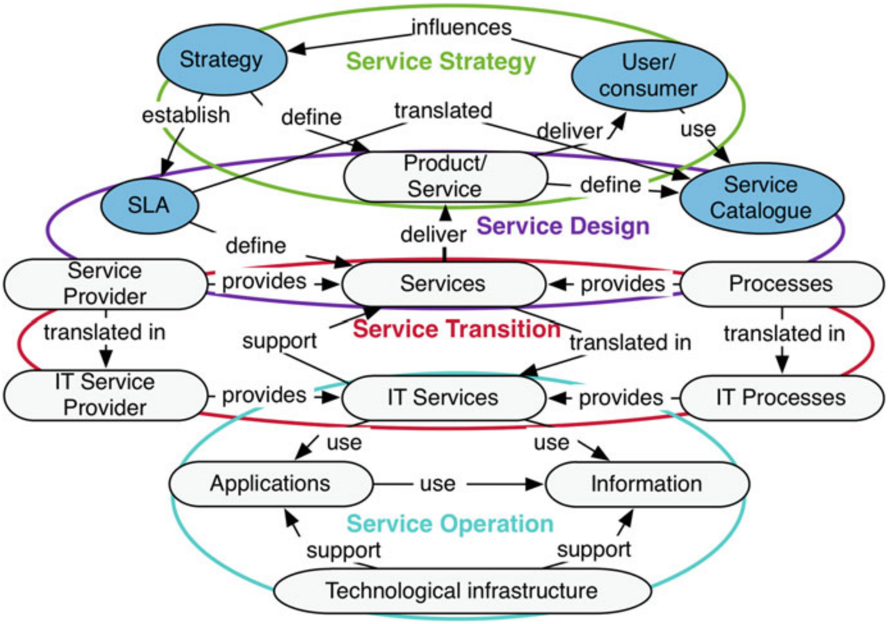


Fig. 1 Conceptual map of integration between concepts

Operation; and Continual Service Improvement permanently addresses the gap analysis between current and desired states.

Therefore, having EA as the basis for organization's representation, through different and independent architectures, we can map the service life cycle with architecture relationships. The entry point of both approaches is the Product/Service definition, the pivotal point of integration between the ITIL and the EA approaches. From the conceptual map, clarified by the concepts, relationship, and integration between both approaches, we are able to define our meta-model. In Fig. 2 we present the simplified meta-model of our proposal.

6 Demonstration

To demonstrate and validate our proposal, we decided to solve an instance of the problem [16] and applied our proposal to an IT department of a Portuguese public organization where there were two different functional divisions with responsibilities related to EA and ITIL. Each team has a clear idea of its functions, developed tools and procedures to help them separately. Moreover, knowledge is rarely shared and changes are barely discussed involving the two teams. As a result, there were constraints due to some subjects' overlap, which led to some competition between teams. This causes duplication of resources translated into wasted of time and money.

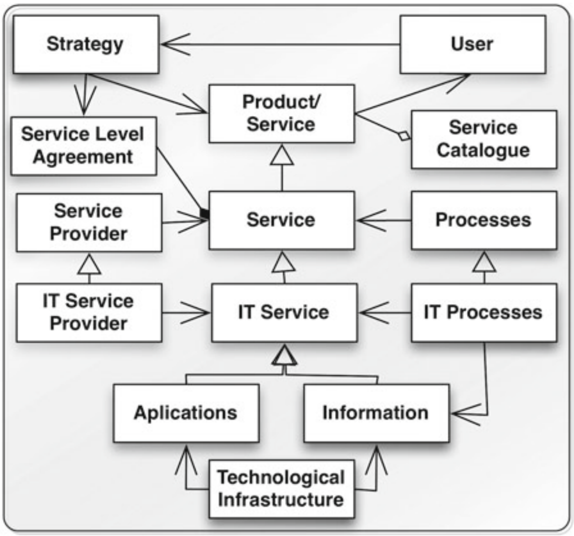


Fig. 2 Simplified meta-model

Despite well-known problems and some initiatives related with the sharing of knowledge and work, the issues remain and it is still difficult to represent and agree to a shared knowledge. To better the service level, the organization must improve the alignment and coordination between teams and, consequently, the integration of the works developed. Our proposal aims at this integration, i.e., to overcome the abovementioned problems.

We started by identifying the research motivation by sharing the proposal with all personnel, edifying the issues we wanted to address and the expected benefits, involving and motivating people. We identified the main concepts in accordance with our proposal and populated a single and shared common repository tool to support all concepts and their relations. A service orientation provided a single guidance for both functional divisions, diminishing misalignments and maintaining up-to-date information about the organizational "as-is." The results were not only very good, but also better and faster than expected.

7 Evaluation

The crucial integration point between both approaches, EA and ITIL, is the services delivered, which is the reason for an organization's existence. This common focus enables an integrated approach, maintaining the EA paradigm aligned with ITIL principles based on SOA. This answered our research question number one: "Which is the pivotal point between the two approaches?" We defined the pivotal point as the services delivered that are based on SOA.

The integration encompassing the relation between EA and ITIL requires a shared and single repository for the alignment between service management and EA artifacts. Otherwise IT is a collection of artifacts to meet technical requirements. The single repository answered our research question number two: “How to merge both approaches?” Thus, both approaches are merged, keeping the EAs, ITIL’s service life cycle, and the meta-model for services. The third question research “How to keep the merged information up-to-date?” is answered by the need to share information among functional divisions, using ITIL processes to keep information updated, through IT service management support processes.

ITIL principles and processes guarantee the update and consistency of information with standard processes like Configuration Management and Change Management, respectively, which ensure the reliability of the data that were recorded and accessed in the common repository and allow us to see the effects of the changes. The data repository was no longer a mere database with CI and their relationships, nor an architectural artifacts map of the “as-is” organizational state. Instead, in this approach, it encompasses all EA principles in an operational way.

8 Conclusion

This paper provides a research over the integration between Enterprise Architecture and IT service management. Being considered the two most important approaches in IT governance domain, we conducted an overview of the research made relating these two frameworks. During the course of our research, it became clear that the integration between ITIL and EA is a subject at its very beginning. We found a small number of studies and none solved the problem in a satisfactory way. Considering that both frameworks are complementary, but no integration proposal answered our research questions, we developed a new integration conceptual proposal based on services through a SOA paradigm.

Having answered our research question by providing a solution to the secondary questions, we may conclude that our proposal was verified, making a contribution to fill in the lack of research in Enterprise Architecture and IT service management integration. As shown in this paper, a relation between ITIL and an EA can be set, providing strong arguments for the existence of only one repository and avoiding duplication of efforts. The consolidation of both approaches in the same platform is imperative to achieve an improved alignment, while reducing the effort that would result from managing similar information in two distinct repositories.

We believe that our research is a contribution to organizational engineering, because it provides a proposal to integrate EA and ITIL initiatives. Therefore, we hope this study will be a key addition to academic efforts by bridging service management through architecture paradigm.

Another area of future research is the identification, relationship, and modulation of processes linked to services. Besides the existence of modulation standards (e.g., BPMN) to design business processes, the accuracy and subjectivity of modulation depend on the level of the people’s organizational maturity, which should be avoided with a modulation maturity-independent framework.

References

1. Spewak S, Hill S (eds) (1992) Enterprise architecture planning: developing a blueprint for data, applications and technology. Wiley-QED, New York
2. Zachman J (1987) A framework for information systems architecture. IBM Syst J 26:276–292
3. Gama N, Silva R, Mira da Silva M (2011) Using people-CMM for diminishing resistance to ITIL. Int J Human Capital Information Technol Prof (IJHCITP) 2:29–43
4. Brenner M (2006) Classifying ITIL processes; a taxonomy under tool support aspects. Business-driven IT management, Vancouver, Canada, pp 19–28
5. Gama N, Mira da Silva M, Francisco RA (2011) A conceptual framework for structuring an IT organisation. 16th Annual conference of the UKAIS, Oxford, England
6. Hochstein A, Zarnekow R, Brenner W (2005) ITIL as common practice reference model for IT service management: formal assessment and implications for practice. International conference EEE. IEEE Computer Society, pp 704–710
7. Correia A, Abreu FB e (2009) Integrating IT service management within the enterprise architecture. 4th International conference on software engineering advances (ICSEA), Porto, Portugal, pp 553–558
8. Johnson MW, Hately A, Miller BA, Orr R (2007) Evolving standards for IT service management. IBM Syst J 46:583–597
9. Sharifi M, Ayat M, Rahman AA, Sahibudin S (2008) Lessons learned in ITIL implementation failure. International symposium on information technology (ITSim 2008), vol. 2, Kuala Lumpur, Malaysia, pp 1–4
10. Braun C, Winter R (2007) Integration of IT service management into enterprise architecture. In: ACM (ed) ACM symposium on applied computing, New York, pp 1215–1219
11. Nabiollahi A, Alias RA, Sahibuddin S (2010) A service based framework for integration of ITIL V3 and enterprise architecture. 2010 International symposium in information technology (ITSim), vol. 1, Kuala Lumpur, pp 1–5
12. Thorn S (2007) TOGAF and ITIL. In: The Open Group (ed), vol. Catalog number W071, San Francisco, CA, p 26
13. Hevner AR, March ST, Park J, Ram S (2004) Design science in information systems research. MIS Quarterly 28:75–105
14. Oates BJ (2006) Researching information systems and computing. Sage, Beverly Hills, CA
15. March ST, Smith GF (1995) Design and natural science research on information technology. Decis Support Syst 15:251–266
16. Peffers K, Tuunanen T, Rothenberger M, Chatterjee S (2007) A design science research methodology for information systems research. J Manage Inform Syst(24)
17. The Open Group (2011) The Open Group architecture framework (TOGAF). Version 9, http://www.opengroup.org/togaf/
18. Lankhorst M (2009) Enterprise architecture at work. Springer, New York
19. Ross JW, Weill P, Robertson DC (eds) (2006) Enterprise architecture as strategy: creating a foundation for business execution. Harvard Business School Press, Boston, MA
20. Schekkerman J (2006) The economic benefits of enterprise architecture: how to quantify and manage the economic value of enterprise architecture. Trafford, Victoria, BC
21. Winter R, Fischer R (2007) Essential layers, artifacts, and dependencies of enterprise architecture. J Enterprise Architecture 3:7–18
22. Pereira C, Sousa P (2004) A method to define an enterprise architecture using the Zachman framework. In: ACM (ed) The 19th annual ACM symposium on applied computing (SAC'04), Nicosia, Cyprus, pp 1366–1371
23. Schekkerman J (ed) (2004) How to survive in the jungle of enterprise architecture frameworks. Trafford, Victoria, BC
24. Taylor S, Lloyd V, Rudd C (eds) (2007) ITIL: service design. TSO, Norwich
25. Wout JVW, Waage M, Hartman H, Stahlecker M, Hofman A (2010) The integrated architecture framework explained: why, what, how. Springer, New York

26. Tsai W-T, Chen Y, Fan C (2006) PESOI: process embedded service-oriented architecture. J Software 17:1470–1484
27. Chen H (2008) Towards service engineering: service orientation and business-IT alignment. 41st Hawaii international conference on system sciences, IEEE Computer Society, Waikoloa, HI p 114
28. Alwadain A, Korthaus A, Fielt E, Rosemann M (2010) Integrating SOA into an enterprise architecture: a comparative analysis of alternative approaches. 5th IFIP CONFENIS, Natal, Brazil, pp 1–15
29. Haki MK, Forte MW (2010) service-oriented business-IT alignment: a SOA governance model. AISS Adv Inform Sci Service Sci 2:51–60
30. Noran O, Bernus P (2008) Service oriented architecture vs. enterprise architecture: competition or synergy? Lect Notes Comput Sc 5333:304–312
31. Alwadain A Erwin F, Korthaus A, Rosemann M (2011) Where do we find services in enterprise architectures? A comparative approach. ACIS, Sydney, Australia
32. Shanks G, Tansley E, Weber R (2003) Using ontology to validate conceptual models. Commun ACM 46:85–89
33. Novak JD, Cañas AJ (2008) The theory underlying concept maps and how to construct and use them. Technical Report, Florida Institute for Human and Machine Cognition (IHMC)
34. Mintzes JJ, Wandersee JH, Novak JD (2000) Assessing science understanding: a human constructivist view. Elsevier Academic Press, San Diego, CA
35. Kieninger A, Baltadzhiev D, Schmitz B, Satzger G (2011) Towards service level engineering for IT services: defining IT services from a line of business perspective. Annual SRII global conference (SRII), 2011, San Jose, CA, pp 759–766

Software Process Management: A Model-Based Approach

L. García-Borgoñon, J.A. García-García, M. Alba, and M.J. Escalona

Abstract Business processes constitute one major asset in an organization and software businesses are not an exception. Processes definition, maintenance, and management are key aspects to control and define how to build software systems up and also to support decision-making. In this paper, a model-based approach is proposed to facilitate these processes. Thus, a global environment for business processes in software development is presented. The final results are illustrated through the NDTQ-Framework, a solution based on this approach that is currently being used in software development organizations.

1 Introduction

Since processes are recognized as fundamental asset in organizations, there is always an evolving interest to define, document, manage, and improve them. The promise of achieving better quality and greater efficiency and effectiveness in the cost and effort resulting from product development has involved the adoption of processes in several domains, some of which have reached a certain maturity level in this field. However, this is not the case of Software and System Processes, which is still in its early usage days.

In the last years software organizations are using Business Process Management (BPM) as a mechanism to control and define how to build software systems up.

L. García-Borgoñon (✉)
IWT2 Research Group, University of Seville, Seville, Spain

Instituto Tecnológico de Aragón, Zaragoza, Spain
e-mail: laurag@ita.es

J.A. García-García • M. Alba • M.J. Escalona
IWT2 Research Group, University of Seville, Seville, Spain
e-mail: julian.garcia@iwt2.org; manuel.alba@iwt2.org; mjescalona@us.es

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_13,

Different techniques for business process modeling and business process execution, as well as for their relationship, have been proposed. Business Process Modeling Notation [23] has become a widely used standard in process modeling environment, and Business Process Execution Language (BPEL) [24] has consolidated as the language for business process execution.

Software Process Engineering has been established as an independent research area from general business process. Its main objective deals with improving software development practices by proposing (a) better ways for designing the organization processes and (b) better ways for improving these processes both individually and as a whole.

In order to efficiently define and execute software development processes, it should be necessary to establish (1) a process modeling language rich enough so as to define all main aspects of software processes, (2) an easy-to-use process modeling environment that is flexible enough for different project categories, and (3) a process execution environment easy to be integrated into existing development tool chains [25].

The evolution of software process modeling has been studied during the past decades. A myriad of process languages and models have been developed; however, software process models are not executable and there are few process execution environments.

In the last years, the model-driven engineering (MDE) [26] has been established as a usual approach for software development [27], what has shaped the software industry to be model centric. Its aim is development through the evolution of one model from requirement until deployment by means of a series of transformations, which can progressively be achieved through coherence between software process modeling and software development paradigm [28].

This paper evaluates how a model-based approach can make easier the BPM in information system development organizations, and it also illustrates a practical example that uses this approach for software process definition, maintenance, and improvement.

The paper is structured as follows: Section 2 shows the main work related to software processes. Section 3 introduces the proposed metamodel, and Sect. 4 presents the Unified Modeling Language (UML) profile, which is used to integrate the metamodel into a tool. Section 5 analyzes a global environment for software process management. These results are illustrated in the NDTQ-Framework, a solution based on this approach that is currently being used in software development organizations. Finally, Sect. 6 outlines conclusions from these studies as well as proposes future work in these lines of research.

2 Related Works

The first section of this related work refers to different proposals that are referenced in the organizational processes definition as guidelines.

A process is defined as the set of partially ordered steps or activities, with sets of related artifacts, human and computerized resources, and organizational structures and constraints intended to produce and maintain the requested software deliverables [1]. Support processes have been developed in order to facilitate software organization activities, different standards, methodologies, and methods focused on management, development, evaluation, and software life cycle and organizational life cycle. At this point, it is necessary to highlight International Organization for Standardization (ISO) standards that prescribe processes, each of them with a specific aim:

- ISO/IEC 122007:2008 [2] establishes a common framework for software life cycle of processes with well-defined terminology that can be referenced by the software industry.
- ISO/IEC 15288:2008 [3] establishes a common framework for describing human-created life cycle of systems. It determines a set of processes and associated terminology that can be applied at any level in the hierarchy of a system's structure.
- ISO/IEC TR 24744:2007 [4] was defined through the large number of standards with similar concepts used for describing process reference models whose process descriptions vary in format, content, and level of prescription. Uniform descriptions combine processes from different reference models, facilitating the development and comparison of new models.

The second section of this related work presents the most popular languages and notations to process definitions.

The usual comparison between software processes and manufactured processes has entailed many efforts to describe and automate them. Thus, these efforts have been addressed in different stages. First-generation languages, known as Software Process Modeling Languages (SPMLs), were developed during the 1990s. Some of them were rule based such as MARVEL [5]; others were Petri net-based such as SPADE [6], or some others programming language-based such as SPELL or APPL/A [7]. All of them were focused on executability, but their complexity and emphasis on formality and inflexibility have made them not to succeed in the industry.

An alternative is BPMN, since it still remains as the preferred technology in the industry. Its simplicity, standardization, and support for executing processes are the key for being widely used. However, this language is more oriented towards business processes description, which constitutes a less specific scope than this of software processes.

As a result, many UML-based approaches were developed and a new language generation for software processes was introduced. Some of them were UML 1.3 based such as in Di Nitto et al. [8]; another uses a subset of UML 1.4 such as Chou's approach [9]. In addition, UML4SPM [10] was proposed as a candidate for the new version of the Object Management Group's (OMG) standard called Software Process Engineering Metamodel (SPEM) [11]; nevertheless it is based on SPEM 1.1 and UML 2.0 behavior modeling concepts. It mainly focuses on the enactment support and two alternatives were defined.

From the standardization point of view and regarding the software-specific domain, there are two main languages today: ISO/IEC 24744 [12] and SPEM 2.0. As it can be noticed, both pursue the same objective despite they differ in some aspects:

- ISO/IEC 24744, Software Engineering Metamodel for Development Methodologies, is an international standard that defines a metamodel for methodologies of development in the software environment. It does not use OMG's strict metamodeling approach, but the power-type pattern that was adopted for metamodeling in the methodologies domain in [13].
- SPEM 2.0 provides a language for software methodologies, takes the Meta-Object Facility (MOF) [14] as a starting point, and is defined as a UML profile. It has a very difficult structure since it introduces extension mechanisms, compliance points, and concepts to distinguish method contents from processes, what make the specification turns out very complex and difficult to understand.

In recent years, model-based engineering has been established as a standard approach for software development. MODAL (Model-Oriented Development Application Language) [15] is a SPEM 2.0-based process modeling language that introduces additional concepts to exploit the potential of MDE. Unlike SPEM, it is more focused on process model execution, even though the standard complexity is reproduced here.

To summarize, efforts in Software Process Engineering area have been headed by two different aspects: on the one hand, methods and standards definition in order to prescribe what a process should accomplish and, on the other hand, the need to have a language for the process definition. Last decades have witnessed the birth of many approaches and a parallelism between software development paradigms and processes for their development has always been evidenced. As a result, proposing MDE for process engineering may be a solution to cope with this classic problem.

3 A Metamodel for Software Process Management

Many approaches have been developed in order to recommend the required elements in a process as well as describing it. The main element in all of them is Software Process, but the concepts included in these approaches differ. This situation has motivated the development of the standard ISO/IEC TR 24744, which is issued as a guideline for the process description.

These and the fact of using MDE to possibly manage the conceptual complexity of Software Process Engineering have been the basis of our proposal, that is, a metamodel for Software Process Support. This approach is presented in the form of a MOF-compliant metamodel as it is shown in Fig. 1.

The *Process* metaclass is the main class in the metamodel. It represents a set of ordered actions executed by someone in order to produce something. The attributes in this metaclass, the name of the process and a short description of it, have been incorporated in accordance with ISO/IEC TR 24744 standard.

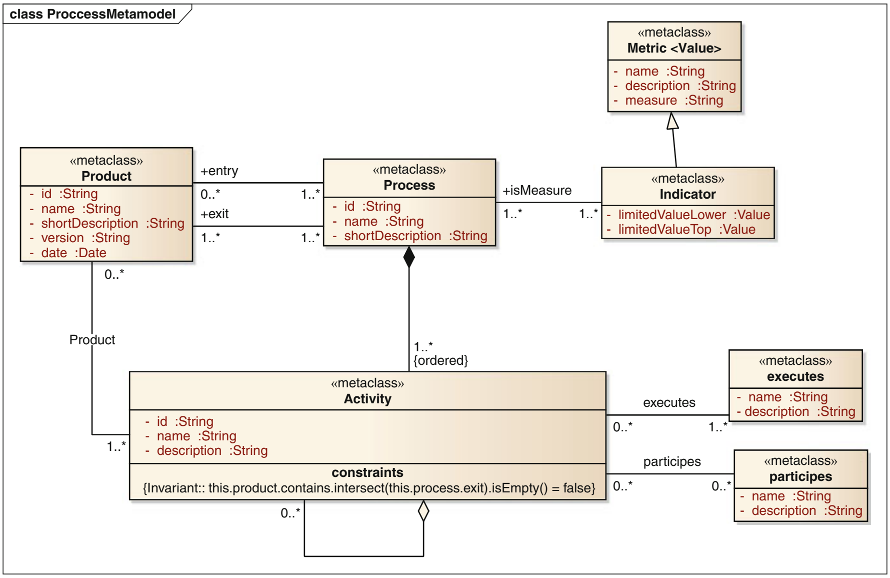


Fig. 1 Process metamodel

The actions included in a process are represented by the *Activity* metaclass. These activities are arranged within the process, although one activity can also contain several activities. This recursive relation is explicitly requested by the standard.

There are two kinds of stakeholders involved in performing an activity, which are classified depending on the involvement degree: an actor, who is the main executor of the activity, is represented by the *Executes* metaclass, whereas the *Participates* metaclass represents the set of stakeholders who contribute, but is not directly the main responsible for it. Each of them includes a name and a description attributes. At least one executor is necessary to define an activity.

The *Product* metaclass represents the product resulting when executing a process. This product can be developed from scratch, during the execution of the activities, or can be provided from a previous product. In this case, the original product will be defined as an entry and modified within the activity in order to obtain the outcome.

Finally, the *Metric* metaclass deals with process information elements. Metrics in processes are limited-value established and represented through the *Indicator* metaclass.

The main feature of our approach is simplicity, unlike others presented in the previous section. This metamodel offers a suitable mechanism for process definition by covering the main software process concepts and being an ISO/IEC TR 24744 compliant. This simplicity must develop a whole model-based solution, as it will be studied in next sections.

4 An Enterprise Architecture Profile

A profile is defined in a UML package through the stereotype <<profile>>, indicating that it will extend a metamodel. There are three mechanisms used to define these profiles:

- Stereotypes. They are defined by a name and a series of elements of the metamodel with which they can be associated. Graphically, the stereotypes are set in boxes, <<stereotype>>.
- Restrictions. They impose conditions on the previously stereotyped elements of the metamodel, so as to describe, among others, the conditions that they have to check in a "well-built" model. A commonly used language of restriction is OCL.[1]

  e.j: Invariant:: this.product.contains.intersect(this.process.exit).isEmpty()=false

- Tagged Values. There are additional meta-attributes that can be associated with the metamodel of a metaclass extended through a profile.

To build the profile, the software process metamodel should be used as shown in Fig. 1. A stereotype is included within the package <<profile>> for each element of the metamodel contained in the profile. It is named as the metamodel elements; thus a relationship between the metamodel and the profile is established. Then, any item needed to define the metamodel can be labeled with a stereotype [29]. In this case "Participies," "Executes," "Activity," "Product," "Process," "Indicator," and "Metric" will be created.

In our example, Participies and Executes extend of the metaclass "Actor;" Activity and Process extend of the metaclass "Activity;" and Product, Metric, and Indicator will extend of Artifact. Tagged values are defined as the profile attributes elements that appear in the metamodel. The definition of their types and possible initial values must be included.

In the given example, the attribute "measure" of the metaclass Metric and the attributes LimitedValueTop and LimitedValueLower of the metaclass Indicator have to be added as tagged values since the metaclass implicitly has other attributes. We will define the profile constraints from the domain constraints. For example, either the multiple associations listed in the domain metamodel or any individual's business rules of the application must be translated when the relevant restrictions are defined. In this case, there is one the invariant of Activity (Fig. 2).

Invariant:: this.product.contains.intersect(this.process.exit).isEmpty()=false

5 An Example: NDTQ-Framework

NDTQ-Framework is based on the metamodel analyzed in Sect. 3. It formally defines all processes currently supported by NDT [16], although it is also flexible and can be adapted to different levels and typologies of developments.

---

[1] http://www.omg.org/technology/documents/modeling_spec_catalog.htm#OCL.

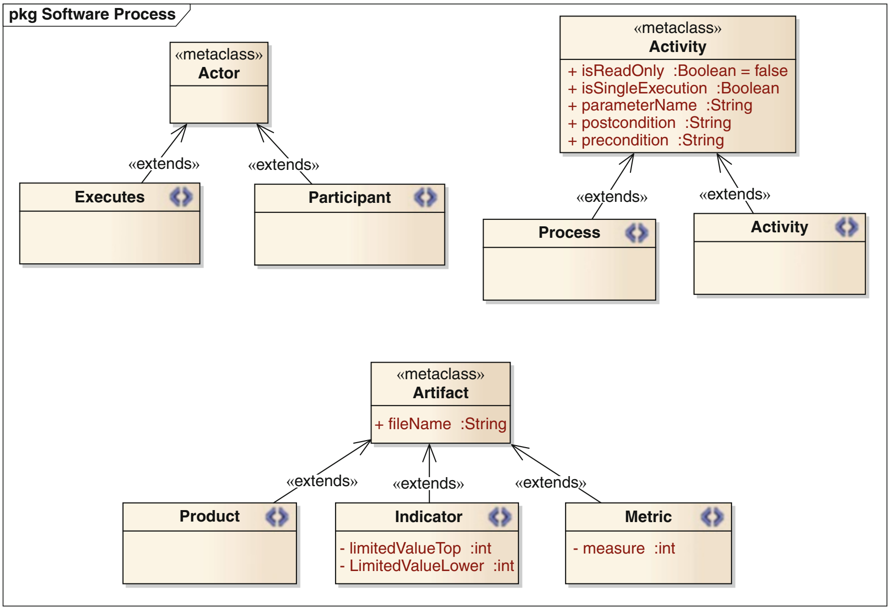


Fig. 2 Software process profile

NDTQ-Framework defines processes on the Enterprise Architect tool through a specific pattern dealing with process description, based on the proposed ISO/IEC TR 24774:2007 and ISO12207. It thoroughly defines the six groups of processes that include NDT-methodology. However, due to their extension, they will not be fully explained in this paper, but briefly introduced below. Only the requirements engineering process, which belongs to the software development processes, will be pointed out.

- Processes of Software Development. These processes support different types of life cycles: classical or sequential, agile, iterative, and incremental. They are defined on the basis of NDT life cycle, although the terminology has been referenced on ISO 12207 standard.
- Processes of Software Maintenance. These processes are based on the best practices defined both in ITIL [17] and CMMI [18]. This group only defines the process beginning when the project is in production and ending when the system falls into disuse.
- Processes of Testing. These processes are based on the first results of ISO/IEC 29119 [19]. This group defines Testing Organization, Testing Management, and Testing Execution.
- Processes of Software Quality. This group of processes is based on ISO 9001:2008 and the good practices of CMMI. This group defines the following processes: Managing Corrective and Preventive Actions, Documentation, Control and Records, Human Resource Management, Customer's Satisfaction, Data Analysis and Review by the Director of the Organization, Technology Watch, Monitoring Indicators, Elaboration of Standards, and Internal Audits.

- Processes of Project Management. They are based on some of the practices of the Project Management Body of Knowledge (PMBOK) [20] and CMMI. This group defines the following processes: Event Management, Personnel Management, Project Monitoring, Change Management, Schedule Management, and Cost Management.
- Processes of Security. This process is based on ISO 27001. This group defines the following processes: Physical and Environmental Security, Asset Management, Risk Analysis, Security Organization, Communications Management and Operations, Security Incident Management, and Business Continuity Management.

The following information is provided by each of the processes mentioned above: roles or participants involved in its execution, indicators, tasks or activities, and deliverables of the process. All this information matches with the attributes included in the metamodel defined in Sect. 3. To illustrate all the processes above is out of the scope of this paper. However, in order to present the approach, the requirements engineering process will be explained in detail. Figure 3 shows the map of activities for the requirements engineering process.

The Requirements Engineering includes the necessary flow of activities to generate the system requirements document. NDT-Suite, which consists of a set of very useful free tools to apply the NDT-methodology, can automatically create and revise the requirement documents [21].

After finishing this process, all participants must reach a consensus and approve the final system requirements document. The requirements engineering process involves four roles or participants: Project Manager, Monitoring Committee, Project Manager at SQA, and Responsible for User's Area.

The Project Manager must carry out the first activity within the requirements engineering process. This activity is referenced in Fig. 3 as "RS01-Get information about the environment and define objectives." In this activity, the Project Manager must approach to the environment where the system will be implemented. During this activity, the terminology used in the project, users and customers who will participate, as well as the main objectives must be set.

In the next activity, the Monitoring Committee must approve the project scope and objectives previously established. This activity is referenced in Fig. 3 as "RS02-To approve the scope." Once the objectives have been identified, the system requirements must be captured. This task is carried out by the Project Manager and includes five activities: (1) identifying and defining storage requirements, (2) identifying and defining actors, (3) identifying and defining functional requirements, (4) identifying and defining interaction requirements, and (5) identifying and defining nonfunctional requirements.

After completing the system requirements document and before being reviewed by the user, the document must be validated by the Project Manager at SQA. This validation involves three other steps: automatic validation through NDT-Suite, which will generate the corresponding report; validation of the technician responsible of SQA; and validation of the technical coordinator.

The Responsible for User's Area, after identifying and describing requirements, must validate it. Audits, thesauri, or ontologies are techniques for requirements

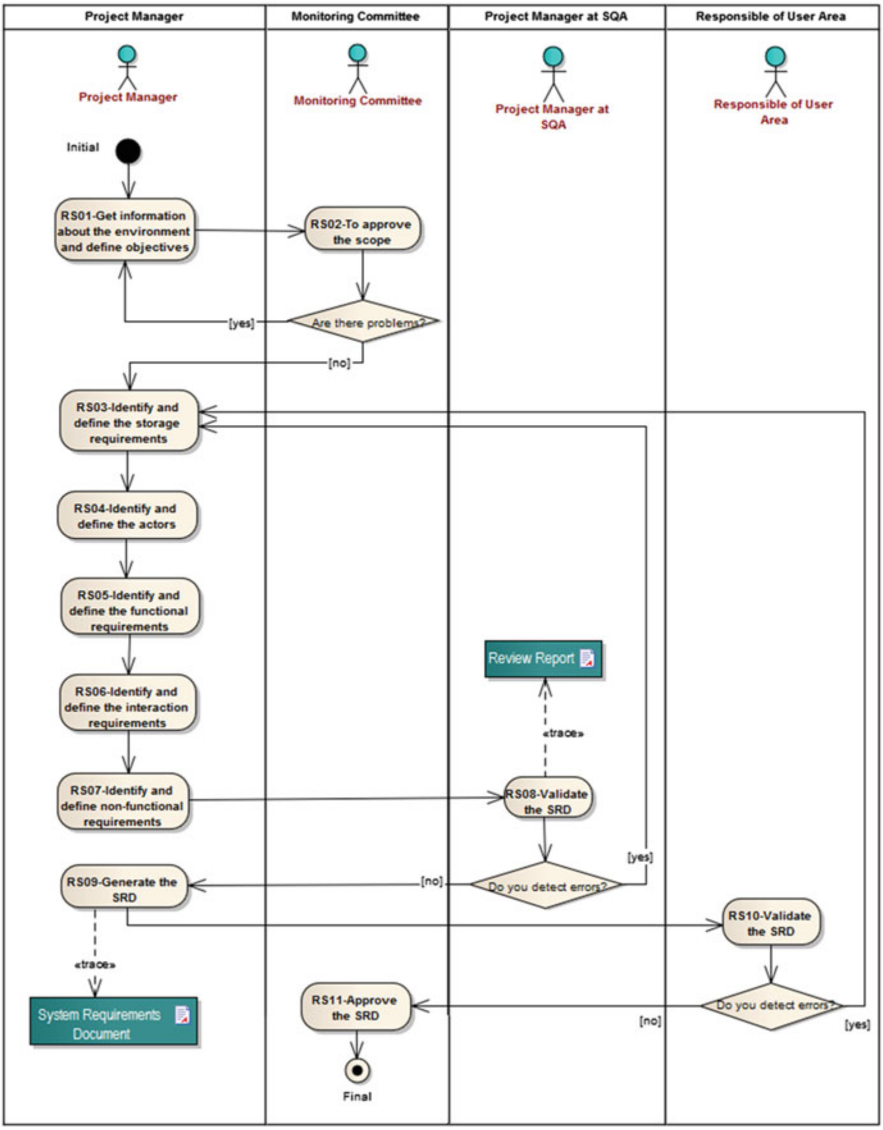


Fig. 3 Map of activities of the requirements engineering process

validation. The final aim of this task is to detect and correct as much errors found during the description of requirements as possible. To elaborate the glossary, NDT-Glossary [22] is recommended (this tool is included in NDT-Suite). Finally, if the Responsible for User's Area has not detected any error or inconsistency in the requirements identification, the Monitoring Committee approves the system requirements document.

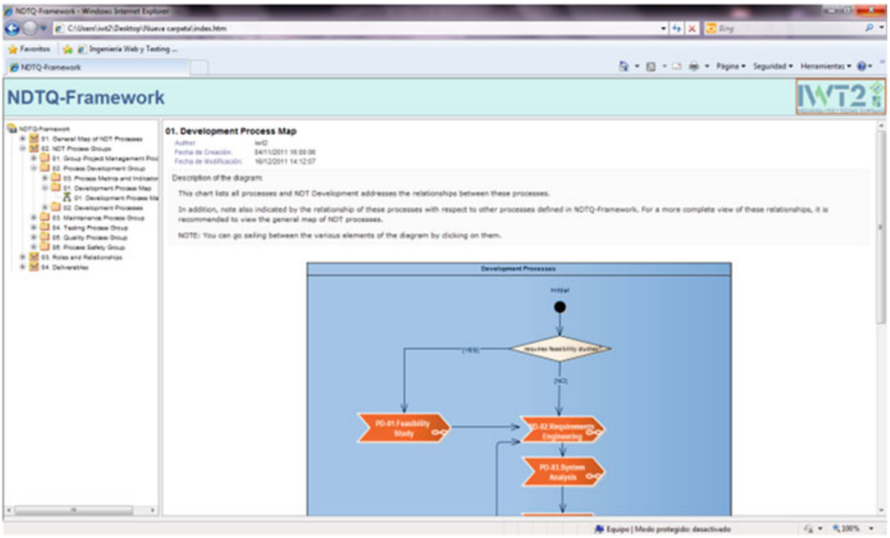


Fig. 4 NDTQ-Framework screenshot

As it has been discussed in this section, each process defined by the NDT-methodology needs to indicate certain relevant information such as the participants involved in its execution, indicators, tasks or activities, and deliverables of the process.

NDTQ-Framework allows quick and direct access to all information associated with an element within the NDT-framework. For instance, from the process, it can be accessed, among others, to all the information related to actors involved in its execution, its associated indicators, its deliverables, or its associated tasks or activities. Besides, NDTQ-Framework guarantees the traceability among these elements. It also offers a set of tools to orchestrate all NDT processes. Today, we are currently working on this line of research (Fig. 4).

6 Conclusions and Future Works

This paper presents a solution for software process definition founded on a model-based approach according to ISO/IEC TR 24744, the standard guideline to establish the concepts related to software processes. This solution is offered by a metamodel and an UML profile and is implemented in Enterprise Architect. A concrete solution named NDTQ-Framework is also presented.

It has been used in several real projects where some relevant conclusions can be deduced. Firstly, a model-based mechanism to define software processes can be very useful, but, if concrete syntaxes and semantic to represent them are not provided, it fails to be used in companies. Otherwise, communication problems can arise.

UML profiles and a UML-based tool seem to be good options to represent them in the software process environment since the development team usually knows this notation.

A tool supposes a required and essential necessity to offer a solution for model process definition. In fact, defining a process under a metamodel guarantees uniformity and a correct definition according to the standard. However, if a suitable tool is not defined, the maintenance of these processes can result too complex, and inconsistencies between the defined process and the real process can arise.

One of the most important aspects concerning this process, which is widely recommended in many standards and good practices manuals, is the continuous improvement. To have a process map, a clear relationship between activities and mechanisms for metric and definition measurement are elements to be taken into account in a continuous improvement program. Consequently, to have a suitable mechanism for defining and maintaining becomes necessary, and it can only be obtained through a tool.

The solution proposed in this paper entails that, in NDTQ-Framework, processes are not only defined under the standard, but they are also connected and interrelated; thus the mechanism to maintain them improves. Besides, this interrelation and connection and the fact that it is based on UML notation reduce the learning curve.

As future work, this approach is aimed for improving in different ways. Firstly, we are working on extending our framework with new processes, like some fragments of ITIL or PMBOK. Secondly, this tool can be improved with a mechanism of orchestration oriented towards the idea of NDTQ-Framework as a whole solution in order to process definition, documentation, and maintenance.

Nevertheless, at this point, there is not support to processes execution, and companies usually have a manual mechanism to solve this situation. In this sense, NDT-Suite can offer a first step to support it, although it is not enough. The processes execution defined by our metamodel represents a very relevant line of research.

Additionally, getting metrics and indicators during the process execution poses another line of research. A solution may allow organizations to identify, extract, and analyze data to support decision-making.

Acknowledgements This research study has been supported by the Tempros project (TIN2010-20057-C03-02) and Red CaSA (TIN 2010-12312-E) of the Ministerio de Ciencia e Innovación, Spain, and NDTQ-Framework project of the Junta de Andalucía, Spain (TIC-5789).

References

1. Lonchamp J (1993) A structured conceptual and terminological framework for software process engineering. In: Proceedings of the 2nd international conference on the software process continuous software process improvement, pp 41–53
2. ISO/IEC, ISO/IEC 12207:2008 (2008) Systems and software engineering – software life cycle processes. International Organization for Standardization
3. ISO/IEC, ISO/IEC 15288:2008 (2008) Systems and software engineering – system life cycle processes. International Organization for Standardization

4. ISO/IEC, ISO/IEC TR 24744:2007 (2007) Software and systems engineering – life cycle management – guidelines for process description. International Organization for Standardization
5. Kaiser G, Barghuti N, Sokolsky M (1990) Preliminary experience with process modeling in the marvel SDE kernel. In: Proceedings IEEE 23th Hawaii ICSS software track
6. Bandinelli SC, Fuggetta A, Ghezzi C (1993) Software process model evolution in the SPADE environment. IEEE T Software Eng 19(12):1128–1144
7. Conradi R, Jaceheri M, Mazzi C, Nguyen M, Aarsten A (1992) Design, use and implementation of SPELL, a language for software process modeling and evolution. Software Process Technology, pp 167–177
8. Di Nitto E, Lavazza L, Schiavoni M, Tracanella E, Trombetta M (2002) Deriving executable process descriptions from UML. In: Proceedings of the 24th international conference on software engineering (ICSE 2002), Compendex, pp 155–165
9. Chou S-C (2002) A process modeling language consisting of high level UML-based diagrams and low Level process language. J Object Technol 1(4):137–163
10. Bendraou R, Gervais M-P, Blanc X (2006) UML4SPM: an executable software process modeling language providing high-level abstractions. Enterprise distributed object computing conference 2006 EDOC 06 10th IEEE International, vol. 6, no. 511731, pp 297–306
11. OMG (2008) SPEM, software & systems process engineering metamodel specification. http://www.omg.org/spec/SPEM
12. ISO/IEC, ISO/IEC 24744:2007 (2007) Software engineering – metamodel for development methodologies. International Organization for Standardization
13. Henderson-Sellers B, Gonzalez-Perez C (2005) The rationale of powertype-based metamodelling to underpin software development methodologies. In: Proceedings of the 2nd Asia-Pacific conference on conceptual modelling, vol. 43, pp 7–16
14. OMG (2011) MOF, meta object facility. http://www.omg.org/spec/MOF/2.4.1
15. Koudri A, Champeau J (2010) MODAL: a SPEM extension to improve co-design process models. In: Proceedings of the 2010 international conference on new modeling concepts for today's software processes: software process, vol. 6195, pp 248–259
16. Escalona MJ, Aragon G (2008) NDT. a model-driven approach for web requirements. IEEE T Software Eng 34(3):377–390
17. ITIL, Information technology infrastructure library. http://www.itil-officialsite.com
18. Chrissis MB, Konrad M, Shrum S (2003) CMMI: guidelines for process integration and product improvement. Addison Wesley, Reading, MA, p 688
19. ISO/IEC, ISO/IEC 29119 Software engineering – software testing standard. International Organization for Standardization
20. Project Management Institute (2008) A guide to the project management body of knowledge (PMBOK® guide) – fourth edition. Project Management Institute, Newtown Square, PA, p 459
21. (2011) NDT-Suite. www.iwt2.org
22. García-García JA, Cutilla CR, Escalona MJ, Alba M (2011) NDT-glossary. A MDE approach for glossary generation. In: Proceedings of the 13th international conference on enterprise information systems. ICCEIS
23. OMG (2011) BPMN, business process modeling notation, Version 2.0. http://www.omg.org/spec/BPMN/2.0/
24. OASIS (2007) WS-BPEL, Web services business process execution language, Version 2.0. http://www.oasis-open.org/standards#wsbpelv2.0
25. Ellner R, Al-Hilank S, Drexler J, Jung M, Kips D, Philippsen M (2010) eSPEM – a SPEM extension for enactable behavior modeling. In: Kühne T, Selic B, Gervais M-P, Terrier F (eds) EdsModelling foundations and applications, vol 6138. Springer, Berlin, pp 116–131
26. Ardagna D, Ghezzi C, Mirandola R (2008) Rethinking the use of models in software architecture. In: Becker S, Plasil F, Reussner R (eds) Quality of software architectures models and architectures, vol 5281. Springer, Heidelberg, pp 1–27
27. Schmidt DC (2006) Model-driven engineering. Computer 39(2):25–31
28. Van Der Straeten R, Mens T, Van Baelen S (2009) Challenges in model-driven software engineering. In: Models in software engineering. Lect Notes Comp Sci 5421:35–47
29. Fuentes L, Vallecillo A (2004) Una introducción a los perfiles UML. Novática 168:6–11

Improving NDT with Automatic Test Case Generation

J. Gutiérrez, M.J. Escalona, M. Mejías, F. Domínguez, and C.R. Cutilla

Abstract The model-driven development defines the software development process as a set of iterations to create models and a set of transformations to obtain new models. From this point of view, this paper presents the enhancement of a model-driven approach, called navigational development techniques (NDT), by means of new models and transformations in order to generate test cases. It also states some conclusions from the research work and practical cases in which this approach was used.

1 Introduction

Model-driven development (MDD hereinafter) is a software engineering paradigm focused on creating and exploiting domain models [1]. Navigational development techniques (NDT) [2] is a development framework which follows MDD. Therefore, NDT describes the full software development cycle indicating which models generate in each phase and how to derive the models of a phase from those of a previous phase. Projects developed with NDT start with a goal-oriented phase of requirements and apply use cases for defining requirements and transformations so as to generate the following models.

NDT was initially defined to deal with Web development requirements, but it has evolved in the last years and nowadays it offers a complete support for the complete life cycle. NDT covers viability study, requirements treatment, analysis, design, construction, or implementation as well as maintenance and test processes. Additionally, it supports a set of processes to bear out project management and quality assurance.

J. Gutiérrez (✉) • M.J. Escalona • M. Mejías • F. Domínguez • C.R. Cutilla
IWT2 Research Group, University of Seville, Seville, Spain
e-mail: javierj@us.es; mjescalona@us.es; risoto@us.es;
fjdominguez@us.es; carmen.ruiz@iwt2.org

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_14,

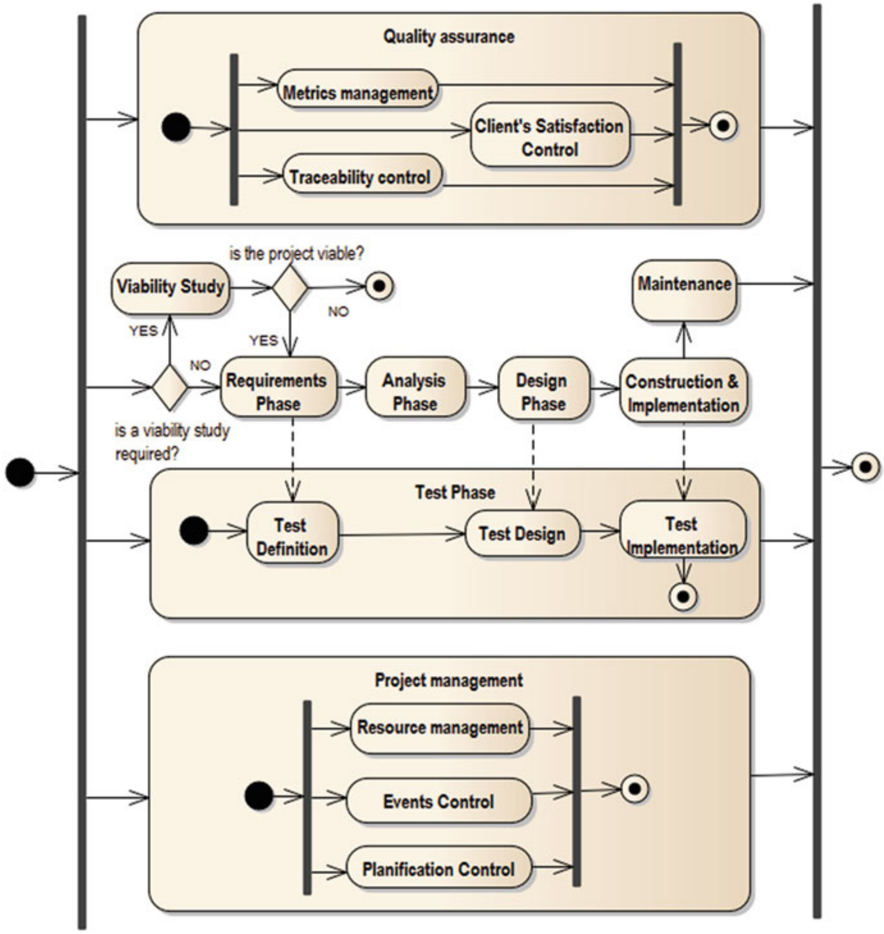


Fig. 1 NDT development process overview

Figure 1 depicts the development process as defined in NDT. This development process is independent from the development life cycle, so any life cycles (e.g., cascade or iterative) may be applied. Testing is always a mandatory task (Fig. 1) regardless of the type of project. Thus, there is a need of incorporating techniques in order to define test cases. Moreover, due to the MDD nature of NDT, these techniques should be described in terms of the proper elements of MDD, mainly models and transformations from models to models.

This paper describes how NDT has been extended to incorporate functional system test cases [3]. These test cases verify that the system under test commits the behavior defined in its functional requirement. NDT models the functional requirements as use cases; thus, both terms will be used as synonyms in this paper.

This paper is organized as follows. After this introduction, Sect. 2 offers an overview of the existing techniques dealing with generating functional test cases from functional system requirements defined as use cases. Section 3 describes the techniques used to generate test cases. Section 4 summarizes the extension of NDT to incorporate those techniques from a MDD perspective. Section 5 presents practical applications for NDT enrichment with test case generation. Finally, Sect. 6 states the conclusions and ongoing work.

2 Related Work

There are several approaches generating functional test cases specifically from a functional requirements model defined as use cases by means of MDD. A survey about this issue, which updates the original survey published in [4], has been published in [5] at the end of 2011. Some specific approaches studied in Escalona's survey are described in next paragraphs.

Frölich et al. [6] introduce an approach describing how to translate a functional requirement from natural language into a state-chart diagram in a systematic way as well as how to generate a set of functional test cases from that diagram. Naresh [7] presents an approach dealing with translating a functional requirement from natural language into a flow diagram and performing a path coverage technique to generate test cases. Mogyorodi [8] introduces an approach describing functional requirements as cause-effect graphs which generates test cases from diagrams. Boddu et al. [9] present an approach divided into two blocks: the first one describes a natural language analyzer generating a state machine from functional requirements, and the second one shows how to create test cases from such state machine.

Ruder's [10] approach starts with functional requirements written in natural language. The result is a set of functional test cases obtained from a coverage criterion based on combinations that support Boolean propositions. Binder's book [11] describes the application of the category-partition method over use cases. The categories are any point in which the behavior of the use case may be different between two realizations of the use case. This application is named the extended use case pattern. Finally, Ibrahim et al. [12] offer a tool, called GenTCase, which generates test cases automatically from a use case diagram enriched with each use case tabular text description.

Escalona's survey claims that there is no definitive approach that closes the problem of generating functional text cases automatically in a satisfactory way, what implies a lack of evolution among the existing approaches. Thus, there are some aspects to be improved, like the use of standards for inputs and outputs, the application of standards and more formal methods to describe the process itself, the need for empirical results or the measure of the possible automation, and a profitable tool supporting, among others. Conclusions of Denger's survey go in the same line.

3 Techniques for Test Cases Generation

Two techniques have been identified for generating test cases from use cases, from the surveys cited in previous sections: round-strip strategy and extended use cases (terminology defined by Binder in [11]). Below, these techniques will be described in depth.

The round-strip strategy consists in the application of a classic algorithm of path finding over a state machine. The behavior described in a functional requirement may be managed as a graph or as state machine despite its concrete syntax. Hence, a path searching allows identifying all the different paths across the behavior. Each path will be a scenario designed together with the system. Each scenario is a potential test case for testing the right implementation of such scenario in the system under test. Generation of test cases from state machines is a widely described topic in research literature. Previous section presented several references about this topic in the specific use cases context, like [6, 7, 9]. Figure 2(a) shows an example of the round-strip strategy using the behavior of a use case defined as an activity diagram.

The extended use case pattern consists in applying the category-partition method [13] to use cases. The category-partition method is a technique based on identifying categories and partitions and then generating combinations among such partitions (Fig. 2b). In the context of functional requirements, a category is any point for which the functional requirement defines an alternative behavior (Fig. 2b). Besides, a partition is defined as a subset of the domain of the condition evaluated in the category which decides whether a concrete piece of behavior is executed or not. Once all categories and partitions are identified, a combination among them is performed and each combination becomes a potential test case. The previous section presented several references about this topic in the specific context of use cases, like [10] or [11]. Figure 2b shows an example of the category-partition method (as described in [11]) using the same behavior as Fig. 2a.

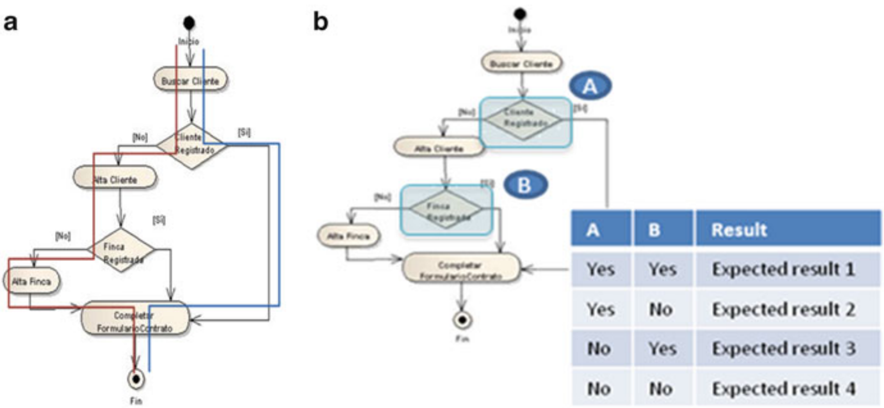


Fig. 2 Examples of round-strip strategy (**a**) and extended use cases (**b**) techniques

4 Extension of NDT

This section describes the work carried out to extend NDT after including the two techniques presented in the previous section. Section 4.1 defines the information involved. Then, Sect. 4.2 defines how to apply both techniques to obtain the target test artifacts from the functional requirements. Finally, Sect. 4.3 offers an overview on the application of Sects. 4.1 and 4.2 results.

4.1 Concepts and Metamodels

Due to the model-driven nature of NDT, the concepts involved in generating functional test cases should be identified and defined as metamodels. A metamodel defines the concept in terms of its attributes and its relationships with other concepts [1].

Four metamodels were designed. The first one (Fig. 3) defines the necessary elements from functional requirements to generate test cases. These elements constitute a subset of functional requirements. Therefore, it only involves the elements used for test cases generation. This metamodel may be applied with other frameworks apart from NDT. The functional requirement metamodel (Fig. 3) includes classic elements of functional requirement defined as *use cases*, Step or Actor, among others, widely described in the literature.

The second metamodel (Fig. 4) defines the concepts resulting from the roundstrip technique (Fig. 2a). Each path is called test scenario (element *TestScenario* in Fig. 4) and the traverse/crossed steps are classified into actions (element *ActionFromTestScenario* in Fig. 4) when performed by an external actor or into verifications (element *VerificationFromTestScenario* in Fig. 4) when performed by the system and, therefore, is suitable to introduce an assert during the test.

The third metamodel (Fig. 5) defines the concepts resulting from the category-partition method. Categories are modeled using the element *OperationalVariable*

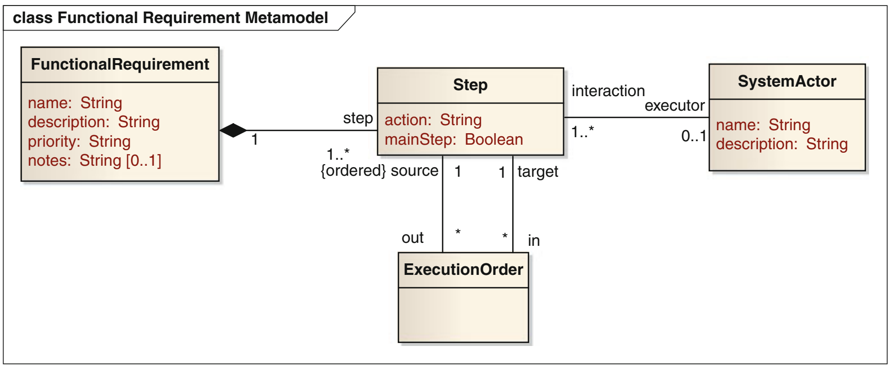


Fig. 3 Metamodel for functional requirements

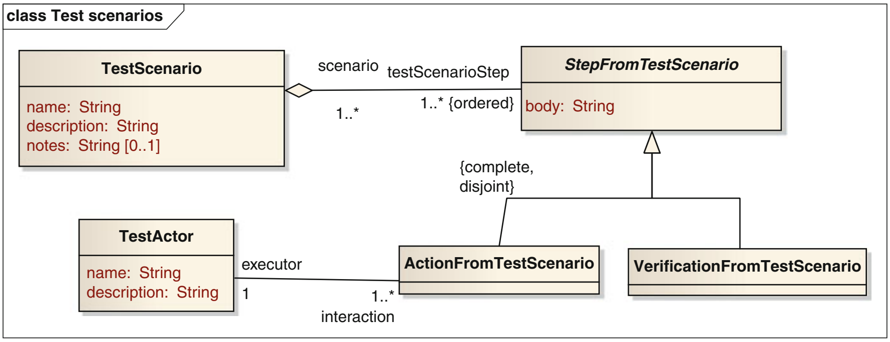


Fig. 4 Metamodel for test scenarios

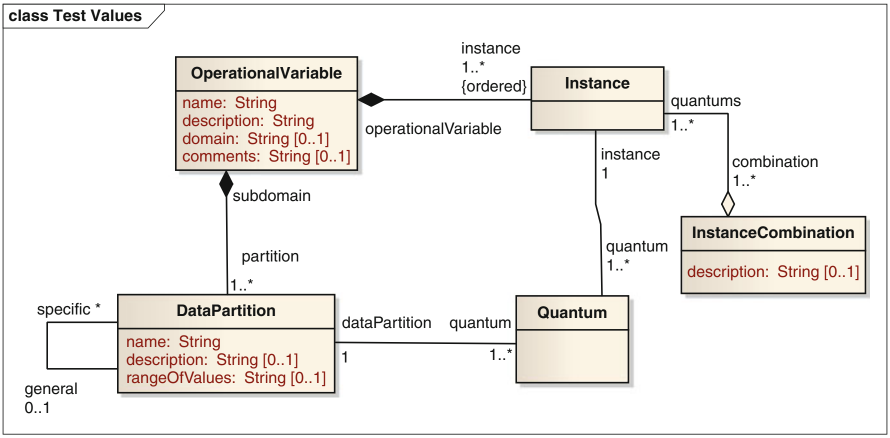


Fig. 5 Metamodel for test values

(as named in [11]), whereas partitions are modeled through the element *Partition*. The element *Instance* points out an evaluation of an operational variable, for example, A or B cells in Fig. 2b, and allows distinguishing it from other evaluations of the same operational variable, in case the behavior of the functional requirement has loops. A *Quantum* element models a value transfer from a partition to an instance. A combination (a row in Fig. 2b) is modeled using the element *Instancecombination*.

Finally, the last metamodel introduces artifacts that combine the results of the two previous techniques in the same model. This last metamodel does not introduce any new information. However, it offers glue elements to represent the information through a common artifact (called test case), the steps from a functional requirement as well as a combination of partitions. Figure 6 shows the tracing relation between the four metamodels. Tracing enables knowing which test artifacts have been generated for each functional requirement.

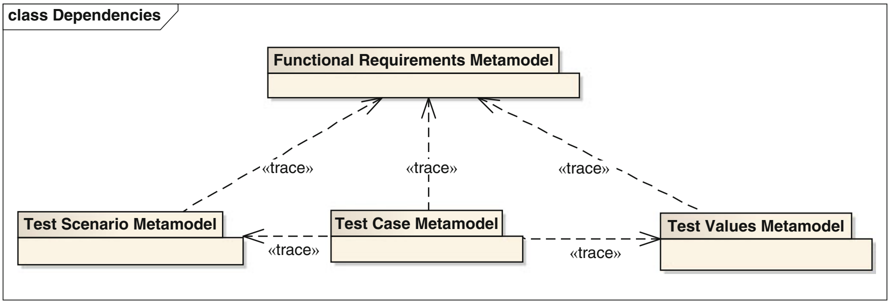


Fig. 6 Tracing relationships among metamodels

Some additional elements from the metamodels have been omitted. These elements introduce additional concepts like preconditions and packages. The four former metamodels have been added to the set of metamodels managed and supported by NDT as part of its MDD development process.

4.2 *Relations and Transformations*

Section 4.1 described the concepts involved in the improvement of NDT to generate functional system test cases. This section goes one step beyond and describes how to apply the two techniques presented in Sect. 2 (round-strip and extended use cases) using the information from the functional requirements metamodels (in the previous section) as source and the information from the testing metamodels as target.

The process of applying both techniques is defined according to the identification of a set of relations between source concepts (functional requirements) and target concepts (test scenarios and operational variables combinations), as observed in Fig. 7. The task of identifying these relations consists in detecting how to build up one target element, for example, a test case, by means of the source elements and their information. Next paragraphs provide an overview of the three relations (named T1, T2, and T3 in Fig. 7) defined to create test scenarios, combinations of operational variables, and test cases from functional requirements.

Relation T1 involves functional requirements and the round-strip strategy. As it was represented in Fig. 2a, the functional requirement behavior may be modeled as a state machine; the concept *Step* from Fig. 3 models the states; and the concept *Execution Order* models the transitions. Thus, a classic coverage criterion may be selected to traverse/cross the functional requirement and generate test scenarios. The all-loops criterion, in which all combinations among loops are traversed at least once, is the one selected to extend NDT. Test scenarios steps are generated from all the functional requirements steps. Action (element *ActionFromTestScenario*) and verification (element *VerificationFromTestScenario*) classifications depend on whether there is a relation with a system actor. Finally, test actors are generated from actors, which, due to their attributes, are the same ones.

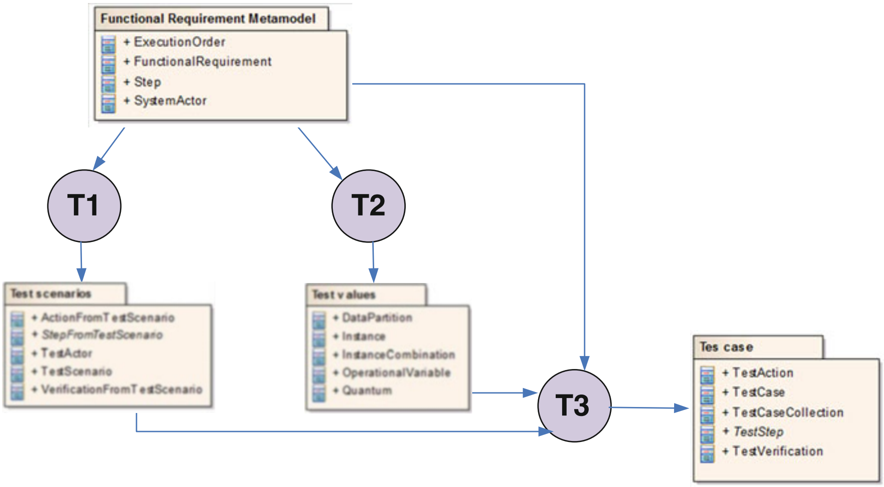


Fig. 7 Transformations among models

Relation T2 in Fig. 7 involves functional requirements and the category-partition method. Operational variables are created from steps that have more than one output transition (modeled as an *ExecutionOrder* element). The outputs of the steps generate the different partition. Again, combinations may be calculated using several criteria, from calculating all possible solutions or calculate just a subset.

Relation T3 (Fig. 7) combines both techniques results. Test scenarios and combinations of operational variables merge using test cases.

4.3 Application

On one hand, previous metamodels and transformations do not impose a concrete representation of the involved elements (functional requirements, test scenarios, operational variables, and test cases), but on the other hand, working directly with the metamodels object may be difficult, as shown in Figs. 8 and 9. NDT does not impose a concrete syntax for requirements, allowing the definition of use cases by means of either a model defined in UML or a text template. As it can be observed in Fig. 8, several concrete syntaxes may be used for defining functional requirements. The "?" indicates that any other syntaxes or formats plus the indicated one may be used.

Thus, the first step to apply the generation of functional test cases from functional requirements deals with defining a process for extracting a functional requirement model in accordance with the metamodel introduced in Sect. 4.1. This process depends on the specific syntax and it is out of the scope of this paper. Some previous work in this line was published in [14].

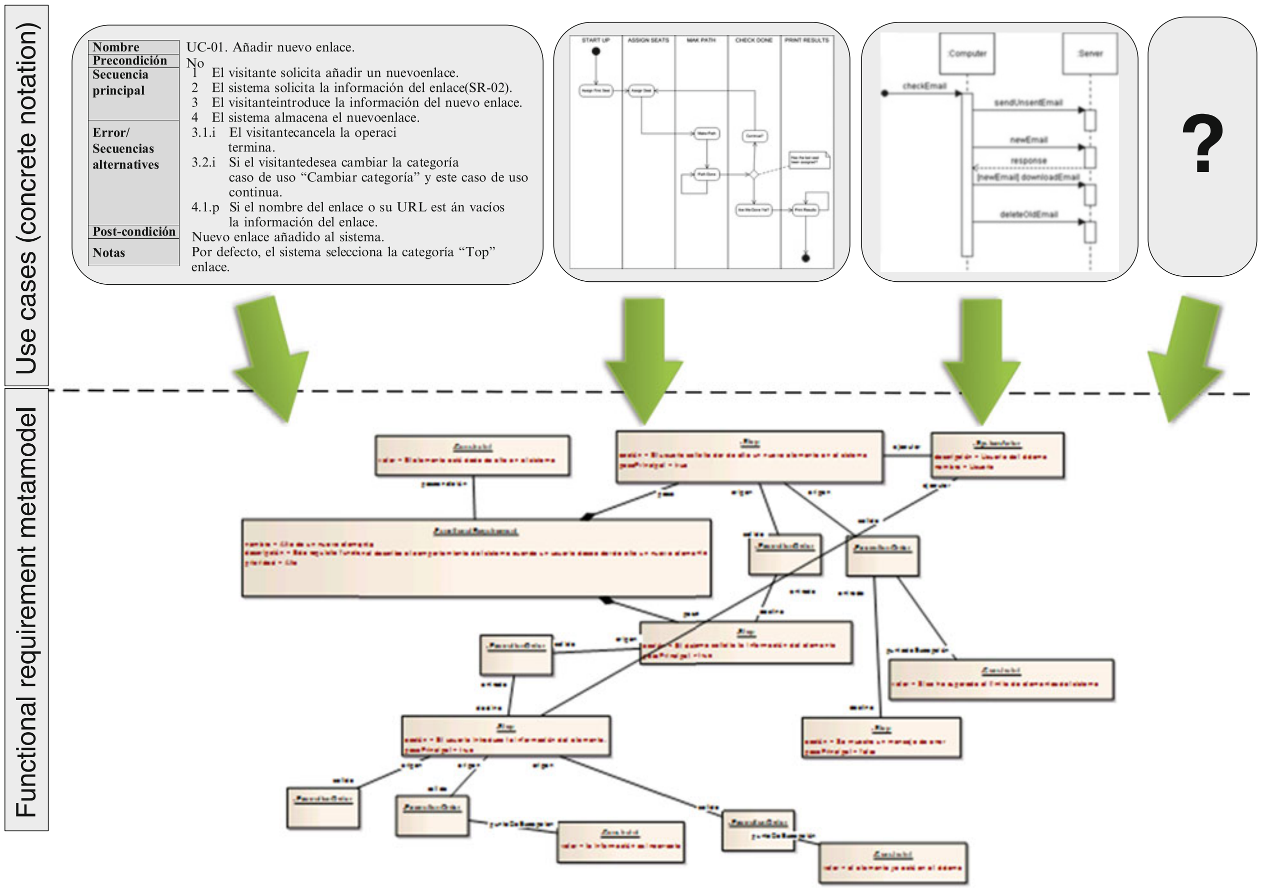


Fig. 8 Concrete syntaxes for functional requirements model

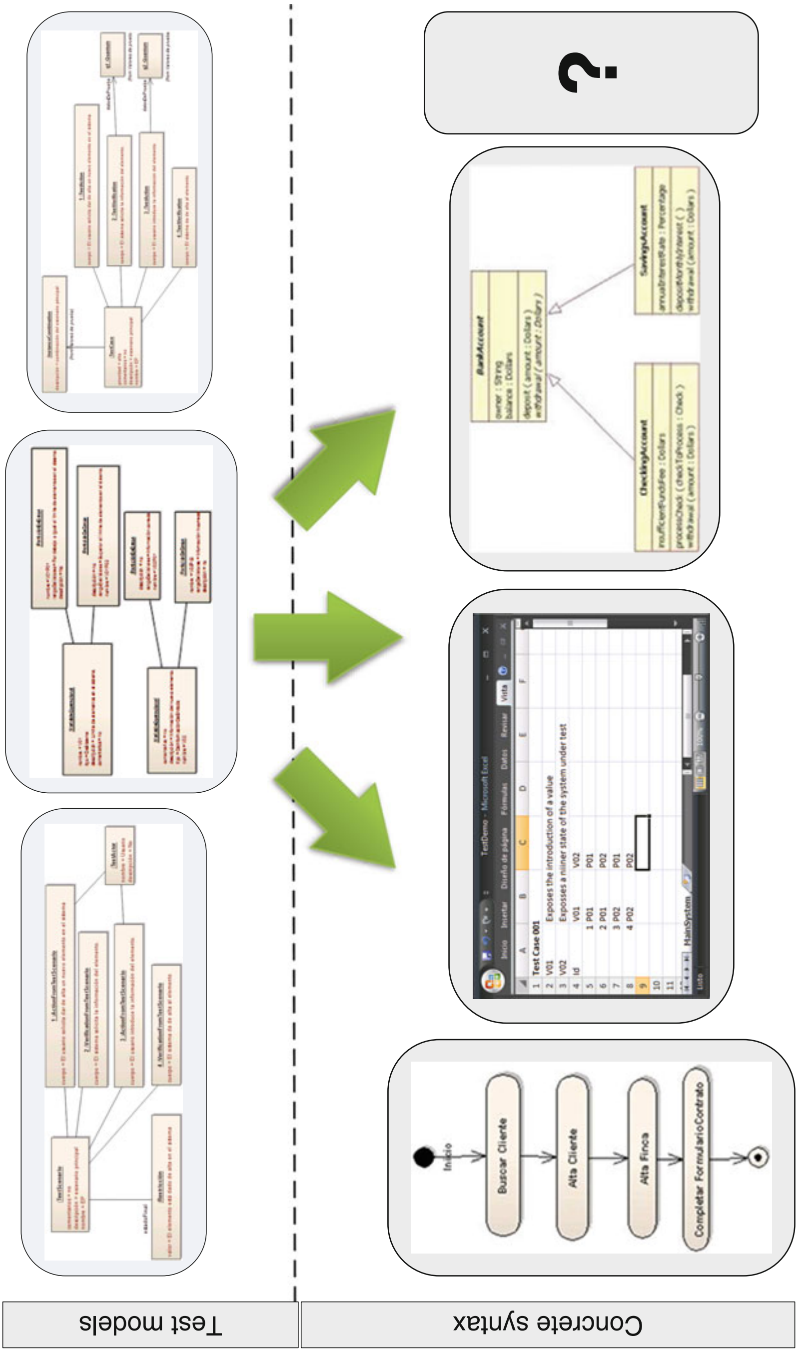


Fig. 9 Concrete syntaxes for test artifacts

Table 1 Metrics for QVT-operational code

	T0	T1	T2	T3
Total lines	124	118	290	170
Lines of codes	104	97	238	124
No. of mappings	1	4	5	3
No. of helpers	1	2	3	1
No. of queries	3	2	1	3
No. of input models	1	1	1	3
No. of output models	1	1	1	1

In the same way, a model including the testing artifacts obtained after the applications of both previous techniques may not result the most suitable syntax (Fig. 9). Even more, different models may require different syntaxes. For example, a valid syntax for a test scenario model (an activity diagram) may not be the proper syntax for instance combinations. In this case, a table or an Excel sheet (Fig. 9) should be more valuable. Again, a question mark in Fig. 9 represents any other valuable syntax. Next section introduces the software tools that implement these techniques and explain the concrete syntax they manage.

The relations stated in the previous section (T1, T2, and T3 from Fig. 7) were defined through the QVT-operational language as a necessary step to know how to implement the transformation process into an automatic tool. The QVT code may be downloaded from [15]. The metrics of the QVT code are collected in Table 1 and defined in [16].

Table 1 adds an additional transformation, called T0, not included in Fig. 7. This transformation contains common code used in other transformations. As reference, the Umls2Rdb transformation written in QVT operational and included in the QVT reference [17] has 65 lines of code, 6 mappings, and 1 query.

5 Practical Experiences

Nowadays, several companies in Spain work with NDT. This is possible due to the fact that NDT is completely supported by a set of free tools, mainly grouped in NDT-Suite [18]. This suite enables the definition and use of every process and task supported by NDT (Fig. 1) and offers relevant resources for quality assurance, management, and metrics with the aim of developing software projects. The suite was also extended to implement the first technique for test case generation using activity diagrams as the concrete input for functional requirements, and for the concrete syntax of the test scenarios generated. The implementation of the second technique is still an ongoing work.

However, the MDD perspective allows the concrete notations independency. Thus, the metamodels and transformations defined in previous section may be used out of the scope of NDT. The only request is that the source functional requirements must include the concepts defined in the functional requirements metamodel used as the basis for the process. To remark this independency, a second tool, called MDETest, was created. The main differences between this tool and NDT-Suite are that MDETest implements the three target metamodels and it generates the tool use instances only for metamodels, so that it does not impose any restrictions over the concrete notations of the functional requirements input. Nowadays, this tool supports activity diagrams such as the syntax for functional requirements whereas it does not support any concrete syntax for the output. This tool is also available in [15].

A very first application of this extension was the AQUA-WS [19] project. EMASESA is a public company which deals with the general management of the urban water cycle, providing and ensuring water supply to all citizens in Sevilla. AQUA-Web-Services (also called AQUA-WS) project consists in the development and implantation of an integrated business system for customer management, interventions in water distribution, cleaning, and net management. This system had 1,808 functional requirements, which individually include several scenarios and alternatives.

During the development of AQUA-WS project the development team used NDT-Tool to generate the test plan, which had over 7,000 test cases generated from the different scenarios out of the 1,808 functional requirements. Estimating 5-min length to create a test scenario in the modeling tool, the amount of time gained with NDT-Tool reached 583 h (73 days, working 8 h a week). Even more, the test cases obtained were classified in the right packages and they had tracing relations with the use cases source. The modeling tool used to manage use cases and test cases has search options to map the tracing relations, which makes more easy the task of working with a wide set of test cases.

6 Conclusions and Ongoing Work

This paper presents a model-driven process, based on metamodels and transformations, with the aim of generating test cases from functional requirements. As a result of this work, NDT has been enriched with metamodels and transformations so as to generate test cases from functional requirements automatically by means of the NDT-Suite tool.

Extension has been tested in several projects and it opens new research lines. Firstly, we have to work in test cases prioritization mechanisms, consisting in giving relevance to functional requirements, as well as in redundant test cases detection. The practice concludes that it continues producing a high number of redundant test cases that the test teams have to detect by hand. One last ongoing work would deal with supporting the semantic of the inclusion and extension relations defined in UML [20] for use cases.

Acknowledgements This research has been supported by the Tempros project (TIN2010-20057-C03-02) and Red CaSA (TIN 2010-12312-E) of the Ministerio de Ciencia e Innovación, Spain, and NDTQ-Framework project of the Junta de Andalucía, Spain (TIC-5789).

References

1. Schmidt DC (2006) Guest editor's introduction: model-driven engineering. Computer 39(2):25–31
2. Escalona MJ, Aragón G (2008) NDT. A model-driven approach for web requirements. IEEE Trans Software Eng 34(3):370–390
3. Myers G (2004) The art of software testing, 2nd edn. Addison-Wesley, Boston, MA
4. Denger C, Medina M (2003) Test case derived from requirement specifications. Fraunhofer IESE Report, Germany
5. Escalona MJ, Gutiérrez JJ, Mejías M, Aragón G, Ramos I, Torres J, Domínguez FJ (2011) An overview on test generation from functional requirements. J Syst Software: Elsevier 84(8):1379–1393
6. Fröhlich P, Link J (2000) Automated test case generation from dynamic models. ECOOP 2000, pp 472–491
7. Naresh A (2002) Testing from use cases using path analysis technique. In: International conference on software testing analysis & review
8. Mogyorodi GE (2002). What is requirements-based testing? In: 15th Annual software technology conference, Salt Lake City, USA, 28 Apr to 1 May
9. Boddu R, Guo L, Mukhopadhyay S (2004) RETNA: From requirements to testing in natural way. In: Proceedings of 12th IEEE international requirements engineering RE'04
10. Ruder A (2004) UML-based test generation and execution. Rückblick Meeting. Berlin
11. Binder RV (1999) Testing object-oriented systems. Addison Wesley, Boston, MA
12. Ibrahim R, Saringat MZ, Ibrahim N, Ismail N (2007) An automatic tool for generating test cases from the system's requirements. In: 7th International conference on computer and information technology, Fukushima, Japan
13. Ostrand TJ, Balcer MJ (1988) Category-partition method. Communications of the ACM, pp 676–686
14. Gutiérrez JJ, Nebut C, Escalona MJ, Mejías M, Ramos I (2008) Visualization of use cases through automatically generated activity diagrams. Lect Notes Comput Sci 5301:83–96
15. Supporting web www.iwt2.org/mdetest. Last updated 15 Apr 2012
16. Kapova L, Goldschmidt T, Becker S, Henss J (2010). Evaluating maintainability with code metrics for model-to-model transformations. Research into practice–reality and gaps, Springer, pp 151–166
17. Object Management Group (2011) Query View Transformation Specification 2.0. http://www.omg.org. Accessed 7 Jan 2012
18. García-García J, Cutilla CR, Escalona MJ, Alba M, Torres J (2011) NDT-Driver, a Java Tool to Support QVT Transformations for NDT. In: 20th International conference on information systems development, Edinburgh, Scotland, 24–26 August
19. Cutilla CR, García-García JA, Alba M, Escalona MJ, Rodríguez-Catalán L (2011) Aplicación del paradigma MDE para la generación de pruebas funcionales. Experiencia dentro del proyecto AQUA-WS. In: Proceeding of Automating Test Case Design, Selection and Evaluation ATSE 2011, Chaves, Portugal
20. Object Management Group (2011) Unified Modeling Language 2.4. www.omg.org. Accessed 24 June 2012

Architectures for Evolving System of Systems in Complex Environments

Igor Hawryszkiewycz

Abstract The paper describes design methods for designing a system of systems. Here a system must be designed to both carry out its function and also to fit in with other systems. Design goes beyond optimizing one system but must provide ways to sustain relationships between systems in open complex environments. The nature of a system of systems is complex as individual systems must respond to any changes in related systems. They are characterized by continuous emergence and the need to maintain system integration as systems change and support creativity and innovation. The paper describes architectural framework made up of two parts that provides a structure to model systems and their emergence. The first is an open system structure that models organizational structures that can evolve. This goes beyond the traditional hierarchical systems used in many models. Instead it sees organizational structures as collaborating open systems that are gradually leading to a social rather than hierarchical structure. The second is the collaborative architecture based on social networking superimposed on the structure to model collaboration between the systems. It thus combines social networking and an open model of business systems to provide a unified framework to model a system of systems.

1 Introduction

The increasingly specialized and competitive business environment today is creating a trend towards greater networking or what is sometimes called Enterprise 2.0 [1]. The emerging business networks are increasingly complex and dynamic in nature. They go beyond simple outsourcing of well-defined functions to providers. Now they also include complex networks where a number of organizations collaborate

I. Hawryszkiewycz (✉)
University of Technology, Sydney, NSW, Australia
e-mail: igorh@it.uts.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_15,

to cocreate products [2] or services [3]. They are characterized by continually changing requirements that require greater agility at all levels of an enterprise. Each network member has its own system and must integrate their processes for effective collaboration.

Jarke and others [4] describe these trends to more complex systems or what is increasingly called systems of systems. They call for greater emphasis on evolving requirements rather than software engineering requirements in this new environment. They suggest more emphasis on using architecture as the foundation while drawing on ideas from complexity theory for sustainable design. Baxter and Sommerville [5] also suggest the trend to continuous construction rather than design. A further emphasis in dynamic complex systems is the need to place greater emphasis on social structure [6] and facilitated collaboration. Collaboration is currently mostly ad hoc with some claim that an average of 75 min a day are spent on ad hoc collaboration [5]. Other writers [7, 8] call for a more focused approach to collaboration. Pisano and Verganti [7] suggest that collaboration should be aligned to the business process whereas Patel et al. [8] defines factors that lead to successful collaboration based on a lengthy study that involved a number of organizations.

The basis of the paper is that complexity can be managed by providing ways to quickly create new collaborative systems within the system of systems to address unexpected change. These need an open system structure and a systematic way to define the collaboration focused on business goals. These collaborative systems can grow if successful while at the same time existing units gradually reorganize to work within the emerging environment. This paper describes an architectural approach to organizing collaboration between systems. It is made up of two parts. One is the structure of businesses as a system of systems. The second part is the collaborative architecture [7] to model the collaborative processes for the systems to work together. The paper uses this architecture as a way to include collaboration and provides guidelines to system designers to create a collaborative architecture that shows the information flows needed to promote collaboration and support decision making in the system of systems.

The paper begins by describing the complex nature of the system environment in more detail. It then develops a way to create an architecture where collaboration between systems can evolve as systems dynamically change.

2 The Impact of Complex Environments on Design

Complexity here is as defined in [9]. It does not focus on mathematical solutions but ways to manage the continuous change in business relationships. It focuses on issues such as emergence of new business activities, self-organization, and ways to quickly respond to environmental change. This places greater demands on system architectures to provide ways to maintain network collaboration [10] as systems evolve. Communication between systems must continue as complexity causes arrangements to change. Naranayan and others [11] note that increasing complexity makes it difficult for firms to realize the full benefits of networking. The difficulty comes from managing communication across systems as systems change. The main high-level requirements here are:

- Designing for continuous scanning for the environment rather than dealing with predefined inputs
- Making choices of how to respond to the changes
- Emergence of new systems or relationships
- Responding to unexpected events by rearranging communication between existing functions
- Reorganization of existing business units
- Create new collaborative activity

Traditional methodologies focus on identifying a problem and proposing a solution to the problem. The solution includes the creation of software that addresses the problem. Once identified the problem defines clear requirements to be satisfied by the software. Traditional methodologies require anyone designing systems in complex environments to predict all possible inputs. Design methods in complex environments cannot simply focus predefined inputs but must continually scan the environment to identify new threats or opportunities and create structures to address these opportunities. The requirements here are:

- *Process integration* is an important goal for a collaborative architecture to enrich knowledge sharing and collaboration across systems [10] as this has a positive impact on firm performance. Baxter [5] suggests a number of perspectives used in such change stressing both organizational and social change that often results from external inputs in a complex environment.
- *Focused collaboration* is essential to address emergent issues. A collaborative architecture requires roles and interactions to be defined in ways that address emergent issues. The paper focuses on structures that go beyond ad hoc collaboration but introduce a more formal structure of roles and relationships that define business arrangements to address the new challenges. The requirement of a collaborative architecture is a clear definition of changes to role responsibilities and information flows.
- *Creativity*—Greater emphasis on a collaborative architecture that emphasizes social structures for organizational creativity [12] and ensuring that information flows support innovative activities.

3 System Architecture: The Structure of Systems

This paper proposes a way to describe systems [13] in terms of layers of social systems and shows how these can be extended to a collaborative architecture. These levels are:

- Society as a whole usually a nation state.
- Communities within the society such as local communities, professional groups, and sporting associations that make up society as a whole. Communities to

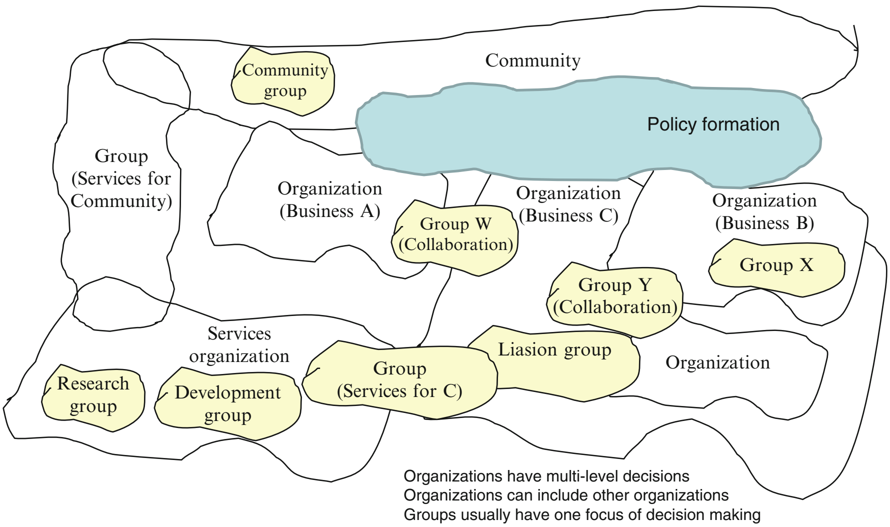


Fig. 1 The role of groups in system networks

same extent can be seen as setting standards of behavior for participants of the community.

- Organizations such as corporations that include levels of decision making.
- Groups that have a well-defined goal and work within the parameters set by a higher-level structure such as business units, committees, teams, or task forces.
- Activities in groups that carry out well-defined tasks.

The levels do not in any sense suggest a hierarchical organization. The difference in the layers is on kinds of processes within the level. They focus more on the responsibility of the level within the system of systems. Thus, groups usually address one function that requires one or two levels of decision making. The organizational level differs from the group level in that they have more than two levels of decision making. Communities are where systems can collaborate to identify common goals. Living systems theory also includes lower-level systems. An activity is where a group engages in such as to decide how to arrange a software module. For example, a small team working on system module would be seen as a group. A project that includes a number of teams would be seen as an organization as here there are decisions on how to create teams and then how to manage them. For convenience a collection of systems is here called an enterprise.

Each of the systems in each level has its own processes and subsystems and interacts through boundary roles with other systems. The architecture shown in Fig. 1 illustrates system architectures by the large number of relationships between systems. These relationships are not simply hierarchical as found in traditional systems. They can take place between any two or more systems. Figure 1 shows a service organization that provides services to the community through one group and services to Business C through another group. There is also collaboration between businesses A and C through collaborative group W. Similarly there is collaboration between businesses C and B through group Y. A number of these organizations (A, C, and B) are involved in policy formation for the organizations. In this paper groups play an important role in system evolution. Groups are seen either as fixed concrete groups as, for example, a development group in the service organization. There are also what we introduce here as "virtual groups." These are groups set up to facilitate collaboration, and members of these groups are also members of other groups, as for example, group W supports collaboration between organizations A and C.

4 System Architecture: The Collaborative Architecture

The goal of a collaborative architecture is to align the collaboration to the business. Each business system has its own system of communication. What is needed are ways to integrate the communication systems. This paper proposes the following:

- All messages must be processed within a system while decisions are passed across system boundaries by boundary roles.
- Systems communicate by the transfer of message between their boundary roles.
- Each system processes all input messages consistently with information subsystems and responds through boundary roles.

The paper now describes the system architecture followed by the detailed design of individual systems. The steps here are:

- Identify the systems and identify their goals and relationships between them.
- Define boundary roles and their rules of interaction.
- Define sustainable collaborative environments.

Figure 2 is an example of a way to describe an enterprise modeled as a system of systems. Here there are two partners and each partner as two systems within a system of system. Here:

Partner A develops products for distribution. The design is carried out in two groups, design groups A and B. Its boundary role is the product manager.

Partner B distributes the product for sale. Its boundary role is the delivery manager.

Each role is assigned some responsibility and links between these roles indicate that people assigned to the roles interact in carrying out their formal duties. For example, the product manager role is responsible for new product design for partner A. The roles are underlined in italics. The role responsibilities are shown in the boxes attached to the role.

There are two collaborative systems shown in Fig. 2. These are:

- Between design groups A and B for partner A, called design collaboration
- Between the two partners called "planning collaboration"

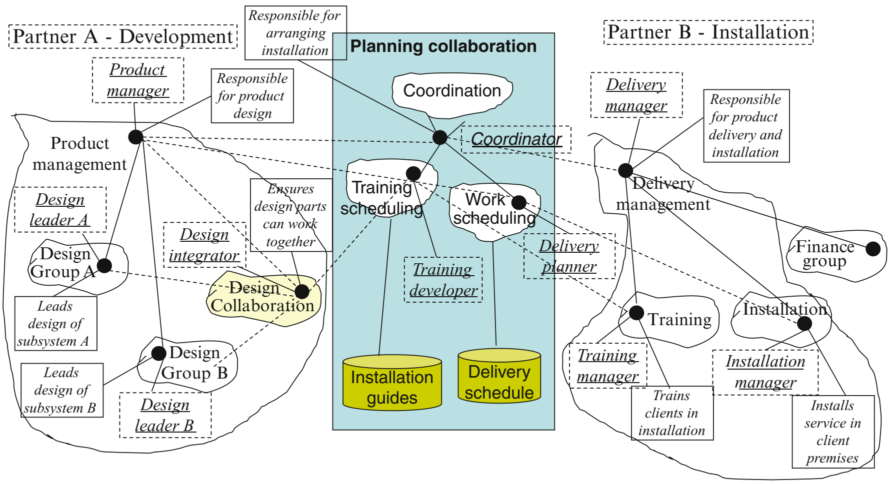


Fig. 2 Communicating systems—an overall architecture

4.1 Guidelines for Well-Formed Collaborative Environments

The important criterion here is to maintain sustainable collaboration.

- The goals of collaboration are clearly defined; these can require the formation of groups within the collaborative environment.
- There should be a coordinating role; this should be assigned to leaders or boundary roles in the participating systems—in this case the product manager and the delivery manager.
- Each group within the collaboration should report to the coordinator.
- Leaders of related system group should be assigned to roles within the collaborative group.
- Leaders of the groups report progress to the coordinator.
- All boundary roles should be included in the collaboration.

Using these guidelines, the design collaboration for Partner A is shown in Fig. 3. Its goal is to maintain consistency between the design subsystems.

Figure 3 shows the systems in product design. It has a product management system that interacts with clients and arranges product design to be carried out by two design groups. The design groups are coordinated through a design collaboration process:

- Identify boundary roles. Ensure there is one role to capture any relevant information. Create a role to transmit decisions of the system. In this case there are a number of boundary roles. Each design group has a communication officer as one of its boundary roles and the department manager as the other. In product

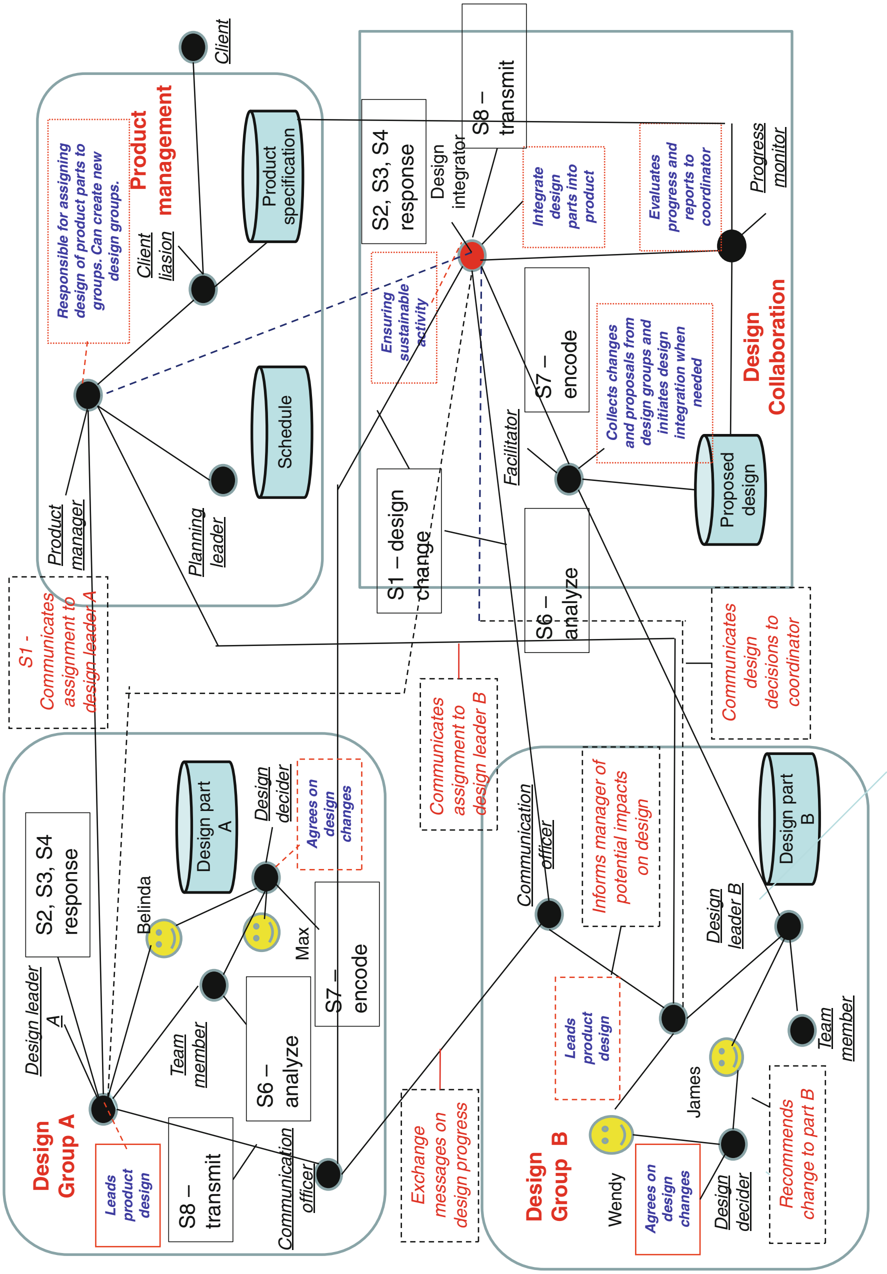


Fig. 3 A system representation of business networking

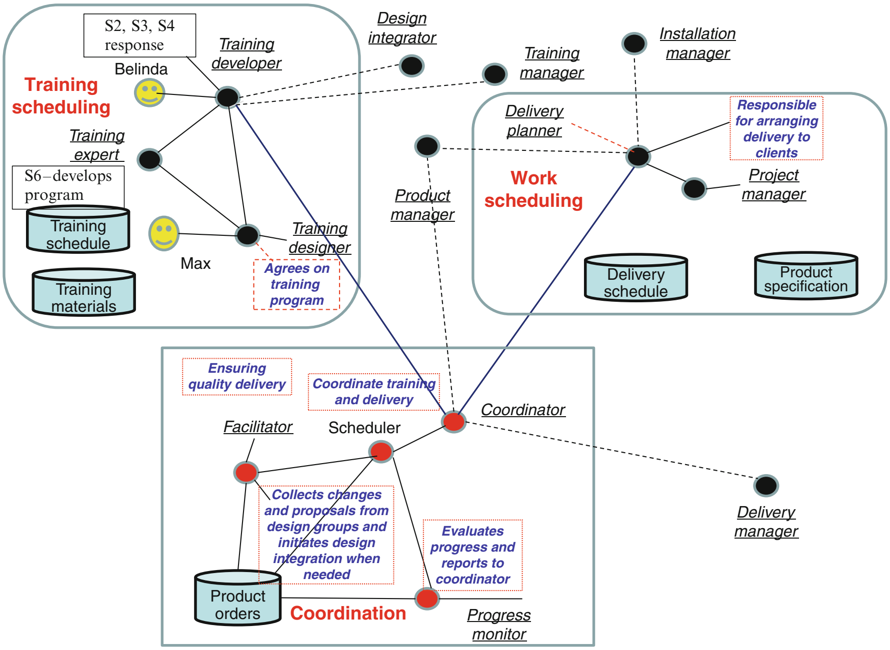


Fig. 4 Model of collaborative environment

development, for example, this is the client liaison, who interfaces with clients, and product manager who manages the design teams.

- Identify the expertise needed to deal with the inputs and distribute to roles with that expertise. In this case the role is taken by the department manager.
- The design requirements are distributed to the design manager, who works together with a number of team members.
- Include support for exploring ideas of how to deal with inputs. This requires ways to come up with ways to respond to change, evaluate them and decide. This is where association of ideas comes in and is supported for the team members and leader.
- Provide ways to initiate new systems as messages are associated. A system is created when new knowledge is required. This particularly happens more often with community systems, for example, running a model as part of policy formulation.

People from the participating systems are assigned to the roles in the collaboration. This is shown by the dotted lines. Examples are Patel et al. [8], who suggests connectors, and Sheate [14], who suggests greater emphasis on brokers. The emphasis on evolving architecture requires the ability to continually change relationships within a system while adapting to changes in requirements. It also implies a greater need to emphasize business rather than software design.

There is a coordination environment between the two businesses. The coordination, which is shown in Fig. 4, is to arrange training on the parts and a delivery schedule. Thus, the business managers of both partners jointly coordinate the collaboration.

To do this the paper uses the idea of messages as going through a number of information subsystems [13]. It sees all the systems in having their own processes that deal with messages that cross the system boundary. Each message is processed through the following subsystems:

S1—A subsystem to receive new messages across the system boundary.
S2—A subsystem to interpret messages created within the social system.
S3—An analysis to see the relevance of the messages and classify them.
S4—Distribution for further processing depending on the classification.
S5—Storing any useful information.
S6—Analyzing message for their effect on the current context.
S7—Making a decision on any action that is needed to respond the message.
S8—The decision is encoded into a form that is understood by the recipient and becomes part of the system boundary.
S0—Reproducer that creates a similar system.

The subsystems are usually assigned to roles as shown in Figs. 3 and 4 that are carried out by people designated to carry out the role. Messages provide a generic basis for defining the transfer of information and knowledge between systems. The messaging is used to identify ways to develop the collaboration to sustain change over extended periods of time.

5 Modeling Evolving Systems

The information subsystems provide guidelines for designing systems in detail. Figure 3 shows one such collaboration. It uses an open modeling approach based on semantics that focus on collaboration (Hawryszkiewycz, 2005). It is more structured than using rich pictures as the goal is to show information flows within systems.

Creativity itself has many dimensions. Nguyen and Shanks (2009), for example, distinguish between product creativity, process creativity, and people creativity. Chang [12] calls for greater structure in the communication needed to support creativity. Pisano and Verganti [7] similarly suggest that a collaborative architecture is needed.

5.1 Describing Emergence

There are thus some minimal basic commands that can lead to any possible emergence.

These semantics are then described in terms of lower-level concepts that can be used to define the detailed system structure. The remainder of the paper now describes lower-level semantics.

The structure of loosely connected by interacting systems provides the architectural foundation for evolving system of systems at the business level. It provides a structure where new groups can be established to foster spontaneous collaboration [15] between existing units with minimal disruption to the existing units. The importance of a business approach is to see evolution at the top levels. Modeling should give insights on ways to address and ways to continually restructure working arrangements in the business activities. A typical evolutionary scenario follows:

- Evolution often starts with role emergence with a role assuming new responsibilities often in response to some unexpected input.
- A particular role can then identify another role in another system to collaborate with.
- Together they create a collaborative group in the same way as described in Krogh [15] as emerging spaces.
- Roles are identified for the collaboration but the roles are taken by people in existing systems.
- New roles are added if collaboration grows and taken by people in existing systems.
- New roles are added and taken by new people if the collaboration becomes a source of new formalized practices and becomes ongoing and permanent.

The flexibility of such evolution provides a framework for managing within complex environments. As new messages come across a systems boundary, it must decide whether it is relevant to how to deal with it. The message can be managed within the system or if necessary passed across a boundary or requires the emergence of a new structure that satisfies well-formed criteria. They must also create a collaborative environment where they can leave together.

The basis of this paper is that people in different systems can initiate new structures. This requires both services and an implementation of these services. Often this is simple to reproduce a known group.

Create collaborative virtual space (often a workspace).
Create virtual role.
Assign existing role to virtual role.
Provide access to artifact.
Provide collaboration tools with the shared space.

6 Cloud Implementation

The cloud provides a way to support the growth of collaboration. How can a cloud be used to facilitate collaborative emergence by providing services to create collaborative emergent spaces within the cloud? A possible way is for a role to create a container in the cloud and then grow the collaboration over time (Fig. 5).

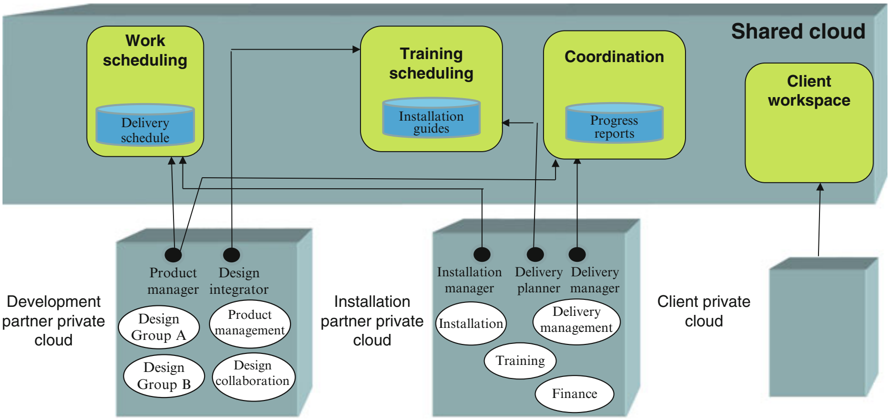


Fig. 5 Mapping to the cloud

7 Summary

The paper suggested that new approaches are needed to manage the evolution of a system of systems in the emerging networking environments. It used ideas from complexity theory to identify the requirements and defined a way to model enterprise architecture as a system of systems. It then defined a way to describe networked architectures as an open system for showing collaboration between systems. It then superimposed a collaborative architecture on the open system to model collaboration within open environments.

References

1. McAfee AP (2006) Enterprise 2.0: the dawn of emergent collaboration. MIT Sloan Manage Rev Spring:21–28
2. Cova B, Salle R (2008) Marketing solutions in accordance with S-D logic: co-creating value with customer network actors. Ind Market Manage 37:270–277, Elsevier Press
3. Vargo SL, Lusch RF (2008) From goods to service(s): divergences and convergences of logics. Ind Market Manage 37:254–259, Elsevier Press
4. Jarke M, Loucopouluds P, Lyytined K, Mylopoulus J, Robinson W (2011) The brave new world of design requirements. Inf Syst 36:992–1008
5. Baxter G, Sommerville I (2011) Socio-technical systems: from design methods to systems engineering. Interact Comput 23:4–17
6. Prahalad CK, Krishnan MS (2008) The new age of innovation. McGraw-Hill, New York, NY
7. Pisano GP, Verganti R (2008) What kind of collaboration is right for you. Harvard Bus Rev 83(8):80–86
8. Patel H, Pettitt M, Wilson JR (2012) Factors of collaborative working: a framework for a collaborative model. Appl Ergon 43:1–26

9. Merali Y, McKelvey B (2006) Using complexity science to effect a paradigm shift in information systems for the 21st. Century. J Inf Technol 21:211–215
10. Fathianathan M, Panchal JH (2009) Incorporating design outsourcing decisions within the design of collaborative design processes. Comput Indust 60:392–402
11. Narayanan S, Jayaraman V, Luo Y, Swaminathan J (2011) The antecedents of process integration in business process outsourcing and its effect on firm performance. J Oper Manage 29:3–16
12. Chang CM (2011) New organizational designs for promoting creativity: a case study of virtual teams with anonymity and structured interactions. J Eng Technol Manage 28:268–282
13. Miller J (1978) Living systems. Mc-Graw-Hill, New York, NY
14. Sheate WR, Partidario MR (2010) Strategic approaches and assessment techniques—potential for knowledge brokerage towards sustainability. Environ Impact Assess Rev 30:278–288
15. Krogh G, Nonaka I, Rechsteiner L (2012) Leadership in organizational knowledge creation: a review and framework. J Manage Stud 49(1):240–276

A Multimodeling Approach for Quality-Driven Architecture Derivation

Emilio Insfran, Silvia Abrahão, Javier González-Huerta, John D. McGregor, and Isidro Ramos

Abstract Product architecture derivation is a crucial activity in software product line (SPL) development since an inadequate decision during the architecture design directly impacts the quality of the product under development. Although some methods for architecture derivation have been proposed in the last few years, there is still a need for approaches that model the impact among architectural design decisions and quality attributes and use this information to drive the derivation of high-quality product architectures. In this paper, we present an approach for integrating quality attributes in early stages of the SPL lifecycle. The approach is based on a multimodel that explicitly represents the product line from multiple viewpoints (e.g., variability, functional, and quality) and the relationships among them, as well as on a derivation process that makes use of this multimodel to derive a product architecture with the required quality attributes from the product line architecture. The feasibility of the approach is illustrated using a case study on the automotive domain.

1 Introduction

A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features developed from a common set of core assets in a prescribed way [7]. Software product line engineering (SPLE) aims to develop

E. Insfran (✉) • S. Abrahão • J. González-Huerta • I. Ramos
ISSI Research Group, Universitat Politècnica de València,
Camino de Vera s/n, Valencia, Spain
e-mail: einsfran@dsic.upv.es; sabrahao@dsic.upv.es;
jagonzalez@dsic.upv.es; iramos@dsic.upv.es

J.D. McGregor
Department of Computer Science, Clemson University,
312 McAdams Hall, Clemson, SC, USA
e-mail: JohnMc@cs.clemson.edu

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_16,

these systems by taking advantage of the massive reuse of core assets in order to improve time to market and product quality.

Software product line development has characteristics, which distinguish it from the development of individual products. In SPL development there is a product line architecture, which includes the variations needed to cover the different products, and the product architectures, which are instances of the product line architecture achieved by exercising variation mechanisms. However, in those cases where levels of quality attributes that fall outside the original specification of the product line are needed (and cannot be achieved by the product line variation mechanisms), some architectural transformations may be applied when deriving a product architecture to ensure that these levels of quality attributes are met [3].

Although several methods for architecture derivation using multiples viewpoints have been proposed in the last few years (e.g., [2, 4, 9, 16]), there is still a need for approaches that model the impact among architectural design decisions and quality attributes and use this information to drive the derivation of high-quality product architectures. The objective of this paper is to present a multimodel (i.e., a set of interrelated viewpoints describing the products that can be derived from the product line) as well as its use for deriving product architectures with the required quality attributes from the product line architecture. This multimodel contains the functional and variability viewpoints and also integrates a quality viewpoint, allowing the explicit representation of the relationships and constraints among the elements of these viewpoints.

The main contribution of the paper is the use of model-driven engineering principles as a way to provide a richer semantic representation of a software product line (the multimodel). It provides a sufficiently formal interrelated model that can be supported by tools capable of automating portions of the product line production planning. In this paper, we illustrate how the multimodel can be used as input in a quality-driven derivation process that will apply architectural transformations (i.e., architectural patterns) to the product line architecture in order to derive product architectures with the required levels of quality attributes, when this cannot be achieved by exercising the product line built-in variation mechanisms.

The remainder of the paper is structured as follows. Section 2 discusses existing approaches for the automatic derivation of product architectures. Section 3 provides details of the multimodel concept and the different viewpoints that can be used for representing a software product line. Section 4 describes a quality-driven model transformation process that allows obtaining product architectures with the desired quality attributes. Section 5 illustrates the feasibility of the approach through a case study in the automotive domain. Finally, conclusions and future work are presented in Sect. 6.

2 Related Work

Several approaches dealing with the use of multiple viewpoints for the automatic derivation of product architectures have been proposed in the last few years (e.g., [2, 4, 9, 16]).

In the approach by Botterweck et al. [2], the product architectures are produced by means of an ATL model transformation process, which takes as input a domain architecture model and an application feature model and generates an application architecture model, simply by copying the elements. This approach describes the product line architecture as the union of components that can be present in any configuration, but does not explicitly address quality attributes. Similarly, Cabello et al. [4] produce product architectural models by means of a QVT transformation. The transformation takes as input the variability view expressed in a feature model and the modular view of the architecture and generates the PRISMA component and connector architectural view. Furthermore, Perovich et al. [16], automate the derivation of product architectures by taking as input a feature configuration model. The transformation encapsulates the knowledge of how to build the product architecture when the corresponding feature is present in the feature configuration model. Finally, Duran-Limon et al. [9] present an approach to modeling the architectural variation points and the variability tree and then using ontology queries to obtain the commonalities, variants, and dependencies of a set of selected features. The output generated by the ontology queries is the input of an ATL transformation that generates the product architecture.

These approaches automate, to varying degrees, the derivation of product architectures by means of transformations using multiple viewpoints. These approaches do not integrate quality attribute concerns in the transformation process.

3 A Multimodel for Describing Software Product Lines

A *multimodel* is a set of interrelated models that represents different viewpoints of a particular system. A *viewpoint* is an abstraction that yields a specification of the whole system restricted to a particular set of concerns. In any given viewpoint, it is possible to define a model of the system that contains only the objects that are visible from that viewpoint. Such a model is known as a *viewpoint model* or a *view* of the system from that viewpoint [1].

In SPL development, the multimodel can be used to represent the different viewpoints of a set of products that can be derived from the product line. We distinguish (at least) three different viewpoints: variability, functional, and quality. The relationships with the quality viewpoint express the relative impact that the elements of the variability or the functional view have on the quality attribute levels of the final product. A quality attribute level indicates a threshold which a system must cross to be acceptable to the stakeholders.

The multimodel plays a key role in the SPL development process at two different levels: (1) in the *Domain Engineering* phase to express the impacts and constraints among variations, functional components, and quality attributes, providing in this way a more comprehensible and integrated model of the product line, and (2) in the *Application Engineering* phase to provide information to the transformation processes that will integrate the production plan. The production plan describes how the products are produced from the core assets including the derivation of product architectures from the product line architecture.

3.1 The Multimodel Viewpoints

The multimodel is composed of three viewpoints: variability, functional, and quality. The *variability viewpoint* has been defined using a variant [11] of the cardinality-based feature model presented in [5]. The main element of this model is the *feature*, which is a user-visible aspect or characteristic of a system. The cardinality-based feature model is a hierarchy of features but also allows the definition of how many clones of a feature can participate in a specific product configuration. Cloning features is useful in order to define multiple copies of a part of the system that can be differently configured [11]. Features can be organized in *feature groups*, which can have *group cardinalities*. This cardinality restricts the minimum and the maximum number of group members that can be selected. In feature models it is common to define relationships and constraints among features. According to [11], these relationships can be hierarchical: (1) *has_a*: structural relationship between a parent and *n* copies of a child feature as defined by its cardinality and (2) *is_a*: structural relationship between a parent feature and a group of child features; non-hierarchical: (1) *implication*: the presence of the feature A implies the presence of the feature B, (2) *co-implication* the presence of the feature A implies the presence of the feature B and vice versa, (3) *exclusion*: the presence of the feature A implies that the feature B should not be present and vice versa, and (4) *use*: specifies that a specific instance of feature A will be related to one (or more) specific instances of feature B as defined by its cardinality. This specific variation of the cardinality view feature model also allows adding OCL-like constraints to the model and, based on these constraints, to check whether the instances are valid or not. The feature model variant has been specifically defined for being applied in a model-driven product line development context.

The *functional viewpoint* on the multimodel represents the structure of the system containing the different software components that satisfy the requirements of the different features and the SPL architecture. The functional view can be defined using different styles (e.g., component and connector, modular, allocation) [6]. In the last years, several Architectural Description Languages (ADL's) and notations has been proposed for documenting architectures such as AADL, SysML, UML, or EAST-ADL. We have selected the Architectural Analysis and Design Language (AADL) [10] for expressing the functional view of the multimodel because (1) it is an extensible standard in the avionics, aerospace, automotive, and medical device industries; (2) it provides mechanisms for dealing with variability at the architectural level, as demonstrated in [18]; (3) AADL models are a good basis for evaluating quality attributes that focus on the design of an architecture [6]; and (4) the structure of the toolsets supporting AADL allows the implementation of analysis models operationalizing different software measures. The AADL standard defines a textual and graphical representation of the runtime architecture of software systems as a component-based model in terms of tasks, their interactions, and the hardware platform where the system is executed.

The *quality viewpoint* on the multimodel has been defined by a quality model which is an extension of the ISO/IEC 25010 (SQuaRE) standard [13] to evaluate the

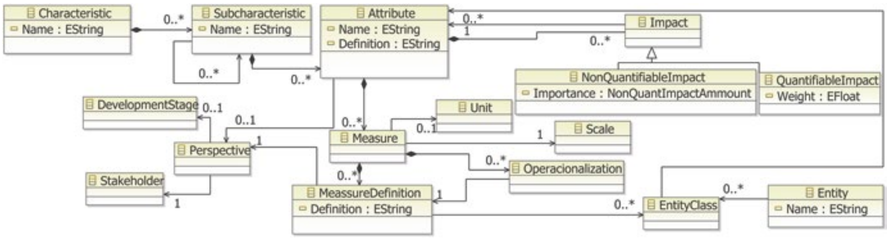


Fig. 1 Quality view metamodel

quality of SPLs. Figure 1 shows the main classes of the metamodel that gives support to the quality view, with the hierarchical decomposition of quality in characteristics, sub-characteristics, and quality attributes. The model shows that a quality attribute can be associated with different quality perspectives, which are characterized by a SPL development phase and/or type of stakeholder. A quality attribute can positively or negatively impact other quality attributes (e.g., testability impacts positively on maintainability, maturity of core assets impacts positively on reusability). Those impacts can be expressed using a nominal or ordinal scale measure (when it is described by a label expressing the relative importance or severity) or using a ratio scale measure (when it is described by a numeric value that expresses the impact). The model also expresses the software measures associated with each quality attribute. Each measure can be operationalized depending on the software artifact and/or the quality perspective that is selected.

3.2 Relationships Among Viewpoints

In order to define production plans to guide the development of software products using the multimodel, we should establish relationships among the elements of the *viewpoint models* or *views* that represent the product line viewpoints. This will allow us to formally analyze properties over the product line as a whole. The multimodel has been defined with the following relationships among views:

- *Functional and variability views*: a set of elements in the functional view (e.g., AADL systems, system implementations, devices, and processes) can be combined in order to fulfill the requirements of one or more features (e.g., a multimedia GPS navigator feature in a car is fulfilled by a combination of software and hardware components) and can be seen as a subset of the relationships that can be defined applying the VML4Arch approach [20]. Furthermore, the constraints defined in the feature model allow this model to become a filter to partially determine what elements can be selected in the functional model.
- *Functional and quality views*: *components* of the functional view may impact one or more *quality attributes* (e.g., integrating *components* in the functional view with low resource consumption will impact on the *resource consumption*

quality attribute of the quality view). During the product derivation, the application-specific quality attributes, selected by the application engineer, and the impacts established among the functional and quality views drive the selection of the most appropriate components that satisfy the quality attribute requirements.
- *Variability and quality views*: *features* or *feature groups* in the variability view may impact one or more *quality attributes* of the product (e.g., a feature group related to safety options may improve the reliability of a system).

The multimodel helps to have a richer semantic view of the product line as well as to ensure the consistency of the relationships among the different elements of the product line viewpoints. Furthermore, the product line production planning can be guided by the structural constraints established by the multimodel.

4 Quality-Driven Product Architecture Derivation

Product definition begins with the derivation of the product architecture. The multimodel can be used to create product-specific production plans that describe how a specific product can be built from the core assets, including how the product architecture can be derived from the product line architecture. The product line architecture defines the allowable variations within the product line's scope and the variation mechanisms for achieving them. In some cases the product architecture may include variation points that are not permitted by the original product line architecture or to assure some product-specific quality attributes. We describe how the multimodel is used for deriving product architectures with the required quality attribute levels from the product line architecture. In this work, we focus on *architectural patterns* [3, 8], which are represented as *architectural transformations*, as a means to ensure the quality of the product architectures.

In the domain engineering phase, our approach requires the domain expert to establish the impacts that the different alternative architectural transformations have on the quality attributes of the product architecture. The impact that each alternative has on the quality attributes can be determined by using empirical evidence or based on the domain experts' experience. In order to perform a trade-off analysis among quality attributes, the Analytic Hierarchy Process (AHP) can be applied. AHP [17] is a decision-making technique used to resolve conflicts in which it is necessary to address multi-criteria comparisons. The result of the AHP is a comparison matrix that shows the relative importance of each alternative architectural transformation with regard to each quality attribute and is used as input for the quality-driven product architecture derivation process. The trade-off process do not consider all the possible quality attributes in a given domain but only those quality attributes that could be directly impacted by the architectural patterns and which are of relevance in a specific context (e.g., safety-critical systems for the automotive domain).

Figure 2 shows an overview of this derivation process, which is performed in the application engineering phase. The starting point is a product line architecture that

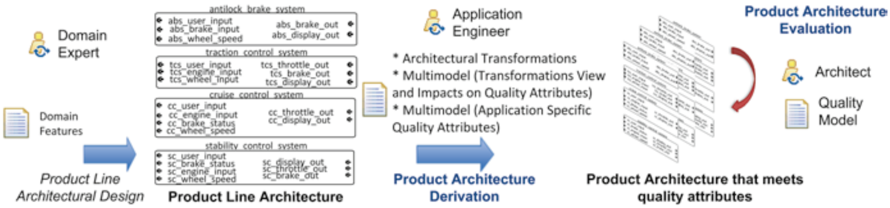


Fig. 2 Overview of the quality-driven product architecture derivation process

has been designed by the product line architect for supporting the whole set of possible products within the product line scope. The process contains two different activities, which can be performed by model transformations: product architecture derivation and product architecture evaluation.

The first model transformation (product architecture derivation) creates the product architecture as an instance of the product line architecture by applying an architectural transformation to meet the required quality attribute levels. This transformation takes as input (1) the product line architecture, (2) the comparison matrix with the impacts among the alternative architectural transformations and the quality attributes, and (3) the desired product-specific quality attributes selected by the application engineer. This transformation generates as output a product architecture. The second model transformation (product architecture evaluation) applies the software measures from the quality view of the multimodel to the derived product architecture in order to evaluate if it satisfies the desired quality attribute levels. This transformation takes as input the derived product architecture, the required quality level, and the quality view from the multimodel and generates as output an evaluation report.

5 Case Study

This section presents a case study in the context of the safety-critical embedded systems from the automotive industry. The Vehicle Control System (VCS), which is based on the example introduced in [12], was modified and extended in order to apply the variation points described in [18]. This system is composed of several subsystems (features) which are (1) Antilock-Braking System (ABS), which ensures that the maximum braking force is transmitted to all four wheels of the vehicle; (2) Traction Control System (TCS), which prevents the wheels from slipping; (3) Stability Control System (SCS), which keeps the vehicle going in the direction in which the driver is steering the car; and (4) Cruise Control System (CC), which attempts to maintain a constant driver determined speed.

These features comprise a set of embedded systems present in modern automobiles. Figure 3 shows the inner structure of the product line architecture for the

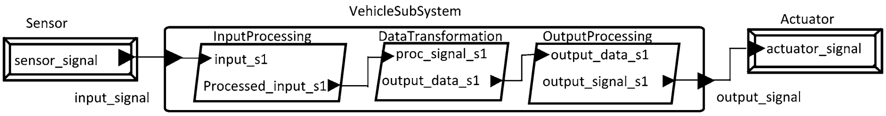


Fig. 3 Inner structure of the product line architecture for the VCS subsystems

subsystems, which are composed of *sensors* that collect input signals, *processors* that process and transforms the inputs, and *actuators* that control the mechanical parts of the system (e.g., engine, throttle position, brakes, security belts).

5.1 Domain Engineering

In the domain engineering phase, the domain expert must identify the quality attributes and the architectural transformation patterns that will allow to achieve the required quality attribute levels when this cannot be achieved by the product line variation mechanisms. The product architecture derivation process is driven by the key quality attributes. Safety, reliability, and performance are considered key quality attributes in real-time embedded systems for the automotive domain [19]. In this case study, we focus on the reliability sub-characteristic *fault tolerance* and the performance sub-characteristic *time behavior*. In particular, the time behavior sub-characteristic is broken down into *latency time*, which is the time elapsed between firing an input event and obtaining a response from the system.

The product architecture derivation process will use the quality view of the multimodel to determine which architectural pattern must be applied to the product line architecture in order to achieve the desired quality attribute level. In particular, the architectural patterns that we analyze in this case study are the homogeneous redundancy pattern and the triple modular redundancy pattern.

The *homogenous redundancy* pattern [8] (HR) improves the *reliability of the product* by offering multiple (in our case two identical) units of subsystem for monitoring and performing the same operations on the input signals. The disadvantages of this pattern are the recurring costs and a lack of coverage for systematic faults (if a fault systematically occurs when running the primary channel, it can also occur when running the backup channel). The primary channel runs as long as there are no problems detected. In the case of a failure in the primary channel, the system is able to detect the fault and switch to the backup channel and vice versa. There is no concurrency at runtime, only replication.

The *triple modular redundancy* pattern [8] (TMR) improves both the *reliability* and *safety* of a system by offering an odd number of channels operating in parallel. TMR negatively affects *performance*, since there are three subsystems running in parallel. The computational results or resulting actuation signals are compared, and if there is a disagreement, then the results with a two out of three majority win and are sent to the actuator. If the system suffers a failure affecting only one channel, then the other two channels will continue to produce the correct result.

Table 1 Architectural patterns and quality attributes trade-off analysis

(a)	Fault tolerance		Latency		(b)	Fault tolerance		Latency		(c)	Impacts	
	TMR	HR	TMR	HR		TMR	HR	TMR	HR		Fault tolerance	Latency
TMR	1	5	1	1/3	TMR	1/1.2	5/6	1/4	1/3/1.3	TMR	0.83	0.24
HR	1/5	1	3	1	HR	1/5/1.2	1/6	3/4	1/1.3	HR	0.17	0.76
Sum	1.2	6	4	1.3								

In order to define the corresponding relationships among architectural patterns and quality attributes, the domain expert must rank each pattern with regard to the Q quality attributes in a trade-off analysis using the AHP technique. For each quality attribute Q_a, he or she compares the N potential architectural patterns in a pairwise comparison. To determine how an architectural pattern Ax supports the quality attribute Q_a, in comparison to the pattern Ay, a weight is assigned (1 for equally important, 3 for moderately more important, 5 for strongly more important, 7 for very strongly more important, and 9 for extremely more important). For example, the domain expert defines that TMR is strongly more important (a weight of 5) than HR with regard to fault tolerance, and that HR is moderately more important (a weight of 3) than TMR with regard to latency.

The result of this comparison is an $N \times Q$ matrix that shows the relative support of the different architectural patterns to the quality attributes as shown in Table 1(a). Then, these values are normalized by applying the (1) to Table 1(a) to produce Table 1(b), and finally, the *impact* that an architectural pattern has on a quality attribute Q_a is calculated by applying the (2) to produce Table 1(c). This information is valid for all the products in the product line.

$$NormQ_y[i,j] = \frac{Q_a[i,j]}{\sum_{k=1}^{n} Q[k,j]} \tag{1}$$

$$I[i] = \frac{\sum_{k=1}^{n} NormQ_a[i,k]}{n} \tag{2}$$

5.2 *Application Engineering*

In the application engineering phase, the application engineer introduces the quality attribute levels Q that the specific product must fulfill as normalized weights ranging from 0 to 1. For k quality attributes, the transformation process calculates the ranking R for each pattern j by applying the (3). For example, introducing a weight of 1 for *fault tolerance* and 0 for *latency* will make the transformation process to select the TMR pattern using the impact values in Table 1(c) (TMR: $1 \times 0.83 + 0 \times 0.24 > $ HR: $1 \times 0.17 + 0 \times 0.76$).

$$R_j = \sum_{i=0}^{k-1} Qi * Iij \tag{3}$$

We have defined the model transformations to perform the product architecture derivation in QVT-Relations [15]. These transformations are not included in this paper due to space constraints. The transformations were applied to the product line architecture shown in Fig. 3 resulting in the product architecture shown in Fig. 4a, b depending whether the quality attribute *fault tolerance* or *latency* is selected.

After applying the architectural transformations, we evaluate the derived product architecture in order to assess if the application of the architectural transformation pattern resulted in an improvement of the quality of the product architecture. This is done by comparing the results of the metrics measured over the product architecture derived directly from the product line architecture with the results of the metrics measured over the product architecture derived after applying the architectural pattern. As an example, we use the fault tolerance quality attribute to illustrate the product architecture evaluation. The fault tolerance quality attribute can be measured by applying the *key node safety* metric [14] over a fault tree. The fault tree analysis is a top-down approach to failure analysis, starting with a potential undesirable event called a *top* event, and thereafter determining all the ways in which this event can happen. The causes of a top event are connected by logic gates OR, AND, or XOR. Figure 5b shows the fault tree associated with the *System Failure* event (the system gives no response due to sensor error, input processing error, or failure on the actuator).

The *key node safety* metric provides a design tool for comparing a fault trees' fault tolerance without requiring a priori knowledge of the component reliability. This metric is based on identifying *key nodes* within a fault tree and considers the impact of these nodes on the safety of the system. A *key node* is a node in a fault tree that allows a failure to propagate towards the tree root if and only if multiple failure conditions exist in the node. Therefore, a *key node* can be identified in a fault tree as a node that connects two or more events by means of an AND or XOR gate. The lower bound of this metric is 0 and occurs when any single component failure in the fault tree causes the system to fail. The upper bound is open and occurs when the system fails if and only if every component fails and its value depends on the number of nodes in the fault tree. The value of the metric expresses how a change in the architecture improves its fault tolerance; the higher the upper bound is, the better the fault tolerance the system has [14]. Figure 5a, b show the fault tree including the *key node* information for the product architecture generated with and without the application of the TMR pattern, respectively.

The (4) operationalizes the *key node safety* metric. Table 2 shows the metric calculation for the product architecture with and without the application of the TMR pattern (TMR and Original). This metric shows that the application of the TMR pattern to the product architecture improves the fault tolerance of the product when

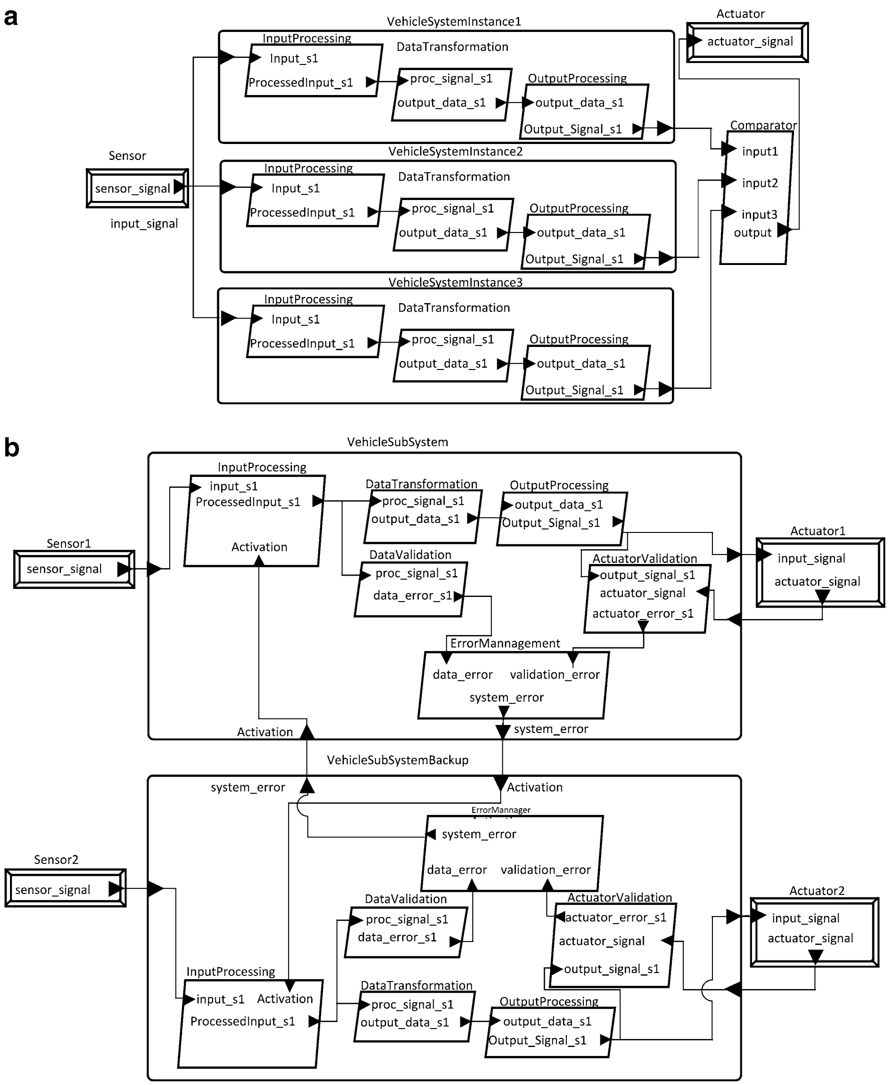


Fig. 4 Alternative product architectures prioritizing fault tolerance or latency

compared with the values for the product architecture without the application of the TMR pattern (TMR: 0.069 > Original: 0).

$$S = \frac{kh'}{n^2} \sum_{i=0}^{k-1} \frac{c_i}{d'_i} \tag{4}$$

This case study has allowed us to illustrate the use of a multimodel to establish relationships among quality attributes and architectural transformation patterns.

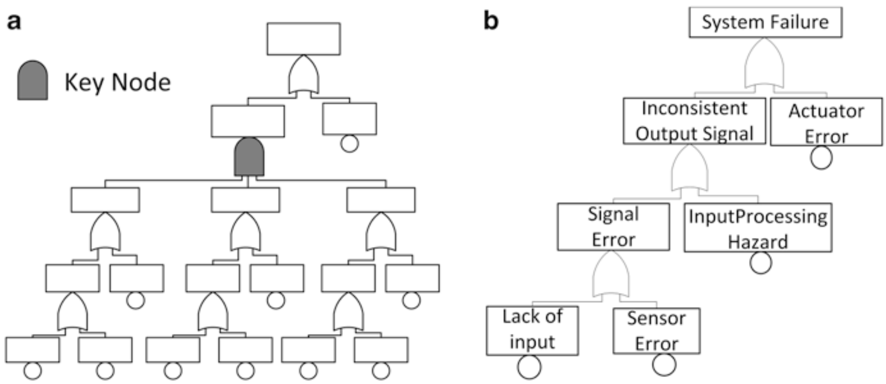


Fig. 5 Fault tree for the product architecture *with* and *without* applying the TMR pattern

Table 2 Key node safety metric calculation

	Original	TMR
k: Number of key nodes in the fault tree	0	1
h': Total height of the fault tree +1	5	6
n: Total number of nodes in the fault tree	7	18
c_i: Number of nodes in the sub-tree rooted at key node k_i	0	15
d'_i: Depth of the sub-tree rooted at key node $k_i + 1$	0	4
S: Key node safety metric	0	0.069

These relationships were used in a transformation process to determine which architectural pattern should be applied to the product architecture in order to meet specific quality requirements.

6 Conclusions and Future Work

This paper presented an approach for quality-driven product architecture derivation. The approach relies on the definition of a multimodel for representing the different software product line views (quality, functional, and variability) and it is used to drive the derivation of product architectures from the product line architecture taking into account quality requirements. The feasibility of the approach has been illustrated through a case study on the automotive domain. This approach has three main benefits: (1) the multimodel helps to have a richer semantic view of the product line by defining relationships among elements of its constituents views; (2) the architectural model transformations are driven by the relationships and the structural constraints established in the multimodel; (3) the quality view contributes to the quality improvement of the product architectures in the product line.

As future work, we plan to introduce the multimodel into the product configuration process and to select and deploy core assets over the product architecture taking into account functional and the quality requirements that the product must fulfill. We also plan to apply this approach in other domains where architectural patterns and quality attributes have been identified, to empirically validate the approach in practice, and to study trade-off and optimization techniques for improving the linear multi-objective AHP technique.

Acknowledgments This research is supported by the MULTIPLE project (MICINN TIN2009-13838), the ValI+D fellowship program (ACIF/2011/235) and the Universitat Politècnica de València PAID program (PAID-00-12).

References

1. Barkmeyer EJ, Feeney AB, Denno P, Flater DW, Libes DE, Steves MP, Wallace EK (2003) Concepts for automating systems integration. NISTIR 6928, National Institute of Standards and Technology, U.S. Department of Commerce, USA
2. Botterweck G, O'Brien L, Thiel S (2007) Model-driven derivation of product architectures. Proceedings of the ASE 2007 conference. ACM, New York, USA, pp 469–472
3. Bosch J (2000) Design and use of software architectures. Adopting and evolving product-line approach. Addison-Wesley, Harlow
4. Cabello ME, Ramos I, Gómez A, Limón R (2009) Baseline-oriented modeling: an mda approach based on software product lines for the expert systems development. Conference on intelligent information and database systems, Dong hoi, Vietnam, pp 208–213
5. Czarnecki K, Kim CH (2005) Cardinality-based feature modeling and constraints: a progress report. Proceedings of the international workshop on software factories, San Diego, CA, USA
6. Clements P, Bachmann F, Bass L, Garlan D, Ivers J, Little R, Merson P, Nord R, Stafford J (2010) Documenting software architectures: views and beyond. Addison-Wesley, Boston, MA
7. Clements P, Northrop L (2007) Software product lines: practices and patterns. Addison-Wesley, Boston, MA
8. Douglass BP (2002) Real-time design patterns: robust scalable architecture for real-time systems. Addison-Wesley, Boston, MA
9. Duran-Limon HA, Castillo-Barrera FE, Lopez-Herrejon RE (2011) Towards an ontology-based approach for deriving product architectures. Proceeding of SPLC 2011 conference, vol. 2, Article 19, Munich, Germany, p 5
10. Feiler PH, Gluch DP, Hudak J (2006) The architecture analysis & design language (AADL): an introduction. Tech. Report CMU/SEI-2006-TN-011. SEI, CMU, USA
11. Gómez A, Ramos I (2010) Cardinality-based feature modeling and model-driven engineering: fitting them together. Proceeding of the VAMOS 2010 workshop, Linz, Austria
12. Hudak J, Feiler P (2007) Developing AADL models for control systems: a practitioner's guide. Tech. Report CMU/SEI-2007-TR-014, SEI, CMU, USA
13. ISO/IEC 25010:2011 (2011) Systems and software engineering, systems and software quality requirements and evaluation (SQuaRE), system and software quality models
14. Needham D, Jones S (2006) A software fault tree metric. In: Proceeding of 22nd IEEE international conference on software maintenance, Philadelphia, PA, USA, pp 401–410
15. OMG (2008) Meta Object Facility 2.0 Query/view/transformation specification. http://www.omg.org/spec/QVT/1.0/PDF
16. Perovich D, Rossel PO, Bastarrica MC (2010) Feature model to product architectures: applying MDE to software product lines. Proceeding of WICSA/ECSA 2010 conference, Helsinki, Finland, pp 201–210
17. Saaty TL (1980) The analytical hierarchical process. McGraw-Hill, New York, NY

18. Shiraishi S (2010) An AADL-based approach to variability modeling of automotive control systems. Proceeding of the MODELS conference Oslo, Norway, LNCS 6394:346–360
19. Thiel S, Hein A (2002) Modeling and using product line variability in automotive systems. IEEE Software 19(4):66–72
20. Zschaler S, Sanchez P, Santos J, Alferez M, Rashid A, Fuentes L, Moreira A, Araujo J, Kulesza U (2009) VML* a family of languages for variability management in software product lines. In Proceeding SLE 2009 conference, Denver, USA

IS Degrees: Sociotechnical or Technosocial? A Reflective Analysis

Jenny Coady, Tessa Berg, and Rob Pooley

Abstract This paper describes a reflective and empirical investigation by staff involved in delivery of the Heriot-Watt University Information Systems undergraduate degree. It looks particularly at the practical relevance of the IS 2010 model curriculum, asking to what extent the Heriot-Watt University Information Systems teaching staff supports the model and identifying areas of conflicting opinion. These are compared with empirical evidence of relevance to employability in the form of statistics on graduate destinations from cohorts in 2010 and 2011. An IS virtual alumni forum was set up for past students to debate the IS degree and discuss areas of relevance to careers and suggestions for further improvement. In agreement with the IS 2010 model, graduate discussion and working panel reflection both propose the inclusion of a project management course. Graduates also suggest the need for more specialised technical skills in software and web development and a stronger grounding in spreadsheet application software including statistical processing methods. In agreement with the working panel, graduates state the importance of having quality experienced teaching delivered with skill and enthusiasm. Further work is then proposed to encourage continuous reflective practice and enhance student experience.

1 Introduction

In 2008 a model curriculum for an IS degree based in a Computer Science department was put forward for Heriot-Watt University (HWU) by a working panel of teaching staff; Coady and Pooley [7] detailed that proposal. This paper considers

J. Coady (✉) • T. Berg • R. Pooley
Department of Computing Science, Heriot-Watt University,
Riccarton Campus, Edinburgh, EH14 4AS, UK
e-mail: J.Coady@hw.ac.uk; tb79@hw.ac.uk; R.J.Pooley@hw.ac.uk

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_17,

how effective the Information Systems degree based on that proposal, running at Heriot-Watt University, has been, from three perspectives:

1. How successful have students graduating from the degree been in finding relevant employment?
2. How is the degree seen by students who have graduated from it?
3. What are the views of the student body on the current programme?

In this evaluation, the team has maintained its practice of benchmarking its syllabus against other proposals for IS degrees. How far such proposals matched experience at Heriot-Watt was of central interest.

In this paper Sect. 2 reviews related work and sets the scene for the studies reported here. Section 3 describes how the Heriot-Watt IS degree was created. Section 4 investigates empirically graduate employability. Section 5 looks at past student satisfaction using a qualitative study on a social networking site for alumni. It also draws on a study of the current degree, to examine qualitatively student attitudes. Section 6 compares the existing status and the IS 2010 curriculum. The results are discussed and proposals for future IS degrees suggested in Sect. 7.

The overall results are consistent and interesting, although it is inevitable that sample sizes are too small for much meaningful quantitative analysis.

2 Background

What to teach often drives debate around professionally oriented programmes. Nearly two decades ago the lessons of many IS system development failures had led to emergence of numerous IS methodologies, which were then taught by universities and subsequently used more widely in industry, Walters et al. [26]. Although researchers have been working to improve the effectiveness of information systems, Stapleton [20] suggested that their successful provision has remained elusive. This continues, with no approach dominant, Stapleton and Murphy [21]. It has been suggested that ISD failures can be attributed to the narrow perspective of the analyst and the need for a more authentic view of ways in which people in organisations process information and make decisions.

Employability has been a second related concern among writers examining IS courses and syllabi. Gupta and Wachter [11] suggested that there is a need for academia to monitor industry continually to ensure that graduates can acquire the necessary mix of skills to compete in the job market. Independently, Abraham et al. [1] suggested that IS programmes from business schools are well positioned to equip students with the skills and capabilities required by professionals in Information Technology (IT) departments. They suggested that business schools have an advantage over alternative schools, such as Computer Science and Information Studies, since they cover the functional areas and management skills that IT executives in non-IT organisations need. This was disputed by Coady and Pooley [7], arguing that, by working within a Computer Science School, with the cooperation of a School of

Management and Languages, an improved, more relevant, IS programme could be created, thereby attracting technically aware students, despite falling overall recruitment to Computer Science, whilst creating more broadly capable, employable IS graduates.

There have been a number of attempts to define standard IS syllabi and curricula. The IS 97 report, Gorgone et al. [9], brought together interested parties to define a syllabus for both business and computing interests. The IS 2002 report of Gorgone et al. [10], built on IS 97, set out to address the mix of technical and sociotechnical subjects; but there was much disagreement on what is core and how to equip students for the real world, Turner and Lowry [24]. Carlsson et al. [5] created a Business IS Design curriculum, BISD 2007, based on the [8] and their own Dublin descriptors. This led Coady and Pooley [7] to question how an IS degree could be implemented effectively within Heriot-Watt University.

Landry et al. [14] suggest that a graduate of an IS programme should be equipped to function adequately at an entry-level position within industry and have a good basis for career development. Programmes need to improve to attract students to address shortages in the workplace and deliver evolving skills needed by industry. Snoke et al. [19] report anecdotal evidence from IS industry interviews suggesting tertiary curricula in IS do not meet the needs of industry. Lynch and Fisher [16] build on this idea to suggest that various studies, Von Hellens et al. [25] and Toleman et al. [23], show that just as IS practitioners are unhappy with IS graduates' skills, so IT practitioners are not satisfied with current, more technically focused IT graduates, who are often seen to be lacking the aptitude and skills to work effectively in collaborative teams.

3 Developing the Current Curriculum at Heriot-Watt

There was agreement by team at Heriot-Watt that the IS curriculum should have a combination of technical and sociotechnical subjects. The programme was built upon the IS 2002 report and BISD 2007 curriculum, above, as most relevant, defining characteristics of the IS profession relatively constant over time and integrated into the curriculum. These are that IS professionals must:

1. Have a broad business and real-world perspective
2. Have strong analytical and critical thinking skills
3. Have interpersonal and team skills with strong ethical principles
4. Design/implement IT solutions to enhance organisations' performance

BISD 2007, Carlsson et al. [5], justified its approach as diverging from IS 2002 due to a need to emphasise capabilities of graduates and not merely a course-oriented approach. This matches the trend in higher education to create graduate attributes, as well as specific knowledge, Barrie [2].

The approach taken by the IS 2008 curriculum for Heriot-Watt was towards the idea from [15] that a revision of the IS curriculum should adopt a more

Table 1 Actual course structure 2008

Stage	Semester 1	Basis	Semester 2	Basis
1	Software development	Technical	Introduction to computer systems	Technical
	Emerging technologies	Applied technology	Technology in society	Sociotechnical
	Praxis	Generic	Introduction to databases	Technical
	Interactive systems	Human centred	Management and enterprise	Organisational
2	Interaction design	Human centred	Software design	Technical
	Fundamentals of marketing	Organisational	Creative design (project)	Application
	Internet technology	Applied technology	Database management	Applied technology
	Organisational behaviour	Organisational	Operations management	Organisational
3	Software engineering (group project)	Application	Software engineering (group project)	Application
	Knowledge management	Organisational	Research methods	Generic
	Critical thinking	Generic	Soft systems	Sociotechnical
	Business policy	Organisational	International strategic management	Organisational

student-centred/active learning approach to achieve a better calibre of graduate for the workplace. There is an emerging consensus that the current role of the academic has shifted from knowledge giver to facilitator guiding active learners, encouraging students to be self-directed learners, preparing them for more realistic situations and developing more soft skills not taught by traditional approaches.

The curriculum originally developed was a reasonable compromise, given the practical decisions necessary, but an ideal IS course remains elusive. The structure of its core 3 years is in Table 1. Stage 4 is made up of advanced options (60 %) and an individual project (40 %).

The modules were categorised by their role in the curriculum or 'Basis'. The original ideal curriculum had four categories of specialism; however, in dealing with practical constraints such as resources and teaching loads, we ended with seven: Generic, 3 modules; Technical, 2 modules; Applied Technology, 3 modules; Sociotechnical, 2 modules; Human centred, 2 modules; Organisational, 7 modules; and Application, 3 modules. There were three constraints forcing these compromises.

1. Resources: Limited teaching resources were available.
2. Cultural bias: Reluctance of some colleagues to accept subjects can be seen from viewpoints other than technical or mathematical.
3. Reciprocal lack of understanding from business and management-based colleagues misses the view that IS has individual, organisational and technical views on system management and development.

Coady and Pooley [7] claimed that Information Systems is a distinct discipline, using information and theories, from, amongst others, Computer Science and Management and Sociology, and should arrive at a new synthesis to match concerns with effective and humane organisations. It should resist being a sub-discipline and, instead, make explicit its unique viewpoint. We have, in the work reviewed here and in the new curricula being developed in many places, entered into a debate which we hope will be informed by our successes and failings. Furthermore, as a community there is a need to continue the debate in a constructive way, leading to new IS professionals with the attributes sought by employers and society.

Strongly argued in Coady and Pooley [7] was the belief in IS 2002 that good teaching staff are vital to the strength of an IS programme. IS 2002 further asserts that to maintain successful development and delivery of the curriculum, the interests and qualifications of staff must allow the course and curriculum not only to be taught but to be modified and changed. This was the core of the team's approach as the programme developed.

To supply industry with workers with relevant skills and ensure greater uptake of courses within the university, addressing the current generation of applicants, the IS degree programme, Coady and Pooley [7], aimed for both a greater technological emphasis and a wider analysis of needs of organisations, including those outside the commercial sector. This was run in a CS department, where technological and sociotechnical aspects can be managed in greater detail. At the same time, working closely with the University's School of Management, we sought to provide the knowledge of organisations to produce graduates sought after by employers.

A conclusion was that words such as 'sociotechnical' and 'technosocial' mean little. The bolder claim was that IS is a distinct discipline and, as a community, we need to continue a constructive debate, leading to a new generation of IS professionals with these wider attributes. The key in moving forward is that any new IS degree continues to address these aims.

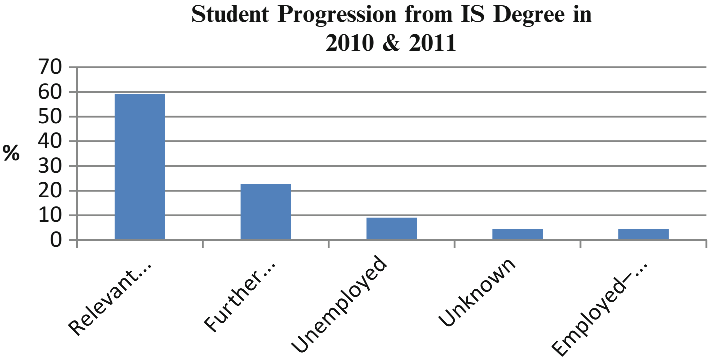
4 Employability

We consider empirical data concerning the employability of our graduates over recent years. Although the numbers are low, we note a consistent trend in our observations.

The motivation for this is to reflect on the improvements suggested in [7] for a curriculum model for an IS degree. Studies were conducted in March and September 2011 gathering employment statistics for the IS graduates of Heriot-Watt University for 2010 and 2011. This was born from both curiosity and a desire to understand the roles IS graduates were being employed in. These students were chosen as they had been a part of the new IS degree for at least 2 years. The studies sought post-degree information from the students, to determine graduate positioning, e.g. who went onto further education and who gained relevant employment, i.e. in work using the specific skills from their IS education.

Table 2 shows the results from 2010 to 2011. Of 22 graduates, 81.8 % (18/22) went onto further education or relevant employment, i.e. graduate or IT-/IS-related positions. One was employed in a non-related IS position, bringing the total

Table 2 Student progression data from IS degree 2010 and 2011



employed graduates to 86.3 % (19/22) overall, 9 % (2/22) had failed to find employment, and 4.5 % (1/22) were uncontactable.

These impressive graduate employment statistics are given to new IS students to make a better informed programme choice. Hence, it must be ensured that relevant and correct information is given, whilst maintaining a fresh and contemporary approach to the curriculum.

5 Analysis of Student Opinion

5.1 Past Student Satisfaction Using Social Networks

The gathering of information from graduate students was conducted using a social network site (SNS) to actively encourage reflective discussion and debate. It was decided that this knowledge-gathering platform would be less formal than a standard focus group or one-to-one interviews and considerably less restrictive than questionnaires. The qualitative data was collected in an action research environment by creating a virtual focus group, with the authors immersing themselves within a reflective forum and encouraging discussion. This style of focus group has opened up new areas for consideration and debate and has therefore been seen as a positive reflective exercise. The IS working panel will consider continuing such an approach as a permanent knowledge-gathering tool for future graduates.

Redmond [18] suggests that SNS offers new ways for researchers to run surveys quickly, cheaply and single-handedly. She concludes that Facebook is currently an ideal SNS for survey research, thanks to size, intensive use and continuing growth. Each Facebook user is directly linked to his or her personal ‘friends’. Brickman-Bhutta [4] proclaims that those to ‘whom we are weakly tied are more likely to move in different circles from our own and thus will have access to information different from that which we receive'. She defines weak ties as the bridges between small clusters of close friends, linking us together to form an elaborate web of social relationships, and infers that Facebook offers researchers a way to capitalise on the strength of these weak ties. According to Facebook's internal statistics, the average user has 120 friends (Facebook 2009). Whilst some of these relationships constitute 'strong ties', the vast majority are acquaintances and old friends from high school or college we rarely see; hence, it was decided that through this medium graduated students could be targeted for this study and input. Researchers can readily sample populations of interest by working through existing groups or creating new ones [4]. Facebook groups and events are virtual communities linking people with some shared interest, attribute or cause. An event for IS graduates of 2010 and 2011 was created and a discussion by the virtual focus group ensued.

The virtual focus group suggested a necessity for changes, some of which align to the IS 2010 recommendations. However, others, such as the removal of application development, were not supported. The graduates considered this to be of prime importance to the curriculum. Recurring themes and responses suggested by graduates imply that employers require IS graduates to have specialisations in areas such as HTML, .NET, SQL and Java. Some respondents went as far as suggesting a deeper exploration of these topic areas be introduced in order for them to gain a more expert knowledge in specific areas.

This paper agrees with the IS 2010 guidelines to further encourage the inclusion of both enterprise architecture and IT infrastructure. The focus group commented encouragingly on the sociotechnical aspects of our degree:

> The mix of socio/techy stuff was good, I like the option of being able to dodge the techy stuff while others liked being able to pick more of it.

The group suggested that courses, such as databases, information systems methodologies and knowledge management, gave them greater insight into large organisations and their workings, and the design and planning processes behind them.

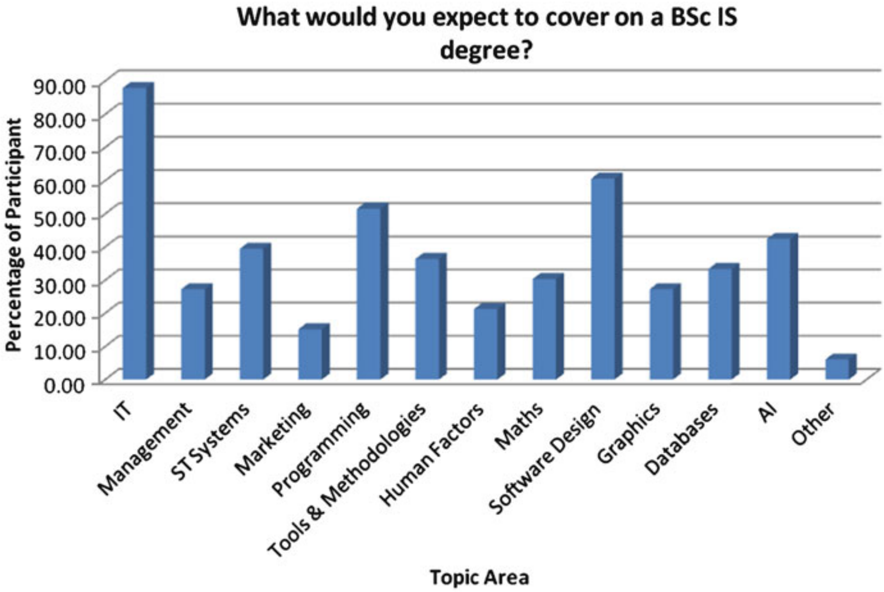
A major element that was seen to be an emerging theme within the focus group was the need to include a project management course. One graduate commented:

> This degree is a great degree and there is no way I would have ended up as a project manager at the age of 23 if I had done strictly computing science.

However, this participant did add to this statement by suggesting the need for specific project management teaching. The majority of the focus group suggested that project management would be well suited to the course with some suggesting that assurance was needed that it would have a sound pedagogical approach involving real-life examples from industry. The need for training on project management software was recommended by the group as an imperative skill for a graduate.

It was noted that the group disagreed with the IS 2010 suggestion to remove the personal productivity tools course (Praxis). There was a strong case made by at least three of the participants for a stronger grounding in Excel/Statistical Processing methods.

Table 3 Student expectation of an IS degree



5.2 *Opinion of Potential Students*

The review of the HWU IS degree used the results of [17], who created a Soft Model of the HWU IS programme. A survey of 31 potential IS students was conducted, based on student expectation of the IS degree. Results, shown in Table 3, suggest there is a general lack of knowledge of what subject areas are typically studied as part of an IS degree. A high percentage expected topics that do not feature in the IS degree, for example, 42 % of participants expected to cover AI which is not part of the IS course. Only 15 % of the participants are expected to cover Marketing which is an important component of the IS course. This further strengthened the case for a review to ensure the correct components were being included in the HWU IS programme.

As in IS 2010, university-level IS curricula need frequent updating to remain effective as all aspects of global computing face rapid and frequent change. As the 2008 programme developed and has gone through a 2-year period, it was felt appropriate to review it to ensure the approach taken was pedagogically sound, in the best interest of the students, and meets industry requirements. This ensures students are gaining the best experience and deemed relevant for differing industry domains

Table 4 2010 curriculum as developed by the IS working panel

Stage/sem				
1/1	Software development 1	Praxis	Interactive systems	Elective 1
1/2	Introduction to computer systems	Web design and databases	Technology in society	Elective 2
2/1	Interaction design	Fundamentals of marketing	Internet and communications	Enterprise and its business environment 1
2/2	Software design	Enterprise and business environment 2	Creative design (project)	Database management
3/1	Software engineering 1 (group project)	Organisational behaviour	Critical and computational thinking	Project management
3/2	Software engineering 2 (group project)	Knowledge management	Sociotechnical and soft systems	Human resource management

6 Comparisons of Existing State and IS 2010 Curriculum

It was stated in IS 2010 that the IS 2002 curriculum had taken a 'one size fits all' philosophy, whereby there is no separate core specified within the curriculum. In essence, all courses are required. The proposal in Coady and Pooley [7] agreed that this model left little room for the specialisms, constraints and innovation in their institution.

One of the key issues pushing this review forward was the results from the UK National Student Survey (NSS) and the university ideals on student employability and transferable skills. A proposal by Coady and Jeyaseelan [6] suggested that employers for graduates were looking for a number of key skills as detailed by leading authors in the field. These were:

- Career Management Skills and Effective Learning [13]
- Graduates with knowledge, intellect and willingness [12]
- Work under pressure, oral communication skills, accuracy, attention to detail, team work, time management [3].

These were inbuilt to the Coady and Pooley [7] curriculum and continue as an integral part of the current 2010 review. The 2010 curriculum as developed by the IS working panel is detailed in Table 4 above.

IS 2010 identified a set of seven core courses common to all Information Systems programmes:

1. Foundations of Information Systems
2. Data and Information Management
3. Enterprise Architecture
4. IT Infrastructure
5. IS Project Management
6. Systems Analysis and Design
7. IS Strategy, Management and Acquisition

IS 2010 indicates that these seven courses need not be delivered independently. This underpinned restructuring of the HWU IS programme, as the team could link the underpinning core topics to the areas covered by the courses delivered in the HWU IS programme. A course on project management was added to the syllabus, as requested by the focus group (Sect. 6). This was encouraged by SML which provides the module. Restructuring within their department affected the courses they had available. Other courses had been reviewed using feedback from various years to ensure a coherent syllabus for IS students.

IS 2010 suggests four core course changes. Whilst the team tended to agree with the suggestions for inclusion of both enterprise architecture and IT infrastructure, along with the positive responses from participants, some of their suggestions might not necessarily benefit our student population.

IS 2010 removes application development from the prescribed core, though not from the IS Programme. It suggests it as an elective, but through student feedback it has been suggested that Programming, whilst not easy or sometimes enjoyable, is useful now that they are working in industry and as such should be core to the syllabus.

IS 2010 recommends dropping personal productivity tools courses from IS programmes. The HWU IS review panel agree that most students entering university are proficient in personal productivity applications such as word processing, spreadsheets and presentation software. The working panel considered the Praxis course in year 1 sufficient for university-specific skills, such as use of libraries, virtual learning tools, academic referencing, undertaking self-directed study and personal development planning. Whilst other productivity-based courses such as Critical Thinking were also praised, there was a reflection of the calibre of lecturer being extremely enthusiastic and engaging, which ties back to the assertion made by Coady and Pooley [7], based on IS 2002, that good teaching staff are vital to the strength of an IS Programme.

IS 2002 suggests a sequenced approach which was adopted by the curriculum in Coady and Pooley [7]. The current IS 2010 approach proposes a flattened curriculum; however, it is the belief of the HWU IS team that this is not suitable. Students need a certain amount of skill and competency in specific areas in order to progress successfully through their degree.

This study has shown that IS 2010s approach is suitable for certain aspects of the HWU student population. Like most suggested frameworks there needs to be a tailoring approach to meet the specific circumstances. The results of this tailoring and methods of improvement shall be discussed in the concluding section.

7 Conclusions and Further Work

The objective of this research was to determine to what extent the HWU IS team experience supports IS 2010 and to what extent they conflict with it. This was discussed in Sect. 6, drawing comparisons with the new elements of the IS 2010 curriculum for both the HWU IS 2008 programme and the new 2010 proposed programme. It then looked at what proposed changes should be implemented moving forward, based on the reflection of the virtual focus group and the HWU IS team's experience.

It is the conclusion of this paper that no 'one size fits all' IS curriculum. Whilst IS 2010 proposes a solid foundation of Core and Choice courses, there still needs to be a tailoring approach taken by institutions in order to meet their specific requirements, budgetary issues and those specialisms peculiar to individual department(s). Coady and Pooley [7] proposed that the IS curriculum developed by an Institution may often consult local industry to ensure relevancy of graduates for the work force. This idea has been reiterated by Stefanidis and Fitzgerald [22] who suggest that academics are charged with a responsibility of designing courses which are industry relevant whilst ensuring a strong pedagogy is embedded in the curriculum.

The results of the virtual focus group reiterate the ideals set out in IS 2002 and implemented in the HWU IS 2008 that quality teaching using creative pedagogic approaches is imperative for the student experience. It is necessary to adopt a more student-centred/active learning approach as suggested by Turner and Lowry (2005) to ensure a better calibre of graduate. Graduate feedback highlighted the need for more real-world examples by which will strengthen the Programme. The core topic of project management as set out by IS 2010, whilst lacking in the original 2008 HWU IS programme, has now been included in the 2010 revision of the programme, backed by graduate discussion.

The results of graduate destinations as shown in Table are impressive for future applicants. We suggest there should be more emphasis on IS graduate alumni and encourage the continuous tracking of graduate destinations in the future.

There is a limited sample size since the Programme has only been running for 2 years (at the time of writing this paper, year 3 cohort results are being collected), so future work from this research will collect graduate destinations at least until 2013, as this would give an overall view of graduates who progressed through the degree from the original programme from years 1 to 4. It would also enable data to be gathered about student numbers, both graduating and intake, whilst ensuring a consistent reflective approach which will ultimately enhance the student experience.

References

1. Abraham T, Beath C, Bullen C, Gallagher K, Goles T, Kaiser K, Simon J (2006) IT workforce trends: implications for IS programmes. Commun Assoc Inform Syst 17:1147–1170
2. Barrie SC (2007) A conceptual framework for the teaching and learning of graduate attributes. Stud Higher Educ 32(4):439–458
3. Brennan J, Johnstone B, Little B, Shah T, Woodley A (2001) The employment of UK graduates: comparisons with Europe and Japan. The Higher Education Funding Council for England, London
4. Brickman- Bhutta C (2009) Not by the book, facebook as a sampling framework. http://www.thearda.com/workingpapers/download/Not%20by%20the%20Book%20-%20Bhutta.doc.-last. Accessed May 5 2012

5. Carlsson S, Hedman J, Steen O (2007) Model curriculum for a bachelor of science programme in business information systems design: organisational impacts. In the proceedings of the 16th international conference on information systems development (ISD2007), NUI Galway, Ireland
6. Coady J, Jeyaseelan J (2004) Employability: theory & practice proceedings of the learning and teaching conference, Heriot-Watt University, Edinburgh
7. Coady J, Pooley R (2008) IS degrees—sociotechnical or technosocial? In: Proceedings of the 16th international conference on information systems development, Paphos, Cyprus, Springer LNCS
8. Education Ministers European Joint declaration of the European Ministers of Education convened in Bologna on the 19th of June 1999
9. Gorgone J, Gordon Davis J, Cougar D, Feinstein DL, Herbert E (1997) Model undergraduate curriculum guidelines for IS programs. SIGCSE Bull Inroads 29(4):209–213
10. Gorgone J, Davis G, Valcich JS, Topi H, Feinstein D, Longenecker H, Jr. (2002) IS 2002 model curriculum and guidelines for undergraduate degree programmes in information systems, Association for Computing Machinery, Association for Information Systems, Association for Information Technology Professionals
11. Gupta JND, Wachter RM (1998) A Capstone course in the information systems curriculum. Int J Inform Manage 18(6):427–441
12. Harvey L, Moon S, Geall V, Bower R (1997). Graduates work: organisation change and students entrepreneuring in education ERIC. Institute for Education Sciences, London, UK
13. Hawkins P, Winter J (1998) Mastering change. Department for Education and Employment, Sheffield, UK
14. Landry J, Longenecker H, Haigood B, Feinstein D (2000) Comparing entry-level skill depths across information systems job types: perceptions of is faculty. In: Proceedings of sixth Americas conference on information systems, Long Beach, CA
15. Lowry G, Turner R (2005). Information Systems Education for the 21st Century: Aligning Curriculum Content and Delivery with Professional Workplace. In Carbonara, D. (Ed) Technology Literacy Applications in Learning Environments, Information Science Publishing, Hershey PA, pp171–202
16. Lynch K, Fisher J (2005) Tomorrows workforce today: what is required by information systems graduates to work in a collaborative information systems workplace. Book Chapter XV in Information Systems and Technology Education: From University to the Workplace. IGI Global pp 311–326 ISBN-10: 1599041146
17. McInnes A (2010). Soft model of the information systems degree. Undergraduate degree thesis, HeriotWatt Library, p 26
18. Redmond F (2010) Social networking sites: evaluating and investigating their use in academic research. Dissertation, Dublin Institute of Technology
19. Snoke R, Underwood A, Bruce C (2002). An Australian view of generic attributes coverage in undergraduate programmes of study: an information systems case study. In: Proceedings of HERDSA 2002, 590–598
20. Stapleton L (2001) From Information Systems in Social Settings To Information Systems as Social Settings: A Study of Large Information Systems in Manufacturing Firms, IFAC Symposium on Automation Based on Human Skill: Joint Design of Technology and Organisation, Elsevier: North Holland
21. Stapleton L, Murphy C (2002) Examining non-representation in engineering notations: empirical evidence for the ontological incompleteness of the functionally-rational modelling paradigm. In: Proceedings of the IFAC world congress 2002, Elsevier, Amsterdam
22. Stefanidis A, Fitzgerald G (2010) Information systems curricula in the UK: a survey of undergraduate courses. Italics 9(1):9
23. Toleman M, Roberts D, Ryan C (2004) retrofitting generic graduate attributes: a case-study of IS undergraduate programmes, informing science + information technology education joint conference—InSITE 2004, 625–635 (Rockhampton, 25–28 June)

24. Turner R, Lowry G (2003) Education for a technology-based profession: softening the information systems curriculum. Book Chapter XIII) in Issues in Information Systems Education. Tanya McGill, editor. Hershey, USA: IRMA Press. pp 153-172. ISBN 1-931777-53-5
25. Von Hellens L, Wong S, Orr J (2000). IT skills requirements: perspective from industry. In: Proceedings 11th Australasian conference on information systems: ACIS 2000. ACIS, Brisbane (ACIS2000 Proceedings, CD-ROM)
26. Walters SA, Broady JE, Hartley RJ (1994) A review of information systems development methodologies. Library Manage 15(6):5–19

Model-Driven Development of Model Transformations Supporting Traces Generation

Álvaro Jiménez, Juan M. Vara, Verónica A. Bollati, and Esperanza Marcos

Abstract This work introduces a framework for model-driven development of model transformations that support traces generation. The proposal starts from a high-level specification of the relationships that must hold between the elements of source and target metamodels. Such specification is subsequently refined into lower-level transformation models until they can be serialized into the source code that implements the transformation. Running such transformation produces not only the corresponding source models but also a trace model between the elements of source and target models.

1 Introduction

The IEEE [17] defines traceability as "the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship to one another." Traceability management, which has always been acknowledged as a relevant topic in software engineering (SE) [31], implies keeping track of the relationships between the different artifacts of the development process. Therefore, appropriate management of traceability information helps to monitor the evolution of system components. In addition, traceability information can be used to carry out different activities, such as change impact assessment, requirements validation, and maintenance tasks [1,2,40].

Unfortunately, generating and maintaining links among software artifacts is a tedious, time-consuming, and prone-to-error task if no tooling support is provided to that end [28]. However, the advent of model-driven engineering (MDE), in which

Á. Jiménez (✉) • J.M. Vara • V.A. Bollati • E. Marcos
Kybele, Rey Juan Carlos University, C/Tulipán s/n, 28933 Móstoles, Madrid, Spain
e-mail: alvaro.jimenez@urjc.es

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_18,

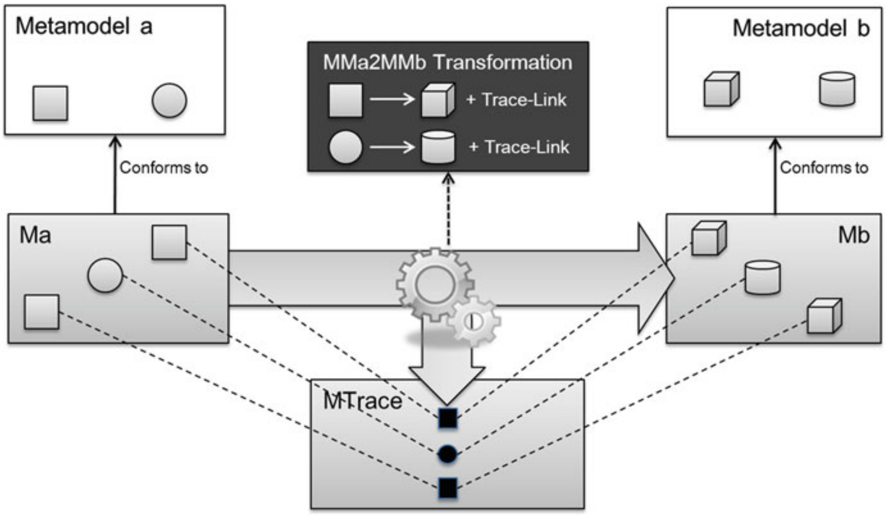


Fig. 1 Using model transformations to support trace generation

principles are to enhance the role of models and to increase the level of automation all along the development process [33], provides a new landscape that can positively influence the management of traceability.

In particular, one of the disciplines that follows the principles of MDE is model-driven software development (MDSD) [36], which proposes to describe the system under development by means of high-level models which are subsequently refined into low-level models until their level of detail is that of the underlying platform. The key to automate each step of the process are the model transformations that connect the different models handled [34]. Simply put, a model transformation defines a set of relationships between the elements of source and target metamodels that must hold between the elements of the models conforming to such metamodels [35]. Implicitly, a model transformation contains valuable information from which trace links (or just traces) can be derived: actually, such links can be seen as instances of the relationships defined at metamodel level. To illustrate this idea, Fig. 1 shows a simple scenario where two models (`Ma` and `Mb`) are connected by a model transformation (`MMa2MMb`) that defines a pair of mapping rules to generate a target model (`Mb`) and a trace model between `Ma` and `Mb` (`MTrace`). In particular, each square will be mapped into a cube plus the corresponding `square2cube` trace, whereas circles are mapped into cylinders plus the corresponding `circle2cylinder` traces.

Therefore, the intensive use of models and model transformations characteristic of any model-based proposal offers an opportunity to increase the level of automation in the management of traceability. In addition, the trace links generated could be collected in trace models that can be subsequently processed by means of MDE techniques, such as model transformation, model matching, or model merging [3].

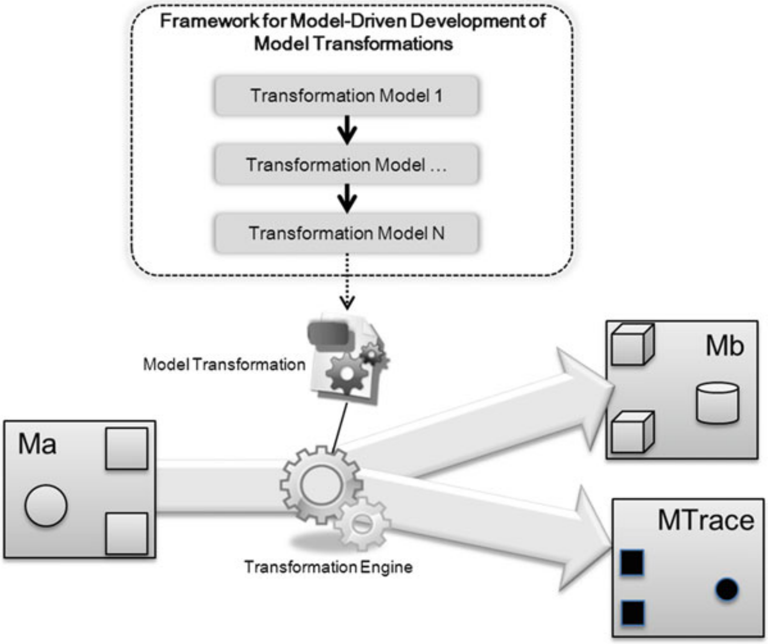


Fig. 2 Model-driven development of model transformations supporting traces generation

Unfortunately, model transformations are not as simple as the one shown in the previous scenario. With the maturity of MDE technologies [39], model-based proposals have started to involve more complex metamodels and thus more complex model transformations. Coding such transformations becomes even more complex if they have to deal with traceability management. We believe that applying MDE principles to the development of model transformations [5,7,16] would alleviate this complexity.

Combining these ideas, this work presents a proposal to support the model-driven development (MDD) of model transformations that include traces generation. The idea is to take a step further the scenario shown in Fig. 1, which is already supported by some model transformation languages [10,24], by applying MDE principles to define a development process that is independent of any model transformation language. Figure 2 illustrates the approach that starts from a high-level specification of the model transformation that is subsequently refined into lower-level transformation models [4] until they can be used to generate the source code that implements the transformation. When executed, such transformation will produce not only the corresponding target models but also a traces model between the elements of the models implied in the transformation.

The rest of this paper is structured as follows: Section 2 analyzes related works on the combination of MDE and traceability management; Section 3 introduces our proposal for model-driven development of model transformations that support traces generation, which is illustrated by means of a case study in Sect. 4. Finally, Sect. 5 concludes by summarizing the main contributions of this work and providing directions for further work.

2 Related Works

So far, there are some model-based proposals to deal with traceability in order to keep the relationships between different software artifacts. NDT-Quality [13] is one of such proposals. It provides a traceability matrix to check that the requirements are satisfied by the artifacts produced at the different stages of the process. In this paper however we focus on the generation of traces limited to the context of model-driven development of model transformations.

Although there exist some proposals focused on the model-driven development of model transformations [5,7,16,25,26,38] and some others which use transformations to generate traces [8,21,27,30], we have found no proposal combining both approaches, according to the idea illustrated in Fig. 2. Therefore, in the following we provide an overview of related works grouped into these two main categories.

The first group collects proposals for MDD of model transformations. These proposals argue in favor of using models to define high-level specifications of model transformations that are independent of any model transformation language. Such specifications are later translated into low-level specifications, directly executable or interpretable. Besides, although they consider different levels of abstraction, all of them consider at least some kind of platform-independent and platform-dependent level. More concretely, while Kusel [25] and Vignaga [38] consider desirable to automate the transition between different abstraction levels, Bézivin et al. [5] and Guerra et al. [16] point to model transformations as the way to do so. The latter goes a step further proposing to generate code from low-level transformation models. Finally, it is worth noting that among all the proposals found focused on MDD of model transformations, just Guerra et al. [16] and Kusel [25] provide with some technical support (limited to partial implementations in both cases). Likewise, none of such proposals deal with traceability management, as we have already mentioned.

On the other hand, there is a number of works that follow the idea depicted in Fig. 1, using model transformations for trace generation purposes. Probably the most relevant is the proposal from Jouault [21] that takes advantage from MDE principles to support traceability management. The idea is to extract transformation models from existing model transformations, enrich them with the machinery needed for traces generation, and generate back an enriched model transformation. Nevertheless, the proposal works only for existing ATL model transformations, and it does not consider the development process of such transformations. Another work in this group is the proposal of Levendovszky et al. [27], which defines explicit traceability relationships into the transformation rules. This way, when the transformation is executed, such relationships are instantiated as trace links. There are some other works, such as those from Bondé et al. [8] and Oldevik and Neple [30], that make explicit the implicit traceability relationships that exist in any model transformation. The main issue with these proposals is that they were devised for a specific transformation language/engine. Therefore, its application to other languages or approaches can be too complex or just unfeasible. Moreover, these proposals serve to reflect again the gap between theory and practice since just Jouault [21] accompanies the methodological proposal by complete tool support.

Finally, we would like to mention apart two relevant works that deal not only with traces generation but with model management in general. In the first one, Boronat et al. [9] present the support for traceability generation embedded in MOMENT, a model management framework implemented atop of Eclipse and EMF [14]. In this framework, model operators are defined at metamodel level to carry out different model management tasks and trace models can be obtained, either automatically by the execution of these operators or manually by the user.

In the second one, Guerra et al. [15] propose a pattern-based approach to define inter-modeling specifications and to solve different MDE scenarios, such as model traceability. In this work, the authors also present PAMOMO, an Eclipse tool to support their proposal.

In contrast with existing works, our proposal does not depend on a particular model transformation language since it combines the model-driven development of model transformations with the generation of trace models. This way, high-level transformation models are subsequently transformed into low-level ones until they can be used to generate the source code that implements the transformation modeled. Such transformation generates not only the corresponding target models but also a trace model between the elements of source and target models. Besides, a complete EMF-based IDE supports the proposal [18].

3 MDD of Model Transformations Supporting Traces Generation

In [7,19] we introduced a first prototype of a framework to apply MDE to the development of model transformations. In the following we introduce MeTAGeM-Trace, an evolution of such framework that supports the MDD of model transformations that include traces generation. To that end, we first describe its methodological proposal to later introduce its technical support.

3.1 Methodological Proposal

The model-driven development process for model transformations defined by MeTAGeM-Trace is illustrated in Fig. 3. The underlying idea is that the model transformation to develop is modeled at different abstraction levels. Concretely, we consider four different levels: PIM (platform-independent models), PSM (platform-specific models), PDM (platform-dependent models), and code.

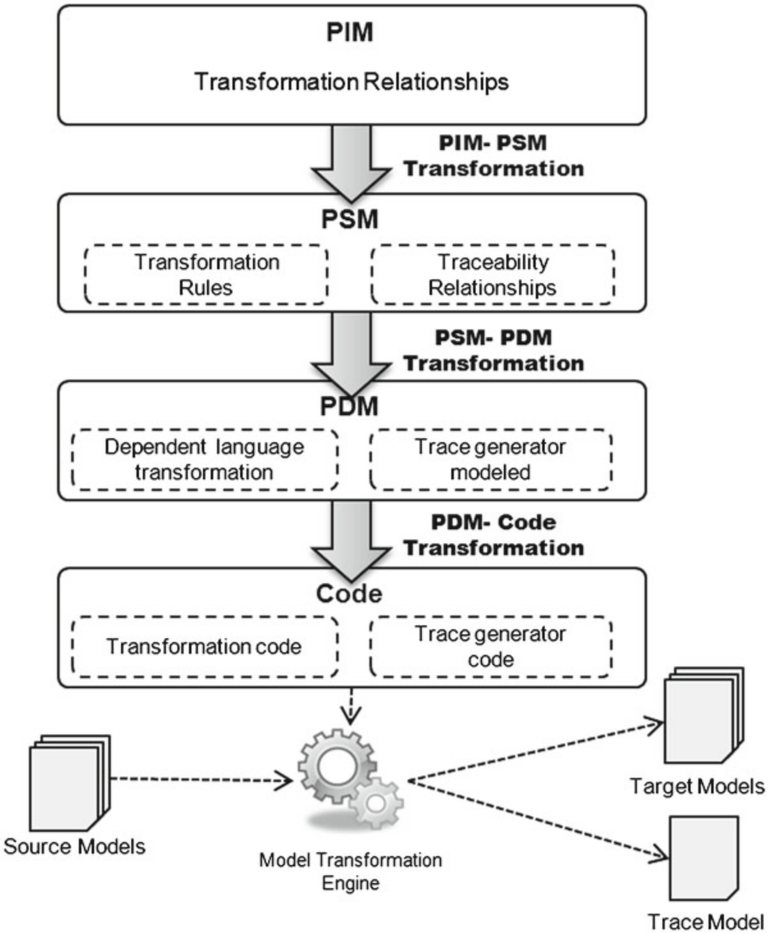


Fig. 3 MeTAGeM-Trace development process

First, a PIM transformation model defines the relationships between the elements of source and target models, without considering any implementation detail. Such model is consumed by a model transformation (`PIM2PSM`) that produces a new one that models transformation rules in terms of a particular approach for model transformation development, namely, imperative, declarative, hybrid, and graph-oriented [11]. Moreover, the traceability relationships that were implicit in the PIM transformation model become explicit at PSM level.

The next step of the process is supported by the `PSM2PDM` model transformation that consumes the previous PSM to produce a PDM. The generated transformation model specifies the transformation rules and the components needed to create trace models (taking into account the traceability relationships specified at PSM level) in terms of a concrete transformation language. Note that such language must be one of those following the programming paradigm selected at PSM level, i.e., if we have chosen the hybrid approach at PSM level, the PDM must be described in terms of a hybrid language such as ATL [22], ETL [23], or RubyTL [32].

Finally, a model-to-text transformation (`PDM2Code`) generates the source code that implements the modeled transformation in the selected language. When such transformation is executed, it generates not only the corresponding target models but also a traces model between the elements of source and target models.

Besides, the different transformation models generated along the process can be manually refined. Likewise, the source code generated can be modified using the IDE of the targeted model transformation language.

3.2 Technical Support

In order to provide with a proof of concept for the above-described proposal, we have developed an EMF-based toolkit that supports the MDD of model transformations that include traces generation [18]. In particular, among the different approaches and languages that can be adopted to develop model transformations, we have considered the hybrid approach at PSM level and the ATL and ETL languages at PDM level. Note that we aimed at showing the applicability of the proposal. Therefore, we find it more convenient to do it for the hybrid approach since we find it to be one of the most commonly adopted [11]. Likewise, we opted for the ATL language due to it being considered as the de facto standard for model transformation development [6,37]. In addition, we wanted to show that we were able to generate model transformation for different model transformation languages; thus, apart from supporting ATL, we needed to show the application of the proposal for another model transformation language. Since existing languages do not use to be metamodel-based, i.e., they do not provide a metamodel capturing its abstract syntax that allows defining transformation models, we opted for ETL because it is very well documented. This fact helped decisively on the task of defining its underlying metamodel. Besides, it will probably favor to increase the level of adoption of the language in the next future.

This way, current version of MeTAGeM-Trace supports the (semi)automatic development of ATL and ETL model transformations that include traces generation. To that end, the toolkit integrates a set of EMF-based DSLs [14] to support the modeling of model transformations at different abstraction levels. Namely, it bundles a DSL to model transformations at PIM level, a DSL to model transformations at PSM level according to the hybrid approach, and finally a pair of DSLs to model ATL and ETL model transformations. Besides, MeTAGeM-Trace includes a set of ATL [22] and MOFScript [29] model transformations to bridge such DSLs and automate the different steps of the process.

For the sake of space, we cannot go into much more detail about the technical support for the proposal. However, the next section illustrates some of the software artifacts just mentioned. As well, the toolkit is available for download at http://www.kybele.urjc.es/members/ajimenez/Thesis/.

4 Case Study

In order to illustrate our proposal, this section presents its application to a simple yet complete case study. We have opted for using such scenario in order to avoid accidental complexity. However, for those interested in showing the application of the proposal to more complex scenarios, the toolkit that can be downloaded from the abovementioned URL includes a project in which MeTAGeM-Trace is used to develop an ATL model transformation to map UML conceptual data models into XML schemas.

This way, here we use MeTAGeM-Trace to develop the `Families2Persons` transformation, one of the learning examples provided in the ATL Web site. The execution of the resulting transformation will produce not only a `Persons` model but also a `Families2Persons` trace model. The input and target metamodels for such transformation are shown in Fig. 4.

To develop this model transformation with MeTAGeM-Trace, we follow the process described in Fig. 3. In the following we will use the mapping of `Mother` objects to `Female` objects as conducting thread.

The first step of the process is to develop a PIM transformation model which specifies the relationships between the metaclasses of source and target metamodels. Figure 5(1) shows an excerpt of such model corresponding to the modeling of the one-to-one relationship `Mother_2_Female`.

Next, we obtain a PSM transformation model where transformation rules are described in terms of the hybrid approach and traceability relationships become explicit. Back to the example, Fig. 5(2) shows that the `Mother_2_Female` transformation rule owns a trace rule object which describes the traceability relationship between `Mother` and `Female` objects. Keep in mind that this generated model can be refined manually by adding, modifying, or removing objects. In this case we have added the `getFatherName()` operation.

The previous model is used as input to generate a transformation model that conforms to the ATL metamodel. So, the PSM transformation rules are mapped into the different types of rules supported by the ATL language (the machinery of the

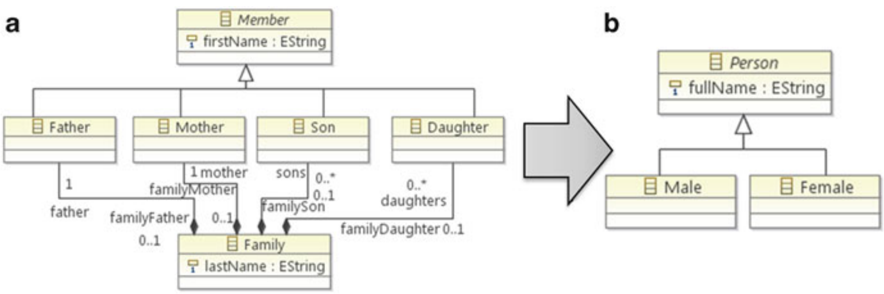


Fig. 4 Metamodels for the case study: (**a**) Families and (**b**) Persons

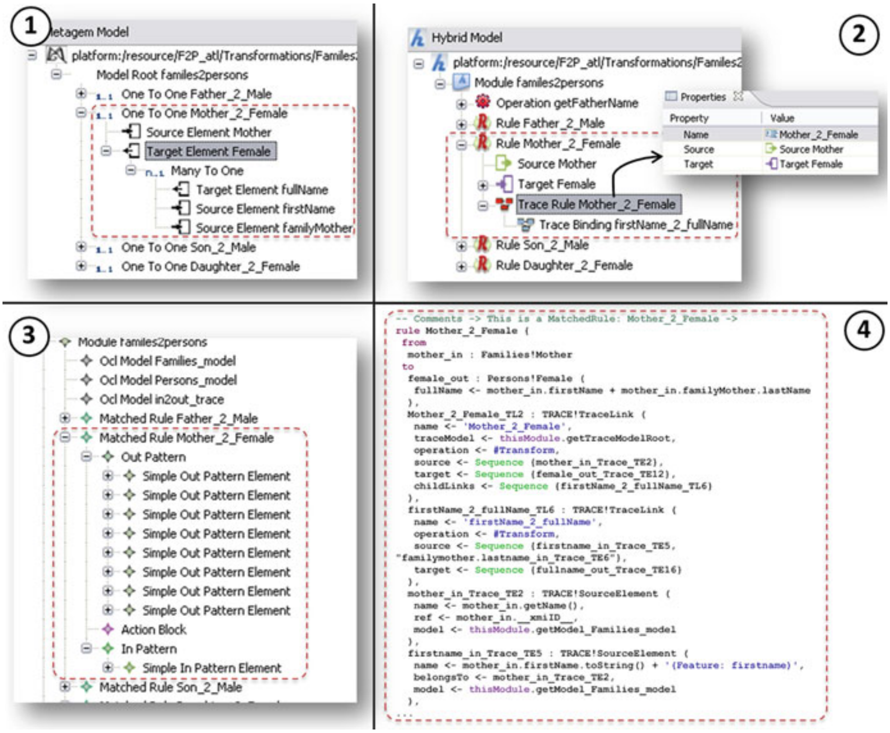


Fig. 5 `Families2Persons` case study: mapping `Mother` objects to `Female` objects

`PSM2PDM` transformation encodes the logics of these mappings). For instance, as Fig. 5(3) shows, the transformation rule `Mother_2_Female` has been mapped into a matched rule. Moreover, since the ATL metamodel does not support the modeling of traceability elements, trace rule objects are mapped into elements of the target pattern of the ATL rules.

Finally, the ATL transformation model is serialized into the ATL code that implements the modeled transformation. For instance, see Fig. 5 (4) that shows the source code of the ATL rule `Mother_2_Female`. Note the extra load of machinery needed to support the generation of traces. This fact serves to back the convenience of raising the level of automation in the development of model transformations. Otherwise, vast amounts of similar code should be replicated all along the transformation.

To conclude, we would like to mention that we have developed another EMF-based DSL for the trace models generated by the model transformations produced with MeTAGeM-Trace, though we have not introduced it here for the sake of space. It includes a multi-panel editor that provides a unified view of the different source and target models and the traces model that connects them [20]. This way, Fig. 6 shows an excerpt of the trace model generated when the user runs the `Families2Persons` transformation developed with MeTAGeM-Trace,

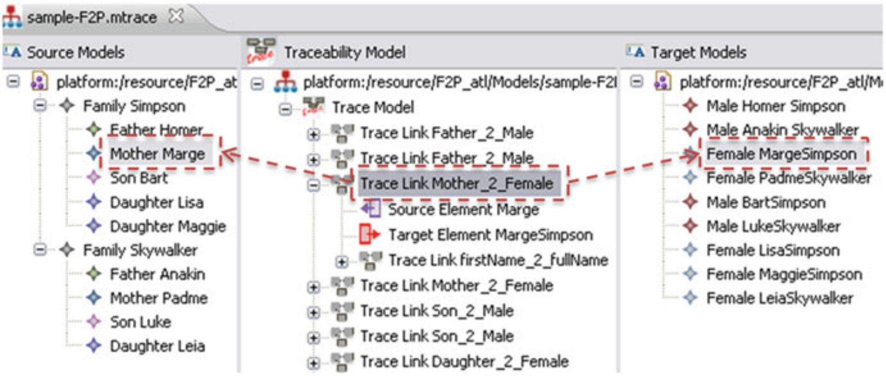


Fig. 6 `Families2Persons` case study: trace model

displayed in such editor. Note that when the `Mother_2_Female` trace link is selected, the source and target objects of such trace (`Marge` and `MargeSimpson`) are automatically highlighted in the corresponding models.

5 Conclusion and Further Work

The key role of models in any MDSD proposal can help to improve the current state of the art on traceability management. Being models the main software assets handled along the development process, traceability information can be reduced to links between the elements of those models. Even better, the traceability relationships that are to be persisted by means of trace links between those models are already implicit in the model transformations that connect them [1].

To take advantage from this scenario, in this work we have introduced MeTAGeM-Trace, a framework for MDD of model transformations that support traces generation. The underlying idea is to apply MDE principles to the development of model transformations [6]: basically modeling the software artifact to develop at different abstraction levels and automating the transition between levels by means of model transformations. The only difference is that in this case the software asset to develop is another model transformation that supports the generation of traces.

It is worth mentioning that the model transformation developer is not completely free of coding tasks when using MeTAGeM-Trace. In our experiments, we have found that some imperative constructions have to be manually added in auxiliary functions. However, a huge amount of the source code is automatically generated. Not only the most repetitive and error-prone parts but also most of the pattern-matching parts, the heart of any model transformation and one of the theoretical advantages of MDE [33], but also most of the pattern-matching parts, the heart of any model transformation. This way, this proposal does not only alleviate the complexity inherent to traceability management but also to model transformation development [12,16].

Finally, this work provides with a number of directions for future work. The most immediate is related to extending the proposal to support other model transformation paradigms such as purely imperative or graph-based [11] (remember that current toolkit supports just the hybrid approach) and other transformation languages, such as the different implementations of the QVT standard [11,24].

The other direction that we are already exploring is how to use the information generated. In some sense, MeTAGeM-Trace can be seen as a way to provide low-level data (all the traces between the elements of the models implied in a given transformation). These data could be aggregated, filtered, or queried in order to provide high-level information. Indeed, being collected in trace models, these data can be processed by means of MDE techniques such as merging, transformation, or weaving [3]. So, we are working to provide mechanisms to analyze traceability data in order to produce high-level information that can be used to support decision-making processes. For instance, the project manager of a MDSD project might be interested in identifying which are the mapping rules generating more trace links in order to improve the performance of a given transformation.

Acknowledgments This research was carried out in the framework of the MASAI project (TIN-2011-22617), financed by the Spanish Ministry of Science and Innovation.

References

1. Aizenbud-Reshef N, Nolan BT, Rubin J, Shaham-Gafni Y (2006) Model traceability. IBM Syst J 45(3):515–526
2. Anquetil N, Kulesza U, Mitschke R, Moreira A, Royer J-C, Rummler A, Sousa A (2010) A model-driven traceability framework for software product lines. Software Syst Model 9(4):427–451
3. Bernstein P (2003) Applying model management to classical meta data problems. In: First Biennial conference on innovative data systems research, USA
4. Bézivin J, Farcet N, Jézéquel J-M, Langlois B, Pollet D (2003) Reflective model driven engineering. In: Stevens P, Whittle J, Booch G, (eds) UML 2003—the unified modeling language. modeling languages and applications, vol. 2863. Lect Notes Comput Sci, Springer, Berlin, pp 175–189
5. Bézivin J, Büttner F, Gogolla M, Jouault F, Kurtev I, Lindow A (2006) Model transformations? Transformation models! In: Czarnecki K, Ober I, Bruel J-M, Uhl A, Völter M (eds) Model driven engineering languages and systems, vol. 4199. Lect Notes Comput Sci, Springer, Berlin, pp 440–453
6. Bohlen M (2006) QVT und Multi-metamodell-transformationen in MDA. In: OBJEKTspektrum, vol. 2
7. Bollati VA (2011) MeTAGeM: a framework for model-driven development of model transformations. PhD thesis, Universidad Rey Juan Carlos
8. Bonde L, Boulet P, Dekeyser JL (2006) Traceability and interoperability at different levels of abstraction in model-driven engineering. Applications of specification and design languages for SoCs: selected papers from FDL 2005, 263
9. Boronat A, Carsí J, Ramos I (2005) Automatic support for traceability in a generic model management framework. In: 1st European conference on model-driven architecture foundations and applications (ECMDA-FA'05), Nuremberg, Germany

10. Boronat A, Carsí J, Ramos I (2006) Algebraic specification of a model transformation engine. In: Fundamental approaches to software engineering, vol. 3922. Lect Notes Comput Sci, Springer, Berlin, pp 262–277
11. Czarnecki K, Helsen S (2006) Feature-based survey of model transformation approaches. IBM Syst J 45(3):621–645
12. Didonet Del Fabro M, Valduriez P (2009) Towards the efficient development of model transformations using model weaving and matching transformations. Software Syst Model 8(3):305–324
13. Escalona MJ, Gutiérrez JJ, Pérez-Pérez M, Molina A, Domínguez-Mayo E, Domínguez-Mayo FJ (2011) Measuring the quality of model-driven projects with NDT-Quality. In: Pokorny J, Repa V, Richta K, Wojtkowski W, Linger H, Barry C, Lang M (eds) Information systems development, Springer, New York, pp 307–317
14. Gronback RC (2009) Eclipse modeling project: a domain-specific language toolkit. Addison-Wesley Professional, Boston, USA
15. Guerra E, de Lara J, Kolovos D, Paige R (2010) Inter-modelling: from theory to practice. In: Petriu D, Rouquette N, Haugen Ø (eds) Model driven engineering languages and systems, vol. 6394. Lect Notes Comput Sci, Springer, Berlin, pp 376–391
16. Guerra E, de Lara J, Kolovos D, Paige R, dos Santos O (2011) Engineering model transformations with transML. Software Syst Model 1–23
17. IEEE (1990) IEEE standard glossary of software engineering terminology. IEEE Std 610.12-1990
18. Jiménez Á (2012) Integrating traceability management in a framework for MDD of model transformations. PhD thesis, Rey Juan Carlos University
19. Jiménez Á, Granada D, Bollati VA, Vara JM (2011) Using ATL to support model-driven development of RubyTL model transformations. In: 3rd International workshop on model transformation with ATL (MtATL2011), Zürich, Switzerland, pp 35–48. CEUR WS
20. Jiménez Á, Vara JM, Bollati VA, Marcos E (2012) Developing a multi-panel editor for EMF trace models. In: 1st Workshop on academics modelling with eclipse (ACME 2012), co-located at the ECMFA 2012, Lyngby, Denmark 2012, pp 63–72
21. Jouault F (2005) Loosely coupled traceability for ATL. In: 1st European conference on model-driven architecture foundations and applications (ECMDA-FA'05), Nuremberg, Germany 2005
22. Jouault F, Allilaire F, Bézivin J, Kurtev I (2008) ATL: A model transformation tool. Sci Comput Programm 72(1–2):31–39
23. Kolovos D, Paige R, Polack F (2008) The epsilon transformation language. In: Vallecillo A, Gray J, Pierantonio A (eds) Theory and practice of model transformations, vol. 5063. Lect Notes Comput Sci, Springer, Berlin, pp 46–60
24. Kurtev I (2008) State of the art of QVT: a model transformation language standard. In: Schürr A, Nagl M, Zündorf A, (eds) Applications of graph transformations with industrial relevance, vol. 5088. Lect Notes Comput Sci. Springer, Berlin, pp 377–393
25. Kusel A (2009) TROPIC-a framework for building reusable transformation components. In: Proceedings of the doctoral symposium at MODELS 2009, School of Computing, Queen's University, Denver 2009
26. Lano K, Kolahdouz-Rahimi S (2011) Model-Driven development of model transformations. In: Theory and practice of model transformations, vol. 6707. Lect Notes Comput Sci. Springer, Berlin, pp 47–61
27. Levendovszky T, Balasubramanian D, Smyth K, Shi F, Karsai G (2010) A transformation instance-based approach to traceability. In: Proceedings of the 6th ECMFA traceability workshop, Paris, France 2010, pp 55–60. ACM, 1814400
28. Mäder P, Gotel O, Philippow I (2009) Enabling automated traceability maintenance through the upkeep of traceability relations. In: Model driven architecture—foundations and applications, vol. 5562. Lect Notes Comput Sci, Springer, Berlin, pp 174–189
29. Oldevik J (2006) MOFScript user guide. Version 0.6 (MOFScript v 1.1. 11) (2006)

30. Oldevik J, Neple T (2006) Traceability in model to text transformations. In: Procedings of the 2th European conference on model driven architecture—traceability workshop (ECMDA-TW'06), Bilbao, Spain 2006, pp 17–26. Citeseer
31. Ramesh B, Stubbs C, Powers T, Edwards M (1997) Requirements traceability: theory and practice. Ann Software Eng 3:397–415. doi:10.1023/a:1018969401055
32. Sánchez Cuadrado J, García Molina J, Menarguez Tortosa M, Ruby TL (2006) A practical, extensible transformation language. In: Rensink A, Warmer J (eds) Model driven architecture—foundations and applications, vol. 4066. Lect Notes Comput Sci, Springer, Berlin, pp 158–172
33. Schmidt DC (2006) Model-driven engineering. IEEE Comput 39(2):25–31
34. Selic B (2003) The pragmatics of model-driven development. Software IEEE 20(5):19–25
35. Sendall S, Kozaczynski W (2003) Model transformation: the heart and soul of model-driven software development. Software IEEE 20(5):42–45
36. Stahl T, Völter M, Czarnecki K (2006) Model-driven software development: technology, engineering management. Wiley, New York, NY
37. van Amstel M, van den Brand M (2011) Using metrics for assessing the quality of ATL model transformations. In: 3rd International workshop on model transformation with ATL (MtATL2011), Zürich, Switzerland 2011, pp 20–34. CEUR-WS
38. Vignaga A (2007) A methodological approach to developing model transformations. Model-driven engineering languages and systems (MoDELS 2007). USA
39. Völter M (2011) From programming to modeling-and back again. Software IEEE 28(6):20–25
40. Walderhaug S, Stav E, Johansen U, Olsen GK (2009) Traceability in model-driven software development. In: Tiako P (ed) Designing software intensive systems: methods and principles. Information Science Reference, Hershey, PA, pp 133–159

An Approach for Identifying Business Requirements for Development of Prospective ERP Systems

Björn Johansson and Sven A. Carlsson

Abstract There are many difficulties in developing business requirements when developing prospective enterprise resource planning systems (ERPs). There exist no approaches for requirements development for prospective ERPs. Using a design science research approach, we develop an approach for requirements development for prospective ERPs. The approach is presented and we discuss how it can be used. The main features of the approach are scenarios and narratives, and we show how these, using different sources, can be developed.

1 Introduction

To identify, gather, and specify requirements in information systems development are non-trivial activities to carry out [1]. The elicitation of requirements is critical in information systems development (ISD) [2–4]. A vast amount of approaches, methodologies, methods, and techniques exist for requirements elicitation in ISD. These approaches, focus on requirements elicitation in situation where one develops an information system for a specific organisation and in many cases also for a limited number of functionalities and users. For enterprise resource planning systems (ERPs), very little exist in terms of approaches. Some have addressed this by developing support for organisations when they are to buy or rent an ERP [3, 5–7]. That is a focus on the user side. When looking at the developer side, the organisations developing ERPs, nothing exists. The aim of this paper is to present and discuss an approach for identifying and describing business requirements for the development

B. Johansson (✉) • S.A. Carlsson
Department of Informatics, Lund University, Ole Römers väg 6,
SE-223 63 Lund, Sweden
e-mail: bjorn.johansson@ics.lu.se

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_19,

of prospective ERPs. If it is hard to identify requirements for information systems in general, it can be argued that identifying requirement for ERPs is even harder. One reason is the fact that ERPs are a type of information systems that are supposed to address more or less all concerned functions of an organisation [8]. After having reviewed relevant literature, we conclude that there is a gap in practice and research on how developer of commercial off-the-shelf (COTS) software, such as ERPs, can identify requirements when developing ERPs for the future. The work presented here aims at describing and developing guidelines for identifying ERP requirements. It does so by employing an ERP requirements approach that suggests sources for finding future requirements.

Hence, the basic problem addressed is how to identify business requirements when developing prospective ERPs. Since the research is solution driven—the development and evaluation of an approach for generating requirements for prospective ERPs—the research approach is design science research (DSR). DSR aims at developing novel artefacts [9] as well as design knowledge [10, 11]. To position our research, we use the work of Hevner et al. [12] and Hevner and Gregor [13]. Drawing on Purao's [14] framework, Hevner and Gregor [13] present three DSR design contribution types: (1) instantiations in the form of products and processes, (2) design principles (e.g. approaches, models, and methods), and (3) emergent design theories about the phenomena under study. The work presented here is a type 2 contribution. The approach presented is general in the sense that any developer of ERPs could use it. Hevner et al. use two dimensions to develop a DSR knowledge contribution framework: (1) maturity of application domain (problems) and (2) maturity of solutions (existing artefacts). Using the two dimensions, they present four types of contributions: (1) invention, invent new solutions for new application problems; (2) inspiration, develop new solutions for known application problems; (3) extension, extend known solutions to new problems, e.g. adopt solutions from other fields and contexts; and (4) routine design, apply known solutions to known problems. They consider the three first as DSR. The work presented here is extension, where we essentially took methods, techniques, etc., having being developed to be used in other contexts. We adapted them to the stated problem.

To address the problem, Sect. 2 presents an approach for ERP requirements management. The approach is a theoretical-driven approach. The aim of the approach is to support ERP developing organisations in identifying, gathering, and specifying ERP requirements for new ERPs. How to use the approach is presented in Sect. 3. The approach is presented and guidelines for the different phases are presented. Section 3 also describes problematic issue in ERP requirement specifications. The final sections then relate the suggested model and its usage to other approaches and problems in ERP requirement descriptions and ends with concluding remarks about the model and the phases it involves.

2 An Approach for Identifying and Presenting Future ERP Business Requirements

As a background to the discussion on how to identify business requirements for prospective ERPs, an ERP business requirement model is presented in Fig. 1. This model builds to some extent on the description that Jarke et al. [15] give concerning scenario management in three disciplines—strategic management, human–computer interaction, and software and systems engineering—and the work they present in suggesting an interdisciplinary framework. The motivation for selecting the framework as a starting point is that it deals with both descriptions of current as well as future realities. This makes it a relevant starting point for an ERP requirements approach, since prospective ERPs need to be built from current realities if they should be able to fulfil future realities. Also, ERPs have connections to all three disciplines. Development of ERPs is definitely related to software and systems engineering, but it also has a close connection to strategic management and human-computer interaction. The model in Fig. 1 builds on the thinking about going from a description of AS-IS situations to a description of TO-BE situations. The model also focuses on scenarios and the role that scenarios could play in identifying and presenting future ERP requirements.

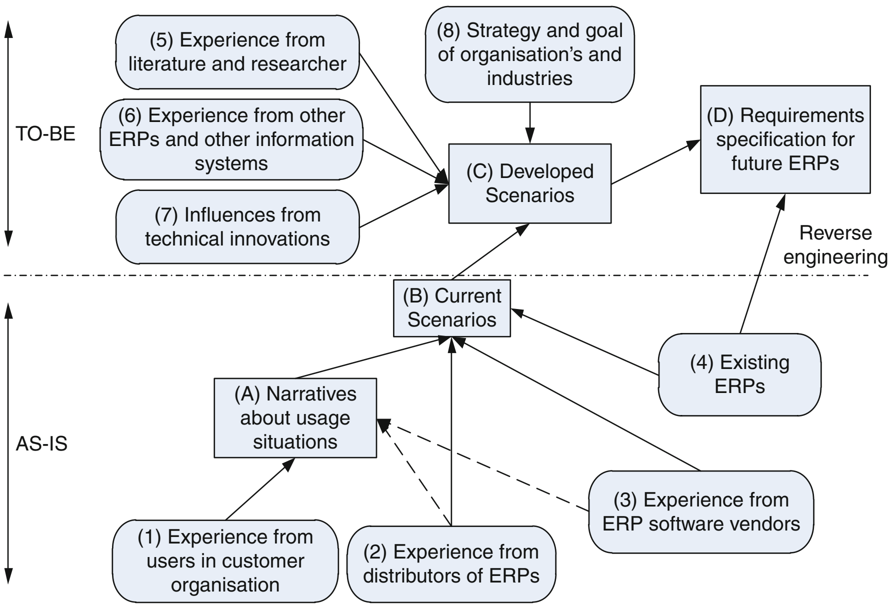


Fig. 1 A model for identifying and presenting business requirements for ERPs (based on Jarke et al., 1998, p. 157 [15])

From the discussion so far, the model shown in Fig. 1 is suggested as an approach for how to capture business requirements of prospective ERPs. The model to a great extent builds on the idea that narratives and scenarios can be used to describe "real" requirements.

A crucial question to answer is then: what are narratives and scenarios in this context? Scenarios and narratives have been suggested and used in the ISD literature. For example, Clausen [16] claims that the use of narratives in system development is "more or less" the solution to the problem of the knowledge gap between end users and systems designers. Two different kinds of narratives are described: "the history", a narrative about the existing situation, and "the scenario", a narrative aiming at describing the future. Hedman and Borell [17] describe narratives as a potential communication medium suitable for presentation of ERP requirements since narratives can manage a high degree of uncertainty and ambiguity. They state: "In fact, narratives might be the only way to capture invaluable input regarding project size and complexity of an ERP implementation" [17]. For discussion and different views on scenarios and narratives in ISD, see, for example, Nielsen [17], Nielsen and Madsen [18], and Schindler [19].

In our approach, the difference between narratives and scenarios is a difference between "what" and "how". Narratives are stories that focus on what the developed system is used for. The scenarios focus on the developed system and the system to be developed and how they can be used in organisations.

The input to the narratives and scenarios could come from at least three different stakeholders: (1) ERP users in customer organisations, (2) ERP distributors (which act as developers by, for instance, customising ERPs and developing add-ons), and (3) ERP software vendors. The narratives will to some extent depend on the functionality already implemented in existing ERPs. They will also be based on suggestions for functionality that is already present in the existing ERP but for one reason or another are not used. From all these information, scenarios will be developed that describe the current situation. The scenarios of the current situation will then be developed into scenarios about the future using input from literature (academic and practitioner), experience, and evidence from other ERP installations as well as ideas regarding technological innovations. By relating the developed scenarios to organisations' goals and requirements for fulfilling these goals, the result should be a better connection between the requirements and the objectives of the organisations. The final step is formalisation from the scenarios in order to achieve a requirements specification for prospective ERPs. The next section will in greater detail describe how the approach could be used.

3 How to Use the ERP Requirement Approach

The model could be described as an overall description of how to identify, select, and present requirements when aiming at developing prospective ERPs. In this section, this is discussed in a step-by-step manner following the numbering A–D in Fig. 1. The numbers 1–8 in Fig. 1 are potential sources for developing scenarios and narratives. However, before discussing how to use the approach, some basic assumptions about requirements in the ERP context are presented, and some concepts used in the figure are defined.

3.1 Business Requirements of Prospective ERPs

Some of the most pervasive and intrusive information systems implemented by organisations are those referred to as enterprise resource planning (ERP) systems [20]. A basic problem in implementation is to align the inherited business processes in ERPs with the ERP requirements the implementing organisation has. This is closely related to the problem of how to identify requirements on prospective ERPs, so the ERPs support business processes in the organisation.

There is a knowledge, information, or data gap between the business analysts and developers of ERPs. A basic problem with developing ERPs is that the one that identifies the needs is often not the same as the one responsible for building the system. This issue is not something unique to developing ERPs; instead, this is more or less always the case in software development, especially in commercial off-the-shelf (COTS) development [21]. However, the problem gets more evident in ERPs since ERPs are supposed to be systems dealing with processes in the entire organisation and therefore have a lot of different users from an entire organisation. From early days, the ambition of ERPs has been to (1) integrate all transaction systems within the same system, (2) share common data and practices across entire enterprises, and (3) produce relevant information for decision-making purposes in real time [22, 23]. An important issue is then to identify and present business requirements. Identifying as well as presenting requirements for information systems is a difficult task to conduct [1].

A first question when it comes to discussing requirements is: what are requirements? Jackson [24] describes requirements as being about the phenomena of the application domain, not about the machine. The application domain is, according to Jackson, in a way similar to environment, but it differs since environment suggests something that physically surrounds the machine (the software product). Using Jackson's view on requirements means that requirements become needs that must have solutions. By having a focus on the application domain, it is possible to focus on requirements and where and how to find them. (For a similar view, see Power [25].) Here, requirements are defined as needs that are not yet fulfilled. This definition makes it necessary to clearly describe the AS-IS situation when identifying requirements for prospective ERPs. One way to do that is by developing narratives about usage situations, as described under the next section.

3.2 Narratives About Usage Situations

In the approach's first phase, narratives (A) are developed from data collected from users (1), experience from distributors of ERPs (2), and experience from ERP vendors (3). The approach could act as input to the development of an interview guide. The interview guide would then specify questions that, for instance, a business analyst has created before interviewing stakeholders involved in giving input to the narratives about usage situations. The concept of sensemaking could be useful in this context [26]. Sensemaking should here be understood as a data collection technique that has context, uses experienced problems, and has problem-solving in focus. However, it could be asked if interviewing is "the best" method to get the correct information for narratives about usage. It is proposed that the interviews should be supported by observations of actual usage situations. We conclude that getting the experiences from users in usage situations could be done as either interviews, observations, or a combination of these two methods. Triangulation is crucial, and it can be achieved by using different data collection methods, like role rep test and laddering [27, 28]. Another possible way to receive this information could be experiments in the meaning that from problem statements the users are supposed to show what and how they use the existing ERP to solve a specific work task. While showing how they use the system, they should think aloud.

When developing narratives, experiences from distributors of ERPs and from ERP software vendors can also play a role. Since the narratives focus on how problems are solved with the assistance of existing ERPs, the narratives should be focused on actual ERP usage. The dotted arrows therefore depict that experiences from distributors and vendors should not be the main sources. These sources are of greater importance when developing current scenarios (B).

3.3 Current Scenarios

The next phase in the approach is to develop scenarios describing the current situation (B). This means that by conducting a reverse engineering exercise from existing ERPs, implemented requirements should be able to document. However, it is not always sure that existing ERPs' implemented functionality reflects the requirements that are needed or desired. What requirements that get implemented are dependent on some historical facts about the system. The architecture that was chosen originally reflects what requirements were implemented. Software vendors have a wish-list with requirements. Due to a lack of resources, there has to be a prioritising of what requirements are implemented.

We assume that distributors of ERPs (2) have a lot of knowledge about what the customers want to have as well as how existing ERPs fulfil the current requirements. The reason is strictly that they have a direct contact with several ERP user organisations. The distributors to a high extent are the ones that help the customers to adjust the basic core requirements to the specific customers' business processes or help customers with advice about how they should reconfigure their business processes. The knowledge from the distributors could act as input to the development of narratives, but most importantly, it probably has an impact on the development of current scenarios (B). This knowledge could be gained from different actors in a distributor organisation, such as sales manager, sales personnel, developers, and personnel working in the help desk. The distributors' experiences differ to some extent from the software vendors' experiences. The distributors are closer to the users, and they are also probably the group that receives direct request regarding desired changes for the existing ERPs.

We assume that ERP software vendors (3) are the ones having least experience about direct usage of ERPs, but despite that the experience they have are highly relevant. The reason for this is that they have general experience that could be said to be relevant when developing systems that are supposed to be used in several organisations. In ERP software vendors' organisations, there exist lists and descriptions of requirements coming from distributors and users, but have yet not been implemented. This means that identified requirements from users and distributors could be validated and the generalisability of these could be checked.

3.4 Developed Scenarios

As described above and shown under the dotted line in Fig. 1, the base for requirements specification are narratives and scenarios, i.e. the reality in which companies operate. The aim of this phase is to develop AS-IS descriptions into scenarios (C) that describe the TO-BE situation. As shown in Fig. 1, there are a lot of different sources of information (5–8) that influence the TO-BE scenario. Organisational strategies and goals are input to developed scenarios. However, it is important to state that even if the model seems to suggest that only executives have the information for the TO-BE situation, this is not the case; of course, they could have and have influential information on the narratives as well as the current scenarios. Most important in this phase is that the scenarios should have a high level of generalisation. This means that the developed scenarios should be described in a way that they do not describe a specific organisation's situation. One way to avoid the "specific organisation's situation" is to develop critical success factors and business processes for industries.

3.5 Requirements Specification for Prospective ERPs

The final phase is to go from developed scenarios to a requirements specification for prospective ERPs (D). This is probably the most complicated step to make and it involves a great extent of formalisation. There are a lot of different methods that could support this step. One of these is the Business Process Modelling Notation (BPMN) [29]. Figures describing processes in BPMN are relatively easy to understand for business people. Another choice could be the Unified Modelling Language (UML).

Independent of what method that is used, the important issue here is to close the gap between different stakeholders in the ERP development chain so that the different stakeholders understand each other. The misunderstandings that exist today could be described as some kind of trade-off between different stakeholders about the level of business requirements. The one that represents the organisational level needs to have the requirements on a more abstract level, which suggests that requirements could be described such as “we want the ERP to support our business processes” or “we want such and such functionality”. These abstract requirements are not at the level that developers with a more technical viewpoint want. The developers’ level of requirements is on the level “what data do you want to be able to see on this screen” or “how many decimals should it be possible to store this data with”. This trade-off means that there has to be some kind of formalisation of the requirements gathered in the narratives and scenarios.

3.6 Summing Up the Approach

The approach consists of two overall parts: a current part and a planned part. The current part of the model (AS-IS) is more or less strictly developed from experiences of and evidences from current usage. The idea is that current scenarios (B) should describe the actual usage situation in organisations. It should describe what and how users use existing ERPs and thereby identify problems in the usage. This also means that the current scenarios are influenced or have to be influenced by existing ERPs. However, this is to some extent problematic when identifying requirements on prospective ERPs. The risk is that if experiences from existing ERPs are used, it could block innovative thinking and might end up in an anchor (in current situation) and adjust. However, we need to look at the history to help us create the future. Another problem when gathering information from existing ERPs is that they often lack documentation. The planned part (TO-BE) builds on the AS-IS data and could or should be seen as an extension of that. There is also some direct influence at the top level in the model about the organisations’ (8) goals and/or objectives that could be gleaned from executives or documentation.

Under the heading “narratives”, we describe how experience from different stakeholders could be collected and described as narratives. The approach suggests that these narratives are transformed into scenarios.

In the approach, the difference between narratives and scenarios should be understood as a difference between “what” and “how”. This means that the narratives are stories that focus on what the developed system should be used for. The scenarios then focus to a higher extent on the developed system and how it can be used in the organisation. It is important to state that the approach focuses on identifying requirements on future development of ERP systems. This means that it has a focus on how to generate and present requirements for vendors of ERP systems when they develop the “core” functionality for prospective ERP systems. The presented approach differs from frameworks, presented by, for instance, Daneva and Wieringa [6], Rolland and Prakash [7], as well as Daneva [5]. These are to be used for assisting user organisations in the selection of a specific ERP system.

4 Discussion

We present an approach and how to use it when identifying, describing, and specifying business requirements for prospective ERPs. During a first evaluation of an earlier version of the ERP requirement approach, it was found that the focus on doing a reverse engineering raised a major problem in ERP requirement management. The problem can shortly be described as ERP vendors developing a core product that are further developed by users of the system, resulting in changes in the core product. This creates two problems. The first is that documentation is sparse, and the second is that the documentation does not reflect the system that is used. This is related to the work done by Power [25]. Power describes the collection of requirements from a grounded theory perspective. He found that requirements documentation has a lot of different names. Power says that this is problematic since it is not clear what the documents are about. There were 17 different names on requirements documents identified by Power. The most common words used were specification and requirements, and labels used on the documents were often some kind of combination of these two words. However, there were also names such as “Business Blueprint and Task Definition”. This indicates that if one searches after documentation on requirements, it is not enough to just search for documents that are labelled “requirements”.

Another concern is then how to categorise requirements. The literature on requirements has a strong focus on classification of requirements [25]. The major part of these classification schemes begins by distinguishing between functional and non-functional requirements [25]. According to Power, this focus is problematic since it does not distinguish between substantive and written-down requirements. By distinguishing between substantive and written-down requirements, a focus on requirements as needs and requirements as text is focused. The major benefit with having this perspective is that it analyses requirements in depth and goes beyond the common breakdown of types of requirements per se.

A major problem in the requirements documentation is that it does not reflect the system that is used. It may in the beginning of the usage of the system, but since the system drifts and the requirements documentation are not updated, it results in documentation not describing the “actual” system. In the ERP case, this is a very common problem, meaning that the vendors cannot clearly state what requirements that are implemented and how they are implemented. This can be described as a product-oriented perspective in contrast to a process-centred perspective. Process centred means that the system development becomes part of an overall ongoing process of organisational change, while product oriented focuses on developing a product without a close connection to any specific organisational change [30]. This can be compared with the two main approaches that exist in information systems design: the traditional system development process and the commercial off-the-shelf product development process [31]. Although both approaches look very similar, there is a substantial difference between these. The former is tailor-made for a specific company, automating processes as they are, while the latter is on-size-fits-all approach where a vendor tries to design an ERP system according to generic business processes based on best practices.

5 Concluding Remarks and Future Research

We have addressed how to identify requirements for prospective ERPs. Using a DSR approach, we developed an approach for requirements development when developing prospective ERPs. In the approach, several sources are used to develop scenarios and narratives. Broadly, DSR consists of two major phases: develop and evaluate. The focus of the paper has been on developing the approach. Hence, future work will focus on evaluation, and based on evaluation, the approach will be enhanced. Evaluation will be based on evaluation guidelines and methods suggested in the DSR literature; see, for example, Hevner and Chatterjee [9] and Carlsson et al. [11]. Another line of research will be to explore how to combine the scenarios and narratives with modelling techniques like BPMN and UML as well as use cases. Yet another line of research will focus on techniques as well as evidence-based information that can be used in the different phases, for example, how strategy research can be used to identify strategies, goals, and key processes in different industries.

References

1. Alvarez R (2002) Confessions of an information worker: a critical analysis of information requirements discourse. Inf Org 12(2):85–107
2. Chakraborty S, Sarker S, Sarker S (2010) An exploration into the process of requirements elicitation: a grounded approach. J Assoc Inf Syst 11(4):212–249
3. Hansen S, Berente N, Lyytinen K (2009) Requirements in the 21th century: current practice and emerging trends. In: Lyytinen K et al (eds) Design requirements workshop. Springer-Verlag, Berlin, pp 44–87
4. Mathiassen L et al (2007) A contingency model for requirements development. J Assoc Inf Syst 8(11):569–597
5. Daneva M (2004) ERP requirements engineering practice: lessons learned. IEEE Software 21(2):26
6. Daneva M, Wieringa RJ (2006) A requirements engineering framework for cross-organizational ERP systems. Reqs Eng 11(3):194–204
7. Rolland C, Prakash N (2000) Bridging the gap between organisational needs and ERP functionality. Reqs Eng 5(3):180–193
8. Worley JH et al (2005) Implementation and optimisation of ERP systems: a better integration of processes, roles, knowledge and user competencies. Comput Ind 56(6):620–638
9. Hevner A, Chatterjee S (2010) Design research in information systems: theory & practice. Springer, New York, NY
10. Carlsson SA (2010) Design science research in information systems: a critical realist approach. In: Hevner A, Chatterjee S (eds) Design research in information systems: theory & practice. Springer, New York, NY, pp 209–233
11. Carlsson SA et al (2011) Socio-technical IS design science research: developing design theory for IS integration management. Inf Syst e-Business Manage 9(1):109–131
12. Hevner A, Gregor S, Jones D (2010) Artifacts and knowledge in design science research. In: ERIC 2010 design science workshop, Maynooth, Ireland, 14 Oct
13. Hevner A, Gregor S (2011) Introduction to the special issue on design science. Inf Syst e-Business Manage 9(1):1–9

14. Purao S (2002) Design research in technology and information systems: truth or dare. 2002, school of information sciences and technology. The Pennsylvania State University, University Park, PA
15. Jarke M, Bui XT, Carroll JM (1998) Scenario management: an interdisciplinary approach. Req Eng 3(3):155–173
16. Clausen H (1994) Designing computer systems from a human perspective: the use of narratives. Scandinavian J Inf Syst 6(2):43–58
17. Hedman J, Borell A (2004) Narratives in ERP systems evaluation. J Enterprise Inf Manage 17(4):283–290
18. Nielsen L, Madsen S (2006) Using storytelling to reflect on IT projects. J Inf Technol Theory Appl 7(4):35–47
19. Schindler E (2007) Getting clueful: five things CIOs should know about software requirements. http://www.cio.com/advice_opinion/development/five_things_it_managers_should_know_about_software_requirements.html?CID=29903&page=1
20. Van Stijn E (2006) Miracle or mirage? An exploration of the pervasive ERP system phenomenon informed by the notion of conflicting memories. University of Twente, Enschede
21. Kesseler E (2008) Assessing COTS software in a certifiable safety-critical domain. Inf Syst J 18(3):299–324
22. Nah F-HF, Zuckweiler KM, Lau L-SJ (2003) ERP implementation: chief information officers' perceptions of critical success factors. Int J Human Comput Interact 16(1):5–22
23. Shanks G, Seddon PB, Willcocks L (2003) Second wave enterprise resource planning systems : implementing for effectiveness. Cambridge University Press, New York, NY, p 449
24. Jackson M (1995) Software requirements & specifications: a lexicon of practice, principles and prejudices. ACM Press, London
25. Power NM (2002) A grounded theory of requirements documentation in the practice of software development, in School 2002, Dublin City University: Dublin. p. 223
26. Gasson S (2005) The dynamics of sensemaking, knowledge, and expertise in collaborative, boundary-spanning design. J Computer-Mediated Commun 10(4)
27. Carlsson SA (1988) A longitudinal study of spreadsheet program use. J Manage Inf Syst 5(1):82–100
28. Schultze U, Avital M (2011) Designing interviews to generate rich data for information systems research. Inf Org 21:1–16
29. White SA (2004) Introduction to BPMN, in business process management initiative. http://yoann.nogues.free.fr/IMG/pdf/07-04_WP_Intro_to_BPMN_-_White-2.pdf
30. Floyd C (1987) Outline of a paradigm change in software engineering. In: Bjerknes G, Ehn P, Kyng M (eds) Computers and democracy: a Scandinavian challenge. Dower Publishing Co, Aldershot, UK, pp 185–202
31. Sumner M (2004) Enterprise resource planning. Prentice Hall, Upper Saddle River, NJ

Conflict in the Value Chain: An Analysis of Competitive Advantage in Requirements Management of ERPs

Björn Johansson and Mike Newman

Abstract There are differing opinions about whether an organization gains a competitive advantage (CA) from an enterprise resource planning (ERP) system. However, the basic question in this paper concerns how receiving a competitive advantage from customizing an ERP influences the feedback loop in the ERP development chain. The paper suggests how the interests ERP development chain stakeholders have in maintaining or improving their competitive advantage in their own markets is complex and potentially conflictual and in some circumstances may hinder the development of future ERPs. From a set of theoretical propositions, eight scenarios are proposed. These scenarios are then illustrated from interviews with stakeholders in the ERP development chain. Just from our initial research, we uncovered evidence for six of these eight scenarios.

1 Introduction

Competitive advantage (CA) and how organizations gain CA from information and communication technologies (ICTs) are subjects that have been discussed extensively. There are different opinions on the answer to the question as to whether ICTs enable organizations to gain CA. Some proponents, such as Carr [1], claim that the technology is irrelevant since it can be treated as a commodity. Others, such as Tapscott [2], argue for its importance, while still other writers say it depends on

B. Johansson (✉)
Department of Informatics, Lund University, Ole Römers väg 6, SE-223 63 Lund, Sweden
e-mail: bjorn.johansson@ics.lu.se

M. Newman
Manchester Accounting and Finance Group, University of Manchester,
M29, Crawford House, Manchester, M13 9PL, UK
e-mail: mike.newman@mbs.ac.uk

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_20,

how the technology is used and that it is business processes that are primary [3]. In reviewing the academic literature, there seems to be a common understanding that it is not the technology as such that eventually provides organizations with CA but how the technology is deployed [4].

However, in this paper, we discuss another perspective of CA in relation to ERPs, and that is how the ERP value-chain stakeholders' interests in maintaining or improving their CA may influence requirements management of ERPs. When we differentiate between the stakeholders in the ERP value chain and their relative positions, the subject becomes more complex. The research builds on a set of propositions suggesting what gives stakeholders in the ERP value chain their CA. The propositions are then discussed using preliminary findings from an empirical study.

The principal question addressed in this paper is: How do stakeholders' beliefs about competitive advantage influence requirements management in ERP development?

The rest of the paper is organized as follows: The next section defines ERPs and describes the ERP value chain and its stakeholders. We then define CA and describe ERPs and CA from the resource-based view of the firm perspective. This is followed by a presentation of the propositions and a table suggesting CA scenarios in relation to the different stakeholders in the ERP value chain. Finally, we present eight scenarios together with some preliminary findings from our own and extant studies and suggest directions for future research.

2 ERPs, the ERP Value Chain, and Its Stakeholders

ERPs are often defined as standardized packaged software designed with the aim of integrating the entire value chain in an organization [5]. Wier et al. [6] argue that ERPs aim at integrating business processes and ICT into a synchronized suite of procedures, applications, and metrics which transcend organizational boundaries. This evolution has increased the complexity not only of usage but also in the development of ERPs. The complexity comes from the fact that ERPs are systems that are supposed to integrate the organization (both interorganizationally and intraorganizationally) and its business processes into one package [7]. It can be assumed that ERPs as well as how organizations use ERPs have evolved significantly from a focus on manufacturing to include service organizations [8]. These changes have created a renewed interest in developing and selling ERPs. Thus, the ERP market is a market that is in flux. This impacts not only the level of stakeholder involvement in an ERP value chain [9] but also how these different stakeholders gain CA from developing, selling, or using ERPs. It is clear that a user organization no longer achieves CA just by implementing an ERP [10]. Fosser et al. [11] present evidence that supports this and at the same time show that for some organizations there is a need to implement an ERP system for at least achieving competitive parity. They also claim that the way the configuration and implementation is accomplished can enhance the possibility to gain CA from an ERP system, but an inability to exploit

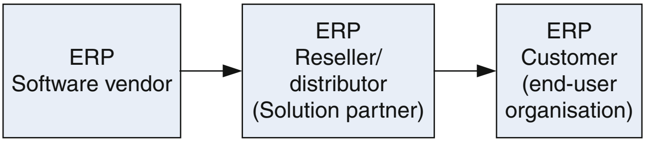


Fig. 1 Stakeholders in the ERP value chain. This is the Microsoft business model. While SAP does not use resellers, they use partners that provide, in some cases, extensive support for ERP customers

the ERP system can bring a competitive disadvantage. This is in line with the assumption from the resource-based view that it is utilization of resources that makes organizations competitive and just implementing ERPs provides little, if any, CA [4]. One reason for this could be that the number of organizations that have implemented ERPs has exploded. Shehab et al. [12] claim that the price of entry for running a business is to implement an ERP, and they even suggest that it can be a competitive disadvantage if you do not have an ERP system. Beard and Sumner [13] argue that through reduction of costs or by increasing organizations revenue, ERPs may not directly provide organizations with CA. Instead, they suggest that advantages could be largely described as value adding through an increase of information, faster processing, more timely and accurate transactions, and better decision-making.

In contrast to the above analysis, we describe the development of ERPs as a value chain consisting of different stakeholders, as shown in Fig. 1.

It can be said that all stakeholders in the value chain, to some extent, develop the ERP further. The software vendors develop the core of the system that they then "sell" to their partners that act as resellers or distributors of the specific ERP. These partners quite often make changes to the system or develop what could be labeled as add-ons to the ERP core. These changes or add-ons are then implemented in order to customize the ERP for a specific customer. In some cases, the customer develops the ERP system further by either configuration or customization. At this stage of the value chain, it can be argued that the "original" ERP system could have changed dramatically from its basic design. This ERP development value chain may result in the ERP software vendors not having as close a connection to the end user that they would choose and they do not always understand what functionalities are added to the end users' specific ERP systems.

The stakeholders in the ERP value chain have different roles; accordingly, they have different views of CA gained from ERPs. One way of describing this is to use a concept from the resource-based view: core competence [14]. Developing ERPs are normally the ERP software vendor's core competence. The ERP reseller/distributors' core competence should also be closely related to ERPs, but it is unclear if development should be their core competency. Their core competences could or should be marketing and implementing ERPs. However, this probably varies between ERP resellers/distributors; for some it could be development of add-ons that constitute one of their core competences. When it comes to end-user organizations, it can be said that ERP development definitely is not their core competence. However, they are involved in the ERP development value chain. To further discuss this, we describe ERPs and CA from the resource-based view of the firm in the next section.

3 ERP and CA Seen from the Resource-Based View

As Mata et al. [4] as well as Kalling [15] claim, whether an organization (the customer in Fig. 1) gains CA from software applications depends on how these resources are managed. The conclusion Mata et al. [4] draw is that among attributes related to software applications—capital requirements, proprietary technology, technical skills, and managerial software application skills—it is only the managerial software application skills that can provide sustainability of CA. Barney [16] concludes that sources of sustained CA are and must be focused on heterogeneity and immobility of resources. This conclusion is made from the assumption that if a resource is evenly distributed across competing organizations and if the resource is highly mobile, the resource cannot produce a sustained CA.

Quinn and Hilmer [17] argue that organizations can increase the CA by concentrating on resources which provide unique value for their customers. Based on the discussion above and the statement made by Quinn and Hilmer [17], Table 1 suggests what the CA is and how it is gained by different stakeholders in the ERP development value chain including the end user.

There are many different definitions of CA; however, a basic definition is that the organization achieves above normal economic performance. If this situation is maintained, the CA is deemed to be sustained. There are some conflicts between attributes for gaining CA, such as developing competitively priced software with

Table 1 ERP value-chain stakeholders and competitive advantage

Stakeholder	Competitive advantage	Gained through
ERP software vendor	High level of market share in the ERP market (e.g., the number software licenses sold)	Competitively priced software Highly flexible software Ease of implementing the software Ease of customizing the software
ERP resellers/ distributor	High level of market share in the ERP consultancy market (e.g., consultancy hours delivered)	Knowledge about the customer's business High level of competence in development of add-ons that are seen as attractive by the ERP end-user organization High level of competence at customization
ERP end-user organization	High level of market share in the customer-specific vmarket (e.g., products or services sold, rising market share, lower costs)	Being competitive in its own market Implementing an ERP system that supports its business processes Implementing an ERP system that is difficult for competitors to reproduce

high flexibility and developing software that is easy to customize and, at the same time, achieve CA by developing exclusive add-ons.

If the organization is a first mover in the sense that it is the first organization that uses this type of resource in a specific way, it can quite easily gain CA, but it will probably only be temporary. The length of time that the CA lasts depends on how hard or expensive it is for others to imitate the usage of that resource. This means that the question of how resources are exploited by the organization is the main factor when it comes to whether the CA becomes sustainable or not.

From an ERP perspective, we suggest that vendors as well as distributors (Fig. 1) provide value by delivering complementary core competencies to their customers. The evolution of ERPs has made these resources easier to imitate. However, a major barrier to imitation is the cost of implementation.

The resource-based view claims that a resource has to be rare or, as described by Mata et al. [4], be heterogeneously distributed to provide CA. In the case of ERPs, this kind of resource is not rare. There are a lot of possibilities for organizations to choose to implement different versions of ERPs, and the evolution of ICT has made it feasible for more organizations to implement ERPs by decreasing the costs of using them. However, as described by Barney [18] and Shehab et al. [12], failure to implement an ERP can also lead to an organization suffering a competitive disadvantage.

The CA from ERPs would probably be negated by duplication as well as by substitution. If, for instance, the ERP resellers sold their add-ons to the ERP software vendor, the duplication of that add-on would be quicker and the CA that the ERP reseller previously had would be gradually eroded. However, if they kept the add-on as "their" unique solution, other ERP resellers or ERP software vendors would probably find a substitute to the add-on or develop their own.

This implies a conflict between vendors and resellers when it comes to CA and the development of "better" ERPs. This can be explained by realizing that ERP resellers/distributors often develop add-ons which have a specific functionality for solving a particular problem for their customer. This can be seen as one way of customization, where resellers/distributors use their domain knowledge about the customers' industry in addition to their knowledge about the specific customer. This, in effect, allows resellers to increase their CA and earn abnormal returns. Another way is for resellers to sell the add-on to other resellers resulting in the resellers decreasing their CA in the long run. It is probable that resellers who sell their add-on solutions to other resellers would see it as not influencing their CA since they sell the add-on to customers already using the same ERP system and this would not make ERP end-user organizations change resellers. However, the question remains whether the same would apply if the resellers sold the add-on to the software vendor. The answer would depend on the incentives that the resellers had for doing that. If the add-ons were to be implemented in the basic software, the possibility of selling the add-on to client organizations, as well as to other resellers, would disappear.

Beard and Sumner [13] conclude that it seems that ERPs are increasingly a requirement for staying competitive (i.e., competitive parity) and that ERPs can yield at most a temporary CA. From this it can be suggested that ERP end-user organizations want a "cheap" system that they can use to improve their business processes, thereby making a difference compared with other organizations in the same industry. But, since ERPs encourage organizations to implement standardized business processes (so-called best practice; Wagner and Newell [19]), organizations get locked in by the usage of the system, and then, depending on whether they are a first mover or not, they receive only a temporary CA. This implies that the ERP end-user organizations often implement an ERP with the objective of having a "unique" ERP system. If the customer believes they have a unique business model, it is likely they would want a unique ERP system. However, they also want a system with high interoperability internally, as well as one compatible with external organizations systems. It is likely that end-user organizations have a need for a system that is not the same as their competitors. This is congruent with the ERP resellers/distributors. They receive their CA by offering their customers the knowledge of how to customize an ERP using industries' best practices and, at the same time, how to implement functionality that makes ERP system uniquely different from their competitor's system. Based on this discussion, we now present some propositions.

4 Propositions

Proposition 1: Both resellers and end users (encouraged by resellers) in the ERP value chain see customization as a way of achieving CA. This results in resistance to providing software vendors with the information necessary for them to develop ERPs further in the direction of standardization and thereby decreasing the resellers' need to customize the system.

Kalling [15] suggested that the literature on resource protection focuses, to a large extent, on imitation, trade, and substitution. Another way of protecting resources is, as described by Kalling, to "protect by development." This means that an organization protects existing resources by developing resources in a way that flexibility is increased by adjusting and managing present resources. In the ERP case, this could be described as customizing existing ERPs, thereby sustaining CA gained from using the ERP system. Kalling describes this as a way of increasing a time advantage. From the different ERP stakeholders' perspectives, it could be argued that both protection by development and trying to increase the time advantage influence the direction in which ERPs are developed.

Proposition 2: The conflict between different parties in the ERP value chain and how they believe they will gain CA decreases the feedback in the ERP value chain. This tends to increase the cost for both development and maintenance of ERP systems.

The discussion and propositions so far suggest that decision-makers in organizations and their beliefs regarding how to gain and sustain CA by customization of ERPs are a major hindrance to the development of future ERPs. This emanates from the assumption that organizations (end users and resellers) protect what customization they have made. The reason why they do so is based on their belief that they will sustain a CA gained by developing, selling, or using customized ERPs. However, returning to Table 1 and the suggestion as to what it is that constitutes CA for the different stakeholders, it can be concluded that there are some generic influencing factors. The conflicting goals of the three parties in the ERP value chain increase complexity in the marketplace. From a resource-based perspective, first mover advantage could be seen as something that influences all stakeholders and their possibility to gain and to some extent sustain CA. The same could also be said about speed of implementation. The main suggestion is that even if the role of history, causal ambiguity, and social complexity influences the organizations' possibility to gain CA, the management skills that the organizations have are crucial.

When looking at what improves their market share of the three different stakeholders in the ERP value chain, it can be proposed that there are no direct conflicts among stakeholders. The reason is that they all have different markets and different customers; therefore, they do not compete directly with one another. In reality, they have each other as customers and/or providers, as described in Fig. 1. It is suggested that further development of ERPs carried out by vendors could result in a higher degree of selling directly to end customers or other ways of delivering ERPs to end customers so that the partners will be driven to insolvency and replaced by, for instance, application service provision (ASP) or software as a service (SaaS). The first step in this direction would probably be signaled if the add-ons that partners currently deliver to end customers are implemented in the core product. It can be concluded that there is a potential conflict between the different parties in the value chain when it comes to how different stakeholders gain CA and how that influences future ERP development.

ERP software vendors become competitive if they utilize their resources to develop ERPs that are attractive to the market. ERP resellers/distributors thus need to utilize their resources to become attractive partners when implementing ERPs. Furthermore, ERP end users need to use the ERP system so that it supports their businesses. In other words, it is how end-user organizations employ the ERP that is of importance, and it could be that having a unique ERP system (Table 1) is not as important as has previously been believed. In other words, while customization is in the interests of the resellers, this may not be the case for the end users.

Millman [20] posits that ERPs are the most expensive but least value-derived implementation of ICT support. The reason for this, according to Millman, is that a lot of ERP functionalities either are not used or are implemented in the wrong way. That it is wrongly implemented results from ERPs being customized to fit the business processes instead of changing the process so that it fits the ERP [20]. However, according to Light [21], there are more reasons for customization than just the need for achieving a functionality fit between the ERP and the organization's business processes. He believes that from the vendor's perspective, customizations might be seen as fuelling the development process. From an end user's perspective, Light describes customization as a value-added process that increases the system's acceptability and efficiency [21]. He further reasons that customization might occur as a form of resistance or protection against implementation of a business process that could be described as "best practices." One reason why end-user organizations get involved in ERP development is that they want to adjust their ERPs so that they support their core competences.

Proposition 3: End users of ERPs and their basic assumption about how they receive CA are encouraged by resellers of ERPs. Resellers want to sustain their CA by suggesting and delivering high levels of ERP customization.

The main conclusion so far can be formulated as follows: Highly customized ERPs deliver better opportunities for CA for the resellers in the ERP value chain, while they decreases the opportunity for both ERP software vendors and ERP end-user organizations to attain CA.

To discuss this further, in the next section, we propose various scenarios supported by some early empirical data.

5 Scenarios Describing ERP-Related Competitive Advantage

In this section, we sketch out how an ERP system could be described from a CA perspective as eight possible scenarios, shown in Table 2.

The description is based on open-ended interviews done with an ERP vendor, ERP reseller consultants, and ERP customers, and in recently published studies, one of the authors was involved in two Norwegian companies [22]. The interviews with the ERP vendor and the ERP reseller consultants were part of an ongoing research project investigating requirements management. The project aimed at gaining knowledge on what factors influence future development of ERPs. In total there were 11 interviews conducted with different executives at a major ERP vendor organization and three interviews conducted with ERP consultants at a reseller organization. The reseller organization implements and supports different ERP systems, and one of their "products" is the ERP system that is developed by the ERP vendor. The interviews with ERP customers (in total 19 interviews) were then part of a research project that aimed at understanding competitive advantage in an ERP context. We use some citations from these interviews to illustrate our findings and flesh out the content of Table 2.

Table 2 Scenarios describing win or lose relationship

Scenario	Vendor	Reseller	Client (end user)
A	Win	Win	Win
B	Win	Win	Lose
C	Win	Lose	Win
D	Win	Lose	Lose
E	Lose	Win	Win
F	Lose	Win	Lose
G	Lose	Lose	Win
H	Lose	Lose	Lose

Scenario A: It can be said that this is probably the situation that all stakeholders in a business relationship ideally want. However, to have a win-win situation in an ERP development value chain is not straightforward. From the vendors' perspective, it means that they should develop an ERP system that is so generic that the reseller could sell it to a lot of different clients to generate licenses and, at the same time, is so specific that the end users could gain a CA. However, if the vendor manages to develop such a generic form of ERP, it is likely that end user would demand an extensive customization effort. The result could then be that the reseller could sell a lot of consultancy hours for adjusting the software to the business processes in the client's organization. A quotation from an ERP consultant at an ERP reseller organization describes a situation when the feedback loop worked as a win-win situation:

Before the ERP vendor merged with a bigger ERP vendor we had a close relationship that actually made it possible to have requests from a specific customer implemented in the system. Now we don't know who to talk with and even if we get a contact with them (the vendor) they are not really interested. (ERP consultant)

He continues with stating that

> We developed a very interesting add-on for a customer, that we then tried to get implemented in the base system but it was impossible. So, we started to sell this add-on to other ERP resellers (of the same system). We did so because we think it will benefit us in the long run if customers feel that the system is interesting—In that way we will probably increase our market. (ERP consultant)

If this continues for some time, it probably ends with a situation as in scenario E. Scenario E is then the situation when vendor loses and the reseller and clients win. We see this as a possibility if the resellers spend so much time with clients developing ERP systems offering CA while generating large consultancy hours but at the cost of not marketing the base ERP system to new clients. Our early data gathering suggests this scenario is common among the stakeholders. One example of support of this situation is the following statement from an executive at the ERP vendor (the same ERP vendor that was mentioned above by the developer at the ERP reseller). The executive at the ERP vendor said that

> We don't have enough knowledge about how the system is used and what the user of the system actually wants to have. This makes that future development of the system is extremely hard and it is a fact that there are problems with requirements management in ERP development. (Director of Program Management)

One way of dealing with this is to get a closer relationship to some ERP resellers—by a relationship program giving some benefits to reseller that have a close relationship with the vendor. However, it demands that they, for instance, follow a specific process for implementation of the ERP.

This could then result in the situation described in *scenario B*, in which both the vendor and the reseller have a win-win situation while the client has a disadvantaged position especially if they do not customize the software to the extent whereby they gain CA. The following quotations from ERP customers describe this situation:

> An ERP system is something you just need to do business today. But the way we have implemented it and configured it has given us a competitive advantage. (Assistant Director of Logistics)

> I believe that it is mostly a system you need to have. But an ERP system can be utilized to achieve a competitive advantage, if you are skillful. (Senior Consultant)
>
> It keeps us on the same level as our competitors. We are focusing on quality products. That is our competitive advantage. An ERP system cannot help us with that. (The Quality Manager)
>
> I don't think we have got any competitive advantage. All our competitors are running such a system, so it is just something we need to have. It is actually a competitive disadvantage because we have not managed to get as far as the others, with the system. (Managing Director)

Another reason why the situation could result in scenario B is that it is shown that if clients customize to a high extent, the long-term maintenance costs of the ERP system become so great that the benefits are lost. The following statement from a developer at the ERP vendor supports scenario B:

> It is clearly seen that when a customer implement the ERP system for the first time they customize a lot. When they then upgrade with a new version the extensive customization is much less and when they upgrade with version 3 and/or 4 they hardly don't do any customization. The reason is most likely that they have discovered that customization cost a lot at the same time as they have discovered that they are not that unique that they thought when implementing the first version (Program Manager A).

In the long run, this could also result in *scenario F*. *Scenario F* describes the situation where the vendor starts to lose market share because clients have problems achieving CA resulting in a bad reputation for the ERP product. The situation of less customization and less demand on add-ons could also result in scenario C. In *scenario C*, we see a vendor bypassing the reseller and working directly with the client enabling them both to gain a CA. This is somewhat supported by an executive at the ERP vendor, who says:

> However, there will probably be a day when the partners are not needed—at least for doing adjustments of ERPs. This is not a problem since the rules of the game always change. And there will still be a need for partners. The partners see themselves as … they understand the customer's problem. (Program Manager B)

Scenario D is an interesting scenario since it is only the vendor that shows a winning position. It could be explained by the fact that if the vendor manages to develop a generic ERP system and thereby gain a more or less monopoly status, they will have the possibility to sell many licenses. A quotation from an ERP customer describes this clearly:

> I try to exploit the available tools in SAP without investing money in new functionality. There are a lot of possibilities in the ERP systems, e.g. HR, which we are working with to utilize our resources more efficiently. (Director of Finance)

It could also be that the client needs to buy and implement the ERP since it more or less a necessity to implement an ERP to obtain competitive parity.

With *scenario G*, it is probably a situation that the vendor would not allow to continue. However, for an ERP customer, it could be described in this way:

> We have a unique configuration of the system that fits our organization and this gives us a competitive advantage. The IS department is very important in this context. (Assistant Director of Logistics)

> I actually hold lectures about how we do things in our organization. I tell others about the big things, but I think it is the small things that make us good. All the small things are not possible to copy. I think it is a strength that we have a rumor for being good at ERP and data warehouse. It gives [us] a good image. Though, we are exposed to head hunters from other organizations. (Director of IS)

It is difficult to believe that either *scenario G* or *scenario H* is sustainable in the long run, and we did not find any evidence to support these scenarios.

6 Concluding Remarks and Future Research

Using an innovative value-chain analysis considering the ERP vendor, reseller, and client, we developed eight scenarios to examine our research question: "How do stakeholders' beliefs about competitive advantage influence requirements management in ERP development?" From our preliminary empirical research, we found evidence to support six of the eight scenarios. As the other two were the least likely to occur, we are encouraged by our findings to conduct further systematic research in the future to flesh out our findings and to look particularly at ERP acquisitions in a variety of settings. As ERP systems are ubiquitous in modern corporations, it is vital that managers consider the value such systems offer in the long term. Furthermore, our analysis offers a more in-depth understanding of the dynamics of the ERP development value chain, its complexity, and its impact on competitive advantage for the different stakeholders.

For the future, we will also try to reveal the patterns that emerge in the value chain and investigate which scenarios are more sustainable in the long term and how clients can position themselves more effectively to improve their competitive advantage.

References

1. Carr NG (2003) IT doesn't matter. Harvard Business Rev 81:41–49
2. Tapscott D (2004) The engine that drives success. CIO Magazine, 2004 (May 1)
3. Smith H, Fingar P (2003) IT doesn't matter—business processes do: a critical analysis of Nicholas Carr's I.T. article in the Harvard Business Review. Tampa, FLA, Meghan-Kiffer 126s
4. Mata FJ, Fuerst WL, Barney JB (1995) Information technology and sustained competitive advantage: a resource-based analysis. MIS Quarterly 19(4):487–505
5. Lengnick-Hall CA, Lengnick-Hall ML, Abdinnour-Helm S (2004) The role of social and intellectual capital in achieving competitive advantage through enterprise resource planning (ERP) systems. J Eng Technol Manage 21(4):307–330
6. Wier B, Hunton J, HassabElnaby HR (2007) Enterprise resource planning systems and non-financial performance incentives: the joint impact on corporate performance. Int J Acc Inf Syst 8(3):165–190
7. Koch C (2001) ERP-systemer: erfaringer, ressourcer, forandringer. København, Ingeniøren-bøger 224

8. Botta-Genoulaz V, Millet PA (2006) An investigation into the use of ERP systems in the service sector. Int J Product Econ 99:202–221
9. Somers TM, Nelson KG (2004) A taxonomy of players and activities across the ERP project life cycle. Inf Manag 41(3):257–278
10. Karimi J, Somers TM, Bhattacherjee A (2007) The impact of ERP implementation on business process outcomes: a factor-based study. J Manage Inf Syst 24(1):101–134
11. Fosser E et al (2008) ERP Systems and competitive advantage: some initial results. In: 3gERP 2nd workshop, Copenhagen Business School
12. Shehab EM et al (2004) Enterprise resource planning: an integrative review. Business Process Manage J 10(4):359–386
13. Beard JW, Sumner M (2004) Seeking strategic advantage in the post-net era: viewing ERP systems from the resource-based perspective. J Strategic Inf Syst 13(2):129–150
14. Javidan M (1998) Core competence: what does it mean in practice? Long Range Plann 31(1):60–71
15. Kalling T (1999) Gaining competitive advantage through information technology: a resource-based approach to the creation and employment of strategic IT resources. Lund studies in economics and management, 55. Lund Business Press, Lund [10], 336 [4]
16. Barney JB (1991) Firm resources and sustained competitive advantage. J Manage 17(1): 99–120
17. Quinn JB, Hilmer FG (1994) Strategic outsourcing. Sloan Manage Rev 35(4):43–55
18. Barney JB (2002) Gaining and sustaining competitive advantage, 2nd edn. Prentice Hall, Upper Saddle River, NJ, p 600
19. Wagner EL, Newell S (2004) 'Best' For Whom?: The Tension Between 'Best Practice' ERP Packages and diverse epistemic cultures in a university context. J Strategic Informat Syst 13:305–328
20. Millman GJ (2004) What did you get from ERP, and what can you get? Financ Executives Int 5(May):15–24
21. Light B (2005) Going beyond "misfit" as a reason for ERP package customisation. Comput Ind 56(6):606–619
22. Fosser E et al (2008) Organisations and vanilla software: what do we know about ERP systems and competitive advantage? In: 16th European conference on information systems, Galway, Ireland

Evolving IT Management Frameworks Towards a Sustainable Future

Marcel Korte, Kevin Lee, and Chun Che Fung

Abstract Information Technology (IT) Management Frameworks are a fundamental tool used by IT professionals to efficiently manage IT resources and are globally applied to IT service delivery and management. Sustainability is a recent notion that describes the need for economic, environmental and social development without compromising the ability of future generations to meet their own needs; this applies to businesses as well as society in general. Unfortunately, IT Management Frameworks do not take sustainability into account. To the practitioner this chapter demonstrates sustainability integration which allows CIOs and IT managers to improve the sustainability of their organisation. To the researcher this chapter argues that sustainability concerns need to be provided through its integration into the mainstream of IT Management Frameworks. This is demonstrated through the high-level integration of sustainability in Six Sigma, CobiT, ITIL and PRINCE2.

1 Introduction

Information Technology Management Frameworks are being widely used in the IT industry and most other market sectors worldwide. While some of these frameworks target a specialised audience or focus on specific business processes, others take a more high-level approach and are being applied across many industries.

Most of these frameworks have been developed during the 1970s and 1980s, in times affected by a strong focus on shareholder value and profit gain. Despite their

---

M. Korte (✉) • K. Lee • C.C. Fung
School of Information Technology, Murdoch University, Murdoch, WA 6150, Australia
e-mail: consulting@marcelkorte.com; kevin.lee@murdoch.edu.au; l.fung@murdoch.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_21,

popularity and the fast pace of change in the IT market current IT Management Frameworks lack a focus on the challenges and concepts of sustainability.

Markets, in particularly in wealthy western countries, have undergone a change in business focus recently. Driven by an increasing public demand and pressure for environmental protection and increasing awareness of social discrepancy, the concept of stakeholder value has become more and more prevalent. The long established profit-driven focus of shareholder value is experiencing an expansion to include ecological and social issues. This is often called the triple bottom line (TBL) or simply phrased "people, planet, profit" [6].

The term "sustainability"—in public use—has become a positive buzzword with no tangible definition. However, throughout the scientific community sustainability is concisely defined by the Brundtland Commission of the United Nations as economic, environmental and social development meeting the needs of today without compromising the ability of future generations to meet their own needs [26, 27]. This chapter uses the term "sustainability" in the context of this definition.

IT is a significant and growing part of the environmental problem. Its carbon footprint is estimated to account for 2–3 % of global carbon dioxide emissions—approximately as much as the aviation industry [2, 12]. However, sustainably managed, Information Systems—the utilisation of IT and organisational activities to further enable operations, management and decision making—offer multiple options to reduce the environmental burden other business processes create and contribute to the reduction of environmental concerns. The use of IT is projected to eliminate 7.8 metric gigatons of greenhouse gasses annually by 2020 [15]. Unfortunately, the issues of maintaining low cost IT operation, building green reputation capital and supporting corporate green strategies haven't been actively researched [16]. While the social impact of growing technological pervasion is widely discussed [7], the opportunities such as in health care and rural development have just begun to unfold. IT is believed to have potential to transform industries [16].

With the IT industry's ongoing growth and increasing impact, it is increasingly challenged to take the concept of sustainable management into account [23]. Information Systems will be a major element in the transition to a sustainable economy [8]; however, it is widely lacking a scientific foundation on how to address this complex demand. This chapter argues that management of all resources in the IT domain needs to integrate sustainability into its supporting frameworks to become the driver towards a sustainable business. It propose integrating ideas from sustainability research into existing IT Management Frameworks.

The remainder of this chapter is structured as follows. Section 2 presents a background on the currently leading IT Management Frameworks and outlines the challenges for researchers and practitioners alike. Section 3 discusses the methodology used and core concepts of sustainability from an IT Management perspective and attempts a high-level application to Six Sigma, CobiT, ITIL and PRINCE2. Section 4 discusses the resulting implications. Finally, Sect. 5 presents some conclusions.

2 IT Management Frameworks

Management Frameworks are structured, descriptive tools that standardise the management process or sub-processes. They can be high-level, utilising existing best practices, umbrella frameworks that utilise other management tools or descriptive approaches to domain specific processes. IT Management Frameworks provide organisations with tools and guidelines to deliver technical, organisational and legal support for its IT provision. A wide range of frameworks are used extensively.

2.1 Categorisation

A Categorisation of IT Management Frameworks (based on [20]) is as follows. The vast majority of these frameworks were developed in the 1970s and 1980s [20]. This study chose to focus on Six Sigma, CobiT, ITIL and PRINCE2 as these are the most prominent frameworks in each category:

1. Quality Management and Improvement Frameworks
2. IT Governance Frameworks
3. Information Management Frameworks
4. Project Management Frameworks

Six Sigma (6σ) [4] has its origins in the introduction of the normal curve by Gauss and Shewhart's work in process quality control. As a further development from Total Quality Management the term "Six Sigma" was phrased in the early 1980s by Motorola. While its origins lie in manufacturing, Six Sigma is now used in multiple industries including IT. It uses a wide range of quality improvement tools (see [19]) and an umbrella methodology. Six Sigma aims to be used beyond process improvement to include business operation, aiming to reduce the variation in the business and take customer-focused, data-driven decisions. Six Sigma focuses on eliminating variability, defects and waste to realise its goal of increasing profits.

COBIT (Control OBjectives for Information and related Technologies) [10, 11] focuses on harmonising the elements of IT Governance and is the de facto standard process model for good practices in IT. It focuses on what should be achieved rather than how to achieve it through effective governance, management and control. It defines IT Governance as consisting of strategic alignment, value delivery, risk management, resource management and performance management.

ITIL (IT Infrastructure Library) [1], developed in 1980 by Central Computer and Telecommunications Agency in the UK, has undergone several revisions being currently at version 3 published in 2006. It focuses on improving the overall QoS to business within cost constraints, while improving the overall economic effectiveness and efficiency of IT. ITIL is a collection of best practices that describe the realisation of IT service management and has become the global de facto standard in this area. ITIL defines five general main activities.

PRINCE2 [13] (PRojects In Controlled Environments) is a project management method developed in 1979; it has undergone its last revision in 2009. PRINCE2 is the de facto best practice project management standard in the UK and is globally widely in use. At its highest level PRINCE2 describes eight main processes explaining what should happen, when it should be done and by which role. One of its key principles is the need for a valid business case agreed on by all project stakeholders. Projects are reviewed constantly, and loses justification when the business case is not given anymore and is discontinued.

2.2 Lack of Sustainability Focus

As the importance of IT is rising globally, top management is increasingly being required to justify the needs for IT services and show evidence that investments pay off. They are expected to account for costs as well as identifying and delivering value to the business. To secure project funding and sustain departmental budgets, CIOs and IT managers are required to align their activities to the organisation's business objectives, to provide effective and efficient planning, ensure competent risk management and demonstrate profitable life cycle management. These requirements are best addressed by applying proven best practices and standardised models [20].

These frameworks mostly focus on increasing profit and improving financial efficiency. The numerous success stories measure their success in increased return on investment (ROI), e.g. million dollars reduced cost. While these frameworks on lower levels mention people, they still consider the human resource from a mere economic perspective. Social and environmental aspects are not explicitly mentioned in any of these frameworks. The lack of TBL integration hinders the operations' transformation towards sustainable business practice. This results into several research questions, affecting the practitioner and the researcher alike:

1. Do the four exemplarily chosen frameworks hinder any transformation towards sustainable business at all or is it possible to conserve their proven advantages and extend their capabilities?
2. If (1) can be answered positively, how will these frameworks need to be transformed to operate in a sustainable business?

3 Sustainability and Existing IT Management Frameworks

There is a wide range and variety of IT Management Frameworks ranging from horizontally focused frameworks, vertically focused frameworks, frameworks specialised in a single area and frameworks with a wider focus. Each framework only

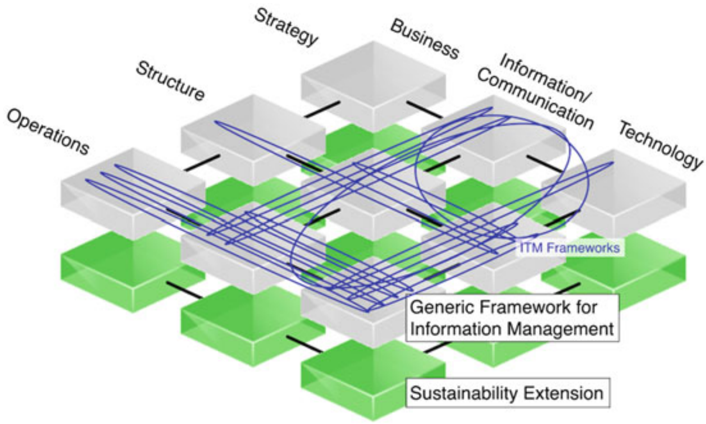


Fig. 1 The coverage of Common IT Management Frameworks in relation to the Generic Framework for Information Management extended by a dimension of sustainability (adapted from [14])

covers a sub-domain or aspect of the whole field of Information Technology Management. This means that any approach to integrate sustainability into IT Management Frameworks must target multiple frameworks to be successful.

3.1 Methodology

IT managers aiming to drive corporate sustainability have to manage their economic, natural and social capital. They need to: (a) contribute to the organisation producing a consistent return to their shareholders while ensuring sufficient cash flow for liquidity; (b) ensure the use of natural resources at a lower rate than their reproduction or the development of substitutes, assure an emission rate below the capacity of the environment to absorb and assimilate these and refrain from activities degrading eco-system services; (c) increase the organisations human capital add value to the community it is operating in and manage social capital in a way that all stakeholders can understand and agree with the organisations activities [5]. To integrate the goals of sustainability in IT means integrating them into the pervasive IT Management Frameworks. An analysis of the leading frameworks shows that ecological and social aspects are not being taken into account.

The Generic Framework for Information Management is a high level model that encapsulates all IT Management activities. Figure 1 illustrates the complexity of the most utilised IT Management Frameworks. The elliptic circles represent the operational areas of 11 of the most used IT management frameworks (from [20]) and their functional orientation related to the Generic Framework for Information Management [14]. The figure illustrates that integrating sustainability effects the entire business

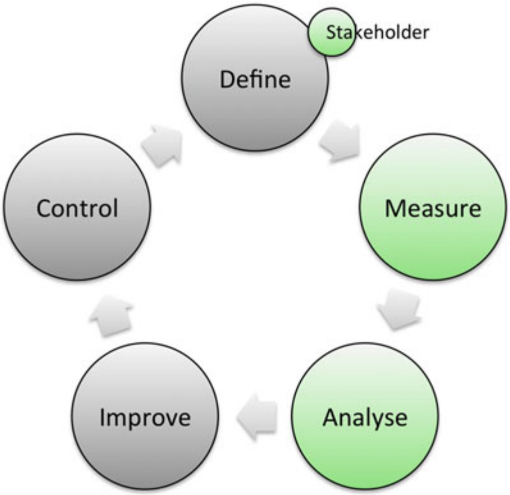


Fig. 2 Sustainability's influence on the Six Sigma project DMAIC life cycle

and not just the IT department while also showing the variety and complexity of existing frameworks.

As individual frameworks target different areas of IT Management, sustainability extensions need to target different aspects of IT Management. The leading frameworks cover one aspect of sustainability—the business case—therefore there is a need to extend IT Management Frameworks by the ecological and social dimensions.

The existence of a business case for sustainability has been widely discussed in the literature [18, 24, 29]. According to [21] a business case for sustainability needs to fulfill three requirements as follows: (a) The activity has to be voluntary or mainly voluntary and aim to solve or improve a societal or environmental problem. (b) The activity has to create a positive ROI or ramification on corporate success which is measurable or at least a convincing reasoning. (c) A convincing conclusion must be drawn that a certain activity has or will lead to the outcome. The following sections illustrate the integration of sustainability into the categories introduced in Sect. 2.1.

3.2 Quality Management and Improvement: Six Sigma

Figure 2 illustrates the Six Sigma DMAIC project life cycle [9] and highlights those processes that lack a focus on sustainability. To successfully integrate sustainability in Six Sigma a focus on these areas is required.

The *Define* process defines the problem (such as excessive consumption of hazardous substances), the customer (internal and external) and the project goals specifically.

So far the current definition of the define process lacks a thorough analysis of all project stakeholders which needs to be seen as a precondition for TBL success, e.g. it could be more effective to outsource a production line to a third world country yet the overall sustainability accounting might be negative due to considerably lower environmental standards and slack work regulations.

Data collection and project assessment is one of the main activities of the *Measure* process. Based on this collection, project key performance indicators (KPIs) are defined and their values measured to determine types of defects and metrics. These activities need to be expanded beyond the focus of economical data towards a view including ecological and social aspects. This raises the question of how to quantify social and environmental aspects in order to allow them to be measured and monitored. As indicated above such measurements cannot always be found and in case need to be replaced by solid and well-founded reasoning. In such cases the statistics driven approach of Six Sigma cannot be applied.

Analysis is focused on the cause and effect of aspects of the project life cycle. Analysing cause and effect has always been a core activity in industrial production processes, analysing cohesion in ecological and social terms is increasingly complex and challenging. Often verifying and quantification of cause and effect correlation for sustainability is little or malpracticed [3, 22], other times it needs to be substituted by solid reasoning. As Six Sigma relies on solid measurable data, it lacks the ability to significance test improvement hypothesis in some cases of ecological or social improvement so cannot be applied to the entirety of sustainable activities. *Improve* and *Control* are not further affected by an expansion towards sustainability.

3.3 IT Governance: COBIT

As an IT Governance Framework CobiT is less directly focused on immediate economical profit but rather set the way to allow for financial success. The framework aims to ensure that IT enables the business and maximises benefits. CobiT focus primarily on what is required rather than how to undertake the activities themselves. Therefore it is less reliant on implementing best practice and more focused on understanding the business requirements for IT. For users of CobiT to meet the aims of more sustainable ways of doing business, sustainability must be integrated. However, it is not CobiT itself needing to undergo a change, it is rather that practitioners when applying CobiT need to assure that they actively account sustainability within the four CobiT process steps. CobiT's core processes are well designed to be able to do this without any additional change.

Figure 3 illustrates the CobiT framework and indicates those steps that need reviewing for sustainable operation. An example of this involves strategic planning—part of the *Plan and Organise* component—that needs sustainability added as technological direction. IT risk management activities and HRM need to expand

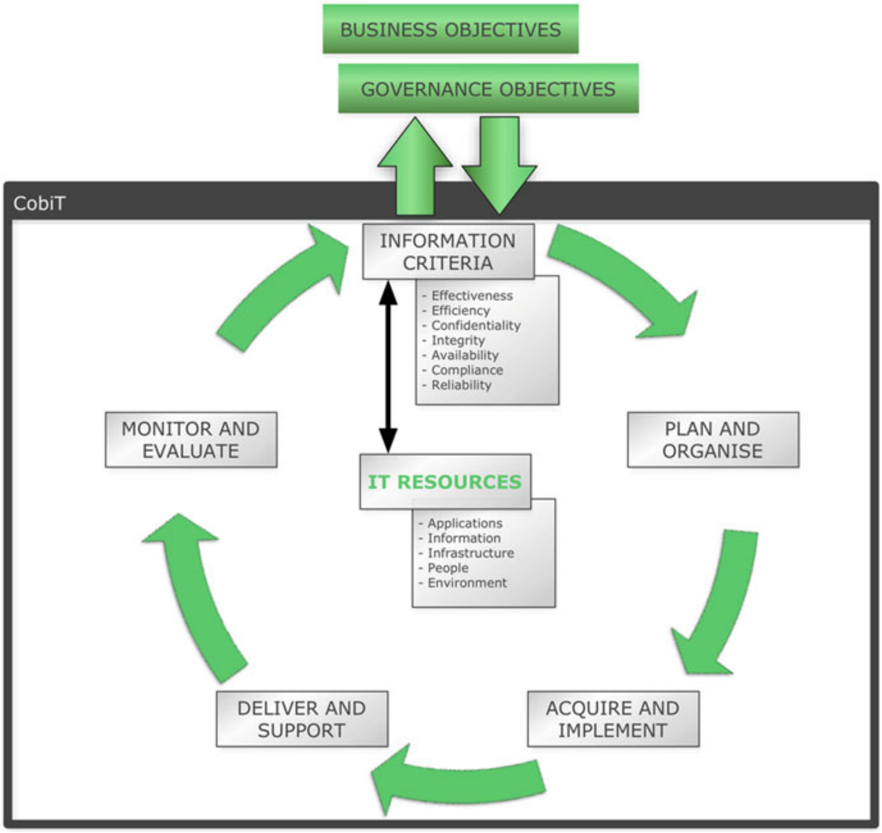


Fig. 3 Sustainability extended CobiT (adapted from [11])

and cover all aspects of sustainability. Examples include the risk of the energy provider reaching its supply limits or the support of academic institutions to ensure sufficient skill availability. In *AI* acquirement, procurement and operation enablement need to be reviewed to assure they are balanced in economic, ecological and social parameters. With a stakeholder focused business approach this is not just within operational limits but also include suppliers and contractors, e.g. ensuring the amounts or types of hazardous materials in the supply chain are kept to a minimum. *ME* will have to ensure compliance with external requirements by assessing all stakeholder pressures—only then will it be able to achieve its full sustainability potential.

Figure 3 shows that CobiT draws from and influences objectives outside the framework itself. Sustainability goals and values have to be agreed on by top-level management making them part of the business objectives. These lead to governance objectives catering for sustainability being the main input factor for CobiT. This top-down approach is beneficial to sustainability improvement. A study of

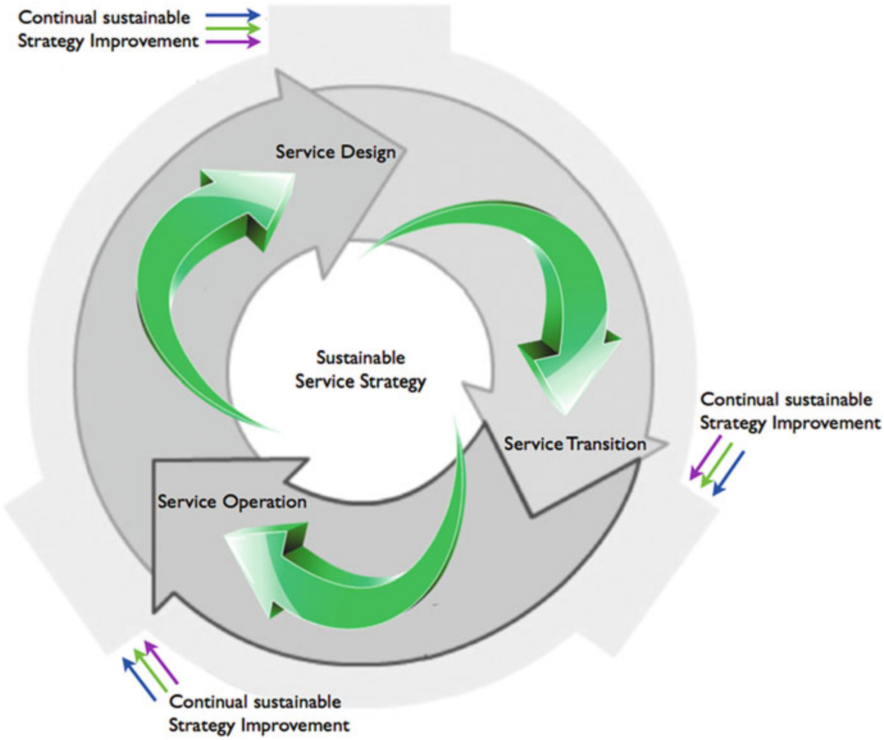


Fig. 4 Sustainability extended ITIL

179 large US firms shows that sustainability strategies are typically top-down [28]; the most effective being those when top management is clearly committed to the strategy.

3.4 Information Management: ITIL

ITIL aims to improve the overall QoS of the business within imposed cost constraints whilst improving the overall effectiveness and efficiency of IT. It needs to extend its orientation by ecological as well as social constraints instead of focusing on financial objectives. Similar to CobiT, ITIL has to assure that sustainability principles are catered for in its core process. As process grows on process, the objectives lead to a “greening” of the entire process cycle.

The ITIL process cycle (Fig. 4) undergoes constant iterations through *Continual Strategy Improvement*. This process is profit driven and therefore needs to expand its orientation to the TBL. Although the ITIL framework describes an iterative, continuous process, it still takes an initial top-down approach *Service Strategy* →

Service Design → *Service Transition* → *Service Operation* → *CSI*. *Service Design* process shows the lack of a holistic sustainable perspective in ITIL. The "4Ps of Service Design"—People, Processes, Partner, Products—indicate a social or stakeholder view, yet the driver here is profit; the effectiveness of the service rather than stakeholder value creation.

3.5 Project Management: PRINCE2

PRINCE2 takes a industry-independent perspective that follows an approach neutral to the resulting product described as "management by exemption". It is not effected by sustainability values; however, the project input and the organisation need to cater for sustainability. However, if senior management only take action triggered by exemption, then this leads to a requirement to integrate specific controls for sustainability in PRINCE2's *Controls* and *Plans* components as well as the underlying product-based planning process. Another component that needs attention is *Organisation*. As organisations evolve to more sustainable practices, they undergo change [16, 17] which needs to be reflected in the projects as well (Fig. 5).

One of the core principles of PRINCE2 is the ongoing reviewing and focus on the project's business case. Projects lacking a valid positive business case are not undertaken. Furthermore, the initial business case is being reviewed as part of each single step in the framework's process flow. Thus, PRINCE2 is subject to the same challenges considering the business case for sustainability as COBIT and ITIL.

3.6 Summary

This section has argued that integrating sustainability into IT Management Frameworks requires both the definition of a business case for sustainability and the integration into multiple frameworks. The requirement to integrate sustainability into multiple IT Management Frameworks is because there is no individual framework that covers all aspects of IT Management. It has been illustrated that sustainability can be added to the most popular IT Management Frameworks.

4 Analysis

Sustainability is not just about managing in an environmentally friendly way, but also about contributing to the long-term well-being of organisations, individuals, society and future generations [25]. In their individual characteristics each framework is affected differently—horizontally as well as vertically. There is no common

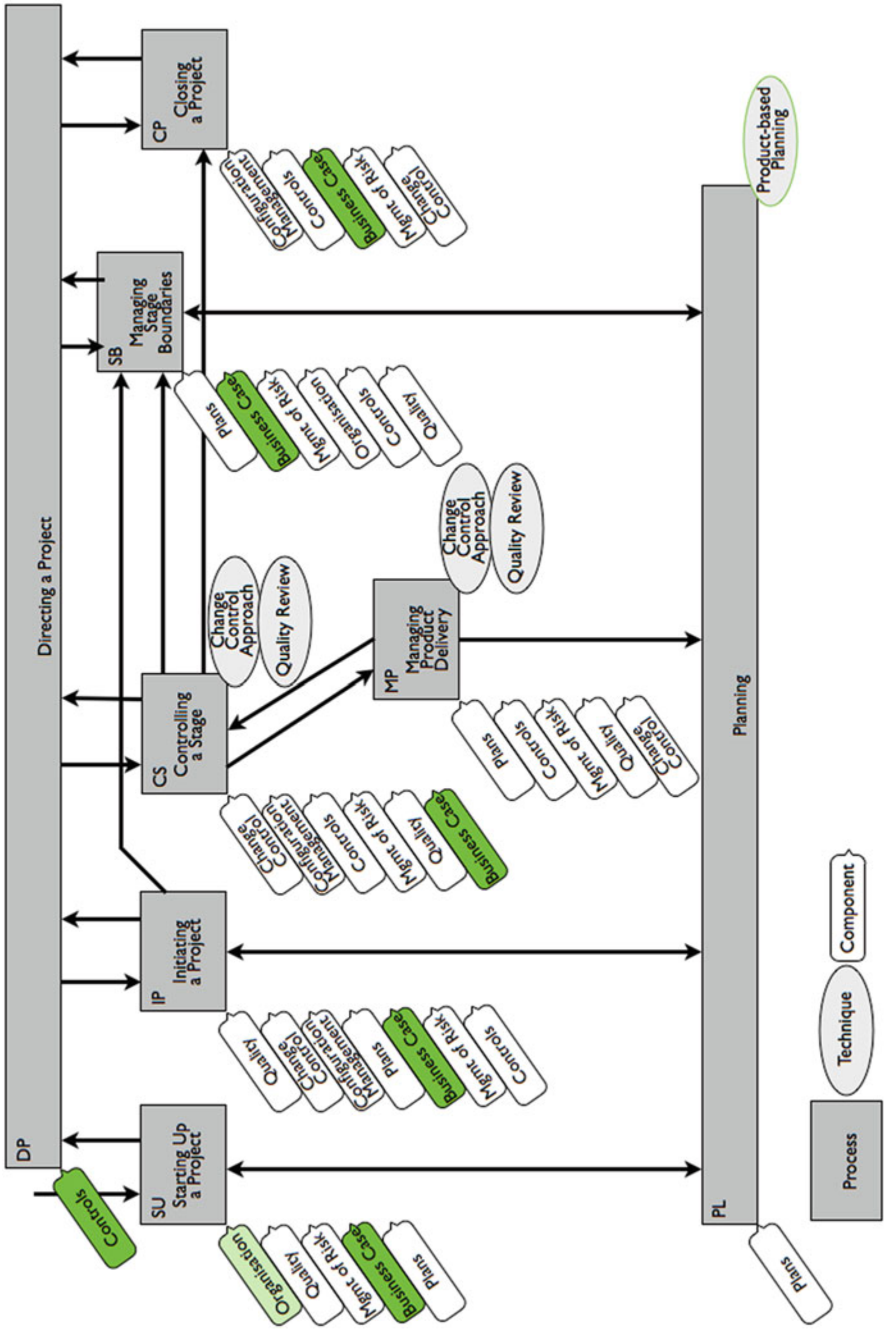


Fig. 5 PRINCE2 processes, components and techniques. Highlighted are those that require attention by concepts of sustainability (adapted from [20])

approach for all IT Management Frameworks; each framework needs to be analysed and extended individually. While some valid business case for sustainability activities can be expressed in testable business metrics, other cases will need to rely on convincing arguments and solid reasoning. The frameworks analysed in this study demonstrate their versatility and high measure of flexibility.

Sustainability can be integrated without redesigning the entire framework. All frameworks analysed need to extend their perspective beyond a profit-driven perspective to ensure compliance with ecological, ethical and societal requirements. There is a need for sustainable business objectives and governance objectives from the organisational structures. This needs to be addressed by extending the governance drivers from a rather regulatory focus to consider all stakeholder's demand, e.g. interacting with local environmental groups in strategic discussions and acknowledge societal goals such as emissions targets.

Governance frameworks such as CobiT gain from a change in business objectives, leading to extended governance objectives. CobiT draws on these and ensures that environmental and societal policies need to be catered for just like economic aspects. As successful sustainability is lived top-down, a Sustainable-CobiT as well as a Sustainable-ITIL have potential to accelerate the business transformation.

Integrating sustainability into the IT Management of an organisation is more than just extending the current frameworks; there are implications for the organisation to further develop these changes down to operational level. Substantial effort has been expended on the implementation of IT Management Frameworks, which will need to be reviewed, updated and extended to reflect the change of focus towards sustainability. This involves updating policies, operational models, software support packages, management systems and management practices. This transformation includes the need for change in the organisational structure and HR skill set.

For these goals to be viable, project definitions need to include the business case for sustainability in normal planning. These can be social improvements including employee childcare facilities which can reduce staff turnover and make the company more attractive to university graduates, improve brand image and employee productivity. A new focus on sustainability issues and business goals leads to a requirement for measuring of business activities in areas not traditionally measured and monitored. New data needs to be collected across the organisation, e.g. the volume of hazardous substances present in the IT hardware assets or greenhouse gas emissions per service. This data will advise the continuing business planning process.

The supply chain (suppliers and subcontractors) and the life cycle of the organisation's products must be considered. This can lead to new reasoning in the choice for suppliers s.a. electricity providers and hardware vendors, in some cases to a change of the business partner. Communication and interaction with the public become increasingly important as decision makers—including CIOs and IT managers—need to balance and mitigate between economic goals and diverse stakeholder interests.

5 Conclusion

This chapter argued for the need for the integration of sustainability with IT Management to allow organisations to address the demands of a sustainable future. It has demonstrated with popular IT management frameworks that although they do not take into account ecological or social aspects, they do not stand in the way of organisations transforming towards a more sustainable way of doing business. Furthermore, it has shown that the concept of sustainability affects every framework in a different way, demanding a integrative approach opposed to extending elements catering for the additional social and environmental aspects. Practitioners benefit from the discussion of the operational implications of a sustainability integration and the implementation of such modified frameworks. For the researcher, the effect of sustainability aspects in IT management has been demonstrated through the high-level integration into the representative frameworks of Six Sigma, CobiT, ITIL and PRINCE2. However, further research is needed how the notion of sustainability affects every level of each framework. An in-depth analysis towards a generic approach can further leverage the acceptance in the IT industry and academia alike.

References

1. APM Group Limited (2011) What is ITIL? Website. URL http://www.itil-officialsite.com/AboutITIL. Accessed 15 Dec 2011
2. Boccaletti G, Löffler M, Oppenheim JM (2008) How IT can cut carbon emissions. McKinsey Quarterly, 1–5
3. Burritt R, Hahn T, Schaltegger S (2002) Towards a comprehensive framework for environmental management accounting—links between business actors and environmental management accounting tools. Aust Account Rev 12(27):39–50
4. businessballs.com (2011) Six sigma. URL http://www.businessballs.com/sixsigma.htm. Accessed 12 Nov 2011
5. Dyllick T, Hockerts K (2002) Beyond the business case for corporate sustainability. Bus Strateg Environ 11(2):130–141
6. Elkington J (1998) Cannibals with forks: the triple bottom line of 21st century business. Capstone Publishing, Oxford
7. Epstein MJ (2008) Making sustainability work: best practices in managing and measuring corporate social, environmental and economic impacts. Berrett-Koehler Publishers, San Francisco, USA
8. Esty DC, Winston AS (2009) Green to gold: how smart companies use environmental strategy to innovate, create value, and build competitive advantage. Wiley, New York
9. Harry M, Schroeder R (1999) Six Sigma: The Breakthrough Management Strategy Revolutionizing the World's Top Corporations. Doubleday, New York
10. IT Governance Institute (2011) Cobit. URL http://www.isaca.org/cobit. Accessed 10 Dec 2011
11. IT Governance Institute (2007) Cobit 4.1 excerpt and executive summary
12. Kumar R, Mieritz L (2007) Conceptualizing 'green' it and data center power and cooling issues
13. Limited AG (2011) What is PRINCE2? URL http://www.prince-officialsite.com/AboutPRINCE2. Accessed 15 Dec 2011

14. Maes R (1999) A generic framework for information management. Department of Accountancy & Information Management, Universiteit van Amsterdam, Amsterdam
15. McKinsey (2009) It's carbon footprint
16. Molla A, Cooper V (2009) Green it readiness: a framework and preliminary proof of concept. Australas J Inf Syst 16(2)
17. van Osch W, Avital M (2010) From green it to sustainable innovation. In: AMCIS 2010
18. Porter M (2003) Csr a religion with too many priests. In: Interview with Mette Morsing, European Business Forum, vol. 15
19. Pyzdek T (2003) The six sigma handbook. McGraw-Hill, New York
20. Rozemeijer E (2007) Frameworks for IT Management. Van Haren Publishing, Netherland
21. Schaltegger E, Wagner M (2008) Managing the business case for sustainability. In: EMAN-EU 2008 conference, p 7
22. Schaltegger S, Bennett M, Burritt, R (2006) Sustainability accounting and reporting, vol 21. Springer, Berlin
23. Schmidt NH, Erek K, Kolbe LM, Zarnekow R (2009) Towards a procedural model for sustainable information systems management. In: 42nd Hawaii international conference on system sciences
24. Shrivastava P (1996) Greening business: profiting the corporation and the environment. Thomson Executive Press, Cincinnati
25. Thatchenkery T (2010) Positive design and appreciative construction: from sustainable development to sustainable value, vol 3. Emerald Group Publishing. Bingley, UK
26. UN (1987) Report of the world commission on environment and development: our common future
27. United Nations (2005) 2005 world summit outcome
28. Wisner PS, Epstein MJ, Bagozzi RP (2006) Organizational antecedents and consequences of environmental performance, vol 3. Emerald Group Publishing Limited, Bingley, UK, pp 143–167
29. Zadek S (1999) Stalking sustainability. Greener Manag Int (26):21–31

Improving Efficiency of Managing Projects by Quality-Oriented Modelling of Processes

Michał Kuciapski

Abstract Decades of evolution in the field of information systems, business processes and universal modelling notations has resulted in the creation of modern modelling languages, such as UML or BPMN. Elaborated languages concentrate on processes as activity flows without taking into account important management categories like quality or risk. These aspects permit the research thesis that there is a room for further modelling improvement especially in the field of quality management. Thus the article's main objective is to propose modelling approach for quality management of processes that supports better quality management in projects. It starts by introducing the current state of modelling notations. The introduction is a starting point for the presentation of elaborated quality-oriented notation for the project management processes discussed in the second section as a separate diagram type. A sample model is also included, based on the notation developed and showing its capabilities. The third part of the article presents verification of elaborated approach for quality-oriented modelling of processes. The article concludes with a summary.

1 Introduction

As opposed to the analysis and design of information systems dominated by UML (Unified Modelling Language), there can be no single denoted key notation in the discipline of process modelling. In many situations, organisations leave methodological and notational aspects to the contractor, even though some methodologies

M. Kuciapski (✉)
Department of Business Informatics, University of Gdańsk,
Jana Bażyńskiego 1A, Gdansk, Poland
e-mail: m.kuciapski@univ.gda.pl

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_22,

used by companies predate the notion of process modelling and most approaches employed are niche in character.

A strong impulse in the development of modelling approaches occurred in the 1990s. This was a natural consequence of organisations' reorienting in the direction of process management and solutions were elaborated for use in business process management. An exception, however, was UML as a standard for general use. Modelling approaches connected with IDEF (Integration Definition methods) [3] were widely practised. Also visual modelling approaches like ARIS (Architecture of Integrated Information Systems), with its characteristic diagram eEPEC, began to have more impact due to ARIS integration on the part of the company SAP into the SAP R/3 ERP system [8]. Another popular modelling system is BPMS (Business Process Management System) implemented by BOC into ADONIS business process management software. The first efforts to unify visual modelling techniques were conducted by the OMG (Object Management Group) consortium [4]. Research and design was concentrated on development of UML in the field of business process modelling [1,7]. Another modern approach considered as a candidate for wide adoption is BPMN (Business Process Modelling Notation) [5], which, unlike UML, has a very specialised character dedicated to business process modelling.

All the above modelling notations concentrate on modelling processes as a flow of activities with distinguishing decision-making aspects. Some integrate notation, allowing modelling document flow (UML and BPMS) and workflow (BPMS), but they still do not take into account important management categories distinguished by PMI (Project Management Institute) like scope, time, costs, quality, control, risk, communication or procurement [2]. These are all especially important in processes connected with project management. Integration of risk management into notation in BPMS and UML shows that organisations responsible for business process modelling are aware of current notation restrictions. Nevertheless, new notation elements can only be exclusively acknowledged as the initial one on a general stage [9].

Risk and quality are suggested as key aspects of project management [12]. Risk management modelling has been presented in separate article [10] as a general modelling concept in the form of an extension for modern notations like BPMN and BPMS [11]. It also has been recently included in UML as extension diagrams. Quality management notation in popular modelling approaches has still not been elaborated. According to The Bull Survey, one of the major causes of project failure is lack of or poor quality control [6]. In 35 % of projects analysed, weak, improper quality management was identified as a failure criterion. Practical project management experience of article's author shows that lack of integrating complex specifications of quality management for activities often causes a cursory quality management in projects. Together with The Bull Survey results, this indicates the importance of proper, systematic and unified quality management of project processes with appropriate support from modelling notation in designing process specifications. None of the modelling approaches presented and analysed here contains diagrams for managing process quality in the projects. So integration of quality management in processes is also omitted. Thus the main objective of the paper is to propose modelling approach for quality management of processes that supports proper quality management in projects.

2 Quality-Oriented Modelling Approach in Managing Project Processes

Using the quality-oriented modelling approach to manage projects, with a strong emphasis on integrating it with business process models, should be an important element in eliminating project failures due to no or poor quality control. Lack of any notation for complex, universal quality management modelling with a close connection to business models or project processes demanded the elaboration of author notation from scratch. Complexity and the number of notation elements connected with quality management did not allow integrating them directly as an extension of popular UML or BPML diagrams of activities (UML) or business process (BPML).

As a key element of quality management modelling, quality management models were assumed to include models of processes. To achieve such a result, the correct stereotypes for business process diagrams were elaborated (Table 1) for various management categories.[1]

As solution to integrate quality management modelling with processes, stereotypes for activities were used. Such integration of quality management is possible for the BPMS by assigning the correct symbols to activity objects. In this way, the correctly ascribed icons fulfil the role of stereotypes for activity objects thus extending their meaning. Such an approach permitted both the integration of quality management modelling with process models and quick reference in locating detailed quality control diagrams connected with particular processes. Figure 1 presents an example of such integration for the process *requirement analysis of e-learning course development* which is the starting process for e-learning course development and implementation projects. Integration of quality management stereotypes also allows for verification as to whether the correct number of activities exists that contain actions for ensuring the requirements of intermediate components and the final product. In example diagram for e-learning projects, such elements are course

Table 1 Notation stereotypes used for identifying management categories

Symbol	Description
	Quality management
	Risk management
	Communication management
	Cost management
	Time management
	Control management
	Resource management
	Document flow management

[1]For all management categories included in Table 1 modelling notation was elaborated, but the paper concentrates only on quality management modelling as justified in the first part.

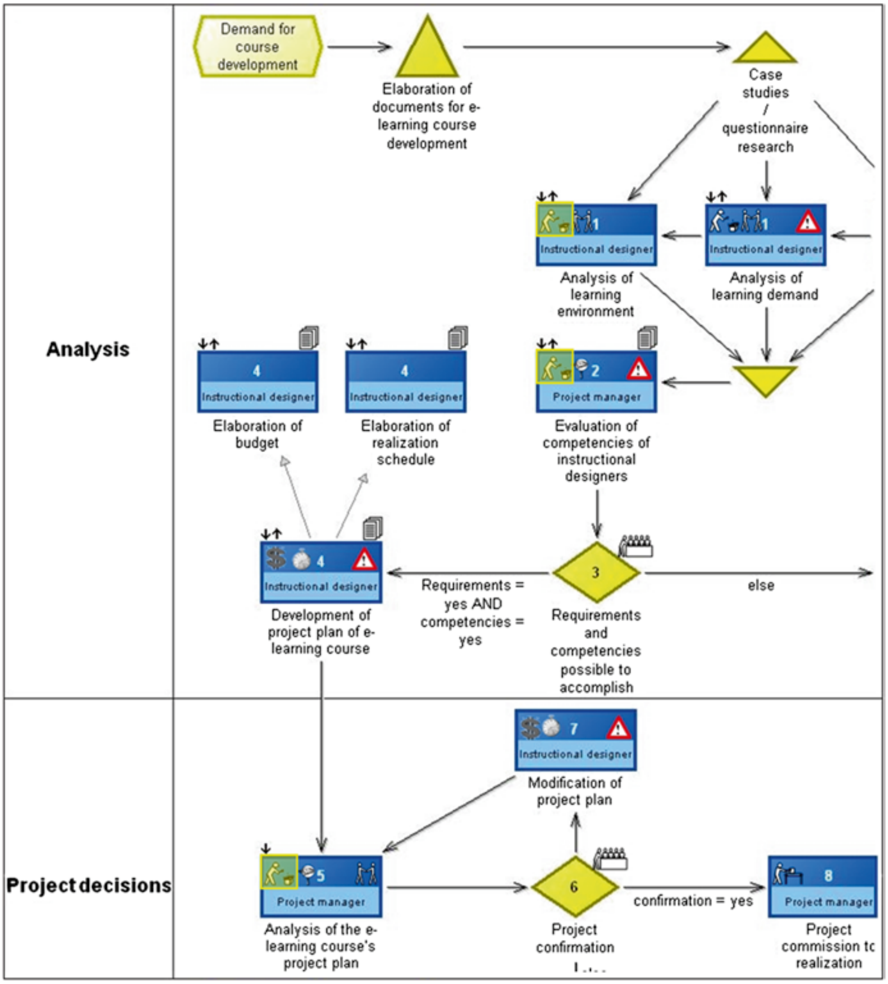
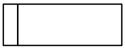
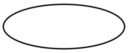
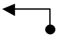
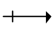
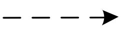


Fig. 1 Process model with integrated quality management—requirements analysis of e-learning course development

outline, e-learning course script, e-learning course instructional design, multimedia objects, authoring tool components and, as a final product, e-learning course implementation packages. The elaborated modelling approach thus demands quality-oriented modelling of project's processes.

The modelling approach used for the integration of quality management with process models was also used for other important project management categories such as risk, resources, document flow or communication. Developed notation was based exclusively on the present author's experience and interviews with project managers. As a new concept it required appropriate verification presented in the third part of the article. Quality management modelling is based on dedicated diagrams consisting of the visual notation elements described in Table 2.

Table 2 Notation for quality management modelling of business processes

Symbol	Name	Description
	Activity	Activity connected with quality management and related by name to the business or project process model
	Coordinator	Person responsible for actions of quality management within the framework of the activity
	Participant	Role responsible for supporting actions of quality management
	Action	Action undertaken for quality management of the activity
	Method	Quality management method applied during activity implementation
	Result	An element, usually a document, which is the result of quality management for the action
	Manages	Connector that assigns the coordinator to the activity for which tasks related to quality management are executed
	Participates	Connector that indicates the roles involved in the actions of quality management for the activity
	Process route	Path of quality management realisation for the process

Elaborated modelling notation was used for development of a complex, integrated project management model for the production and implementation of e-learning courses as case study example. A sample quality management diagram elaborated with the use of developed notation, for the process of the *requirements analysis of e-learning course development*, is presented in Fig. 2.

Quality management in e-learning projects is mainly connected with activities related to assessment, training, consultation and verification. Thus, quality management in the process of the *requirements analysis of e-learning course development* refers to activities such as requirements analysis of e-learning course or evaluation of competencies of instructional designers. The aim of these activities is to provide evaluation data of requirements of e-learning course development and possibility of course creation and implementation.

Each activity has at least one action identified in the field of verification or raising quality, such as comparing the competencies of instructional designers with e-learning course development requirements. For each action, input elements were identified, usually in the form of documents to be reviewed or consulted. For those activities highlighted and dedicated to quality management of development projects for e-learning courses, the following are also defined: alternative methods of quality management execution, coordinator as the role responsible for supervising activity implementation and, lastly, participants supporting the coordinator. Activities are accompanied by results, mainly in the form of previously developed and improved components or as a list of changes, such as competencies evaluation report in Fig. 2.

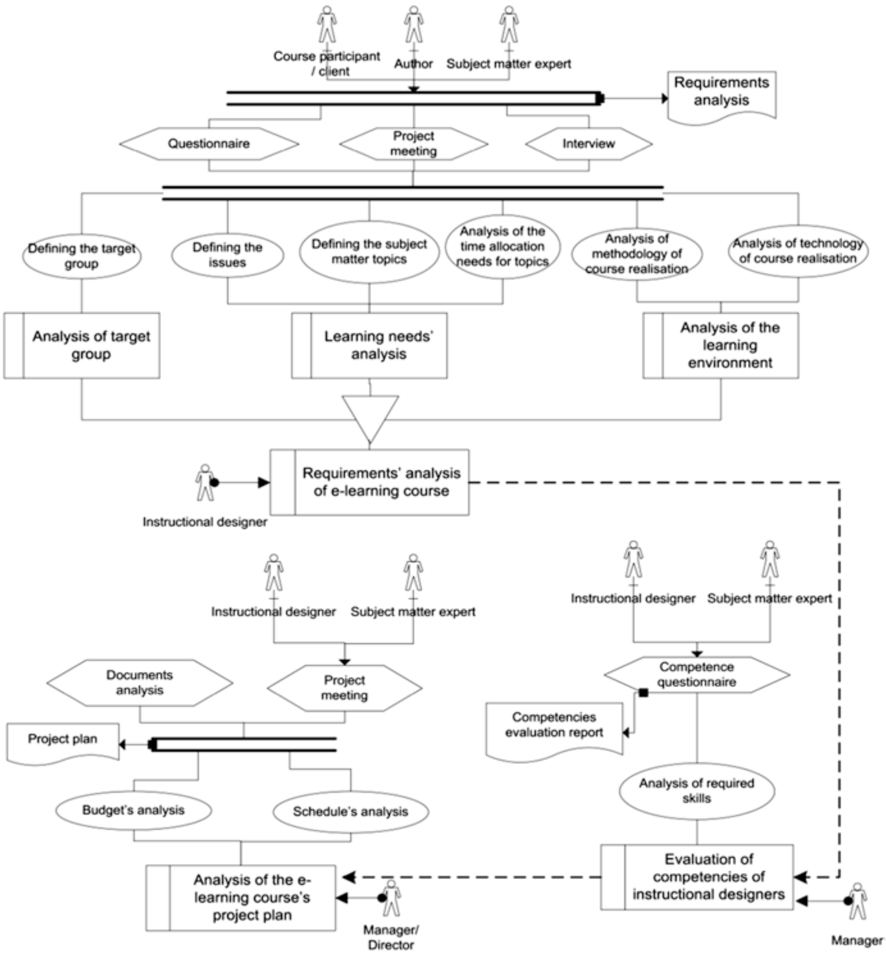


Fig. 2 Quality management diagram for the process of requirements analysis of e-learning course development

Appropriate quality management diagrams were elaborated for all processes outlined in the e-learning course as case study example: requirement analysis, elaboration of script, instructional design production, evaluation and implementation, evaluation and (periodical) revision.

Presented, elaborated notation allows the development of diagrams that visualise, in a complex manner, the concepts connected with quality management for a sequence of activities identified on process flow diagrams. It required verification of assumptions for positive impact of elaborated quality-oriented modelling approach for business and project processes.

3 Evaluation of an Elaborated Quality Process Management Modelling Approach for Processes

Proving the usability of an elaborated quality-oriented modelling approach for processes was only possible by implementing it for specifying processes of real-word projects. Such approach was carried for developing model for managing e-learning projects. It occurred that none additional notation elements neither modifications were required for modelling quality of processes in projects. It indicated the completeness of elaborated quality-oriented modelling approach and its usability.

Next evaluated element of developed quality-oriented modelling approach was whether it assists in distinguishing important quality management activities during designing project management models. Therefore verification of impact on quality management of developed model for e-learning projects was conducted in three areas:

- Compliance of projects with budgets and schedules by comparing actual and planned schedules and budgets
- Time required for correcting errors connected with design and production processes, through analysis of quantitative data collected while monitoring the implementation of e-learning projects
- The quality and attractiveness of the e-learning courses prepared and implemented by means of a survey carried among course participants

Verification was conducted for six projects of e-learning course development and implementation at the University of Gdańsk between 2008 and 2010. For three projects, management was based on the elaborated quality-oriented model presented in the second part of this paper. The process for the other projects was established on project manager's experience and the general project management model developed by PEUG.

Verification of project compliance with budgets and schedules was checked by a comparison of projects realised using the elaborated model and those without it. For projects based on project manager's experience and the basic model, serious diffractions were noticed both in the initial stage and in the duration of the process. In one of the projects, time for implementation of the processes such as design, production and implementation exceeded the schedule by 141, 75 and 120 days, respectively. Haste in finishing the e-learning course with the plan and script author's requirements caused the production of too many low-quality multimedia objects and was the main reason for the refusal of final acceptance in executing implementation. This caused serious negative consequences:

- Rerunning of processes like design and implementation
- High time outlays for correcting multimedia and learning objects
- Rescheduling of the course from winter to summer semester
- Transfer of the execution of one project to a different manager with the use of an elaborated quality-oriented model

Detailed analysis has revealed that the main reasons for project failure were connected with incorrect risk and quality management. In the quality management area, project irregularity was related to the fact that many activities included in the model explicitly to prevent inappropriate quality management were passed over. For the design process, they related to substantial consultations for designing multimedia objects and to the evaluation of a list of planned multimedia objects with their initial concepts. The lack of such elements caused many faults even on the specification level of the e-learning course. It was also the reason for the choice of inappropriate script parts for multimedia adaptation according to author requirements. This led to a need for additional multimedia object development with a different instructional design approach. Furthermore, actions connected with the verification of implementation for multimedia objects and their conformity with projects was not carried out and only took the form of general check on specification completeness. Together with design process faults, this brought about the preparation of many low-quality multimedia and learning objects with a number of serious errors.

E-learning course projects realised with the use of a quality-oriented modelling of project processes approach had only minor diffractions from the schedule. Two projects were executed precisely as planned in the schedule and budget. One had insignificant diffractions only in the scope of the duration of particular actions and not their starting dates. Non-adherence to the planned schedule was caused by projects being realised without the use of a prepared model, which had a great impact on the production processes of all e-learning course projects. Nevertheless, the use of an elaborated model for project management of e-learning course development and implementation, founded on strong integration of quality management modelling, with processes and activities also being supported by proper risk and communication management (modelling), allowed projects to be successfully completed.

Quality-oriented modelling of business processes was also evaluated in the field of time requirements for correcting errors connected with the design and production processes, through analysis of quantitative data collected during e-learning project realisation. Fault monitoring was carried out on multimedia objects as they are the most labour-intensive element. As with schedule and budget research, analysis was based on comparing e-learning projects realised with the use of the model based on elaborated modelling approach and those that had not applied it. Multimedia object verification was carried out both by instructional designers and authors. The verification conducted by instructional designers during the processes of design and production indicated that, regardless of the use of an elaborated quality-oriented model for e-learning project management, the percentage of multimedia objects with errors was the same (57 %) as presented on Fig. 3.

For the requirements of multimedia object error analysis, a category of defect complexity was interposed. In this way, categories of corrective action complexity for multimedia objects were distinguished based on estimated time of implementing correction: very low (up to 15 min), low (changes between 16 and 60 min), moderate (between 61 and 120 min), high (between 121 and 240 min) and very high (over 240 min). In accordance with the results in Fig. 3, e-learning course multimedia objects developed with the use of the model display very low or low complex error in contrast to e-learning courses prepared with the use of general models and project manager's experience. Thus the quality-oriented model for the processes of

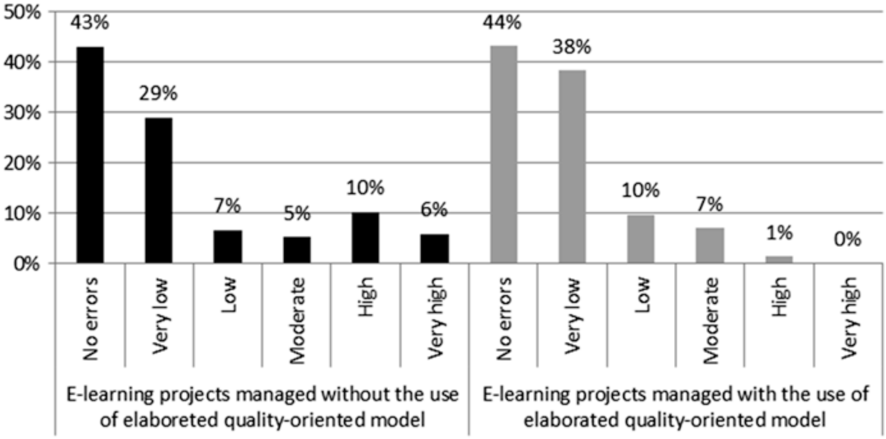


Fig. 3 Assessment of the level of errors in the process of design and production of multimedia objects conducted by instructional designers

management of e-learning course projects helps in reducing the complexity of errors in the multimedia objects developed, thereby greatly eliminating the time and expenditure required for remedial actions.

Verification of the quality of multimedia objects was also conducted by the e-learning course script authors and showed a much worse assessment of media objects prepared without the use of an elaborated model in which only 11 % of objects were error-free. A significant amount was noticed by the authors and was often also extremely complex, requiring difficult modifications with large time and financial expenditure. In contrast, 85 % of multimedia objects where the processes of design, production and implementation had been realised with the use of elaborated quality-oriented project management models had no errors listed during verification by the authors. Detailed analysis exposed the occurrence of a significant number of errors was associated with inadequate management of the project in the fields of quality (lack of systematic evaluation processes and of standardisation in the technical design of object specifications), risk (no time-consuming or cost-consuming analysis for the preparation of multimedia objects) and communication (insufficient consultation within project team). This resulted in negative assessments of the e-learning courses by their authors and the need to introduce several amendments, which significantly delayed the completion of the projects and had serious negative impact on efficiency of project realisation. Many elaborated elements modelled based on presented in the article notation, integrated into the developed model for process management of e-learning projects ensured a lower number of errors in the multimedia objects produced, thus reducing the time and expenditure required for remedial action.

The last evaluation element in the elaborated quality management modelling approach for processes of e-learning project management was conducted in the form of a survey among course participants, consisting of 47 questions investigating the quality of e-learning courses in relation to the processes of requirement analysis, script elaboration, design, production, implementation and realisation. One hundred and two surveys were received. Research was conducted by comparing the results for e-learning courses with and without elaborated quality-oriented project management models on the basis of the formula:

$$Average\ for\ questions = \frac{\sum_{1}^{n} Course\ averages}{Courses\ numbers},$$

where n indicates the project management approach used, based or not based on an elaborated model.

Synthetic analysis of survey results conducted among participants has allowed the formulation of general conclusion of better assessment of e-learning courses prepared on the basis of an elaborated model in all fields such as arrangement and completeness of topics presented in e-leaning courses, attractiveness of the presentation approach for material used as an e-learning course and concepts used for adaptation of static material as multimedia and interactive learning objects.

The research conducted supports opinion that the use of a quality-oriented model allows for better realisation of processes according to schedules and budgets, limits the number of errors committed during various processes and helps to increase the quality of the end products as e-learning courses. Also conducted researched showed that using elaborated approach supports better quality management for business and projects processes by inducing focusing on:

- Integrating quality management on activities level and not only general one
- Stimulating quality management thinking about processes in the field of distinguishing missing activities dedicated for quality management

Also not taking into account quality-oriented modelling of processes has been evaluated for projects that did not use elaborated approach. It proved in such situation there often occurs lack of many activities dedicated for quality management of processes.

4 Summary and Conclusions

The present study showed the concept of quality-oriented modelling of processes for effective management of projects. As a starting point, a review of modelling notations and systems was offered with an outline of their weaknesses. Analysis proved that none of the modelling approaches presented and analysed contained diagrams for managing quality in projects. This meant that the integration of quality management with business processes was also omitted. Lack of quality modelling is an important aspect in process management, as according to The Bull Survey one of the major causes of project failure is no or poor quality control.

Lack of available modelling approaches for quality management, taken in conjunction with the failure to include aspects associated with them, was a cause to propose original modelling notation in this field, presented in the second part of the article. Elaborated notation was used in modelling processes for project management. One of the key factors was appropriate integration of quality management with processes, which was achieved by using stereotypes for activity objects. An elaborated, integrated modelling approach enabled the development of a quality-oriented model for managing the processes of an e-learning project as case study example.

The article concluded with an assessment of a quality-oriented approach for modelling processes. Verification included evaluation of completeness of elaborated notation. Also analysed was its usability for project management model development that supports proper projects realisation according to budgets and schedules. The results of the research confirmed that the use of models based on elaborated quality-oriented modelling approach allows for better realisation of processes according to schedules and budgets, limits the number of errors committed at various stages of the projects' execution and helps to increase the quality of end products like e-learning courses. In this manner, it enables higher efficiency for managing projects. It is necessary to point out that analogous verification should be conducted for other than e-learning projects to fully verify modelling approach usability and capabilities.

References

1. Ambler W (2005) The elements of UML 2.0 style. Cambridge University Press, Cambridge
2. Baca C (2005) Project manager's spotlight on change management. Sybex, Almeda
3. Eriksson H, Penker M (2000) Business modeling with UML: business patterns at work. Wiley, New York
4. Eriksson H, Penker M, Lyons B, Fado D (2004) UML 2 toolkit. OMG Press, Indianapolis
5. Harrison-Broninski K. The future of BPM. Part 2. http://www.bptrends.com/publicationfiles/09-06-ART-FutureBPM20f6-Harrison-Broninski.pdf
6. http://www.it-cortex.com/Stat_Failure_Cause.htm#The Bull Survey 1998
7. Johnston S. Rational UML profile for business modeling. http://www.ibm.com/developerworks/rational/library/5167.html
8. Kalnins A, Barzdins J, Podnieks K. Modeling languages and tools: state of the art. In: Proceedings of second international conference on simulation, gaming, training and business process reengineering, 2000
9. Kuciapski M (2010) Model for project management for development and implementation of e-learning courses. In: Forbig P, Gunther H (eds) Perspectives in business informatics research, proceedings of 9th international conference, BIR 2010. Springer, Rostock
10. Kuciapski M. Risk management in e-learning projects of courses development and implementation, studies and materials of Polish Society of Knowledge Management, Congress of Young Scientists, ISBN 978-83-7518-245-3, Międzyzdroje 2010
11. Levine H (2002) Practical project management, tips, tactics, and tools. Wiley, New York
12. van der Aalst, WMP. Formalization and verification of event-driven process chains. http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.33.1015

A Performance Management Model for Agile Information Systems Development Teams

Garry Lohan, Michael Lang, and Kieran Conboy

Abstract The prevailing trend in ISD is one of poor project performances, with budget overruns commonly in excess of 300 % and many failing altogether. To address this trend ISD research always focuses on the ISD process, user involvement and the people involved. Rarely, if ever are wider organisational processes questioned. This paper argues for a cohesive and ongoing inclusion of wider organisational factors in efforts to address and improve ISD project performance. Given the poor budgetary performance of ISD projects, budgeting is one that we feel requires particular attention. Contemporary research in budgeting (e.g. Beyond Budgeting) and in ISD (e.g. agile methods) attempts to address similar issues albeit from a different angles. This paper draws on two case studies of seven ISD teams to apply the Beyond Budgeting model to an ISD environment. We demonstrate the value of using the Beyond Budgeting model to develop a cohesive research agenda within ISD used to identify gaps and suggest improvements to agile methods, probably the most well-known and accepted contemporary ISD approach.

1 Introduction

Despite over 40 years of ISD research, the statistics on project failure in information systems development (ISD) projects are still of significant concern. ISD projects often spiral out of control and far exceed original budget and schedule projections [24]. The Standish Group's 2003 CHAOS Report which incorporates data from

G. Lohan (✉) • M. Lang
National University of Ireland, Galway, Ireland
e-mail: garry.lohan@nuigalway.ie; michael.lang@nuigalway.ie

K. Conboy
The University of New South Wales, Sydney, Australia
e-mail: k.conboy@unsw.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_23,

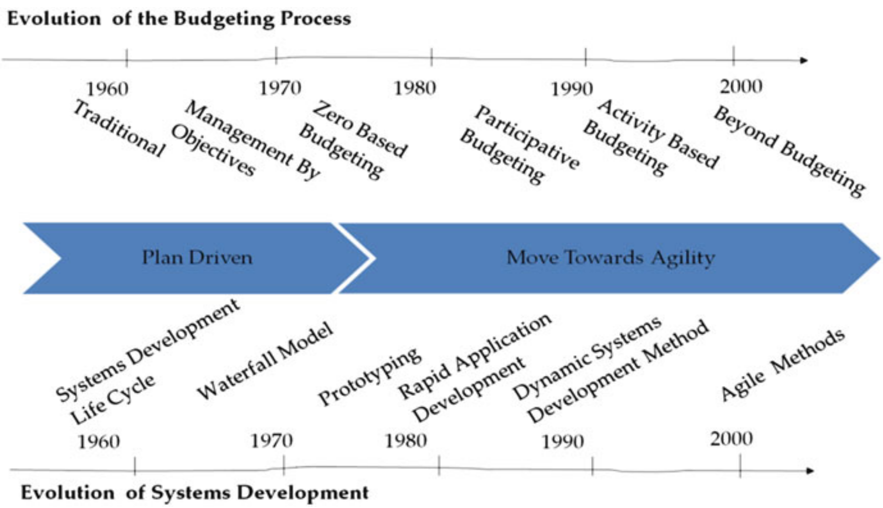


Fig. 1 The evolution of budgeting and ISD

several thousand projects suggests that 43 % of projects were over budget [36]. The 2006 CHAOS study update reveals that only 35 % of IT projects started in 2006 were categorised as successful, 19 % were judged to be outright failures and the remaining 46 % were completed over-budget, behind schedule or failed in some way to meet user requirements [38].

When we examine budgeting theory and literature, it is perhaps not surprising that ISD projects are continuously running over budget. Indeed it is clear that effective budgeting is something that has not really been mastered in any discipline. In its own literature, the budgeting process has attracted much criticism in recent years [19]. The traditional budgeting process has been labelled as "broken" [22], an "unnecessary evil" [40], a "thing of the past" [18], "ineffective and inefficient" [20], and many have questioned the value of budgeting as a management control mechanism in contemporary environments at all [6,13,35]. Contemporary budgeting research is attempting to fix these issues, and in recent years an innovation from the management accounting literature called Beyond Budgeting has received a lot of attention [6,19,20].

Researchers within ISD have highlighted the pressing need to keep up to date with developments in other fields that have a direct impact on ISD in practice [29]. One area that has a direct impact on ISD in practice is the budgetary control process. The evolutionary path of the budgeting process shares many similarities with the evolutionary path of ISD approaches (Fig. 1) with both moving towards a distinctly agile or flexible state [30,37]. It is particularly interesting that the suitability and performance of more contemporary approaches to budgeting are not being studied in contemporary ISD contexts. Rather than reinventing the wheel, we can extend the field of ISD by using contemporary budgeting approaches as a lens to examine contemporary ISD practices. Crossing the chasm into budgeting theory and literature offers an opportunity to explore contemporary ISD practices with a grounded reference to wider organisational contexts [1,2,23], thus answering calls for a more cohesive research agenda within the ISD field [8]. The objectives of this study are to address this research agenda, specifically to:

- Apply contemporary thinking in budgeting to an ISD context
- To demonstrate the value of such an application to extend the ISD body of knowledge

The next section of this paper briefly outlines the evolution of systems development and the evolution of budgeting, which culminate in the ASD and Beyond Budgeting concepts being introduced in the early 2000s. Section 3 describes the research methodology used in the study. Section 4 presents the findings and analysis, and Sect. 5 is a discussion outlining the implications for theory and practice.

2 Evolution of Systems Development and Budgeting

2.1 Evolution of Systems Development

Information systems development is arguably the core topic for the field of information systems. Early development in the 1960s and 1970s occurred without explicit or formalised development methods [4]. Although the exact origins of the traditional Information Systems Development Life Cycle (ISDLC or SDLC) are unclear, during the late 1970s and 1980s, it became an established concept and was widely used for systems development. Critics of the SDLC model point out that it is difficult to gather and understand a complete list of requirements at the outset of the development project [39]. Many viewed the SDLC as a troublesome, time-consuming process, and the call for a more flexible development approach has been around since the early 1980s [3,16].

It was the continued dissatisfaction with the available development methods that led to the introduction of the various agile approaches. The agile approach seeks to help address the key problems in software development, such as quality, time and cost [15]. While having conceptual roots dating from the early twentieth century, it was the formation of the Agile Alliance in 2001 and the publication of the Agile Manifesto (http://agilemanifesto.org) and principles behind the manifesto that formally introduced the term agility to the field of software development [9]. Agile methods include those that have entered into the spirit of agile. Among these methods XP and Scrum are the two most widely used in practice [14].

2.2 Evolution of Budgeting

Since the establishment of modern enterprise, there have been three major evolutions in the management and structure of organisations. The first took place between

Table 1 The beyond budgeting model

Leadership principles	Process principles
Customers: focus everyone on improving customer outcomes, not on hierarchical relationships	*Goals*: set relative goals for continuous improvement; do not negotiate fixed performance contracts
Organisation: organise as a network of lean, accountable teams, not around centralised functions	*Rewards*: reward shared success based on relative performance, not on meeting fixed targets
Responsibility: enable everyone to act and think like a leader, not merely follow the plan	*Planning*: make planning a continuous and inclusive process, not a top-down annual event
Autonomy: give teams the freedom and capability to act; do not micromanage them	*Controls*: base controls on relative indicators and trends, not variances against a plan
Values: govern through a few clear values, goals and boundaries, not detailed rules and budgets	*Resources*: make resources available as needed, not through annual budget allocations
Transparency: promote open information for self-management; do not restrict it hierarchically	*Coordination*: coordinate interactions dynamically, not through annual planning cycles

1895 and 1905 with the introduction of professional management which distinguished management from ownership and established management as work and task in its own right. The second took place during the 1920s when Taylor's "one best way" and Henry Ford's assembly line production introduced the command-and-control organisations with their traditional budgeting and control mechanisms. The third evolution sees a paradigm shift from command-and-control to information-based organisations employing knowledge workers and operating in an ever-changing knowledge economy [12]. Organisations can no longer rely on traditional budgeting and control mechanisms which were especially suited to a pre-information age era [32]. Researchers and practitioners in the field of management and management accounting have highlighted the issues with traditional control mechanisms such as the budget process and called for new and innovative approaches to managing in a knowledge-based economy [20,30,32].

While the traditional command-and-control management model based on a traditional annual budgeting process has been the main management control mechanism used to manage and control employees, the move to the knowledge-based economy of the information age has seen organisations search for more flexible management control models [6,20]. In the modern, turbulent and ever-changing business environment, organisations must develop a conscious agility competency in order to survive [11]. Beyond Budgeting, an innovation from the management accounting literature, suggests that in a knowledge economy, organisations must go beyond the budgeting process and manage through a series of flexible controls and processes (Table 1) [20].

The Beyond Budgeting model is upper management's answer to achieving the flexibility the software development community achieved through the formalisation of agile methods. It highlights an evolution in management thinking from traditional command-and-control to leadership-and-collaboration management styles and has an extraordinary conceptual similarity to agile methods [26,37]. The Beyond Budgeting model has previously been conceptualised relative to ASD [28]. By applying the Beyond Budgeting model in its entirety to an ASD environment, we begin to gain a more coherent understanding of how agile practices can be improved to foster agility in a wider context. The following section outlines the research methodology used in this study.

3 Research Methodology

Case studies are particularly suited to IS research and are increasingly used in the IS discipline [41]. We chose a case study method as the focus of our research is on contemporary events and the phenomenon is to be examined in a natural setting. Miles and Huberman [31] suggest outlining a strategy for the selection of informants prior to data collection. We used what they call a comparable case selection strategy which allowed for replication of the results and added to the validity and analytical generalisability of the findings. Yin [42] also suggests that having access to all relevant and required information is crucial to doing good case study research. Both organisations chosen for this study had implemented the Scrum methodology within the past 3 years and were eager to participate in the study. A confidentiality agreement was in place with both sites and all data was to be used for academic purposes only.

To establish the reliability and validity of the case study evidence, we followed the three principles of data collection outlined by Yin [42]:

Use multiple sources of evidence: Data was collected through on-site observation at iteration meetings and daily Scrums, review of documentation, three workshops and 19 formal interviews. As well as this, several informal interviews took place and a continuous dialogue was established with key informants through emails, phone calls and site visits.

Create a case study database: All formal interview transcripts were recorded and transcribed. The transcriptions were imported into QSR NVivo for coding. A tree structure of codes was developed using each of the 12 principles of the Beyond Budgeting model as the initial coding nodes. All notes, documents, interview protocols, and narratives were stored in this NVivo database.

Maintain a chain of evidence: A clear link was established between each step of the process. The case study objective was linked to the interview protocol questions, which are linked to the evidentiary sources in the NVivo database, which are in turn linked to (a) the case study reports provided to the participating organisations and (b) the findings discussed in this paper.

Data analysis was performed following established standards [31]. The Beyond Budgeting model provided a list of 12 seed categories for initial open coding. Data were partitioned into different sets or cases in NVivo to allow comparisons across cases. Documents created from cases were compared to reveal patterns in the data. Follow-up phone calls, emails and site visits were arranged where possible and further documentation obtained when further information was needed or clarification was required. Precautions were taken to corroborate the interpretations made. Findings were discussed continuously with key informants in each of the case sites. Emerging categories were checked for representativeness by examining them across participants, for example, team members' reports of their experience with their customers were checked against the reports from other team members and the project managers or Scrum masters. The use of case nodes helped with cross case comparisons and provided external validity through analytical generalisation [42].

4 Findings and Analysis

By applying the Beyond Budgeting model to an ASD environment, we find that there are many areas where ASD may be extended. We demonstrate the value of the application by presenting the findings under each of the 12 headings of the Beyond Budgeting model and outlining a set of recommendations (represented by *R*) based on an analysis of the findings.

Customers: In the teams we studied, we found that the interface between the team and the customer was a source of potential problems. The role of the customer was played by either a customer representative or the product owner, and this customer proxy acted as both a conduit and filter for information and ideas. Some developers felt that their suggestions were being filtered by the customer proxy, while others felt that they did not receive quality and timely requirements. One developer points out that "*the proxy customer very much said she wasn't fully aware of everything the customer is going to do*". However, some teams were very happy with the relationship they had with their customer as one developer says "*there are some customers who are really eager, really involved, they really know the area and they know the tool*". This highlights the importance of the role the customer or customer proxy plays in ASD and the need to have informed and trained customer proxies (*R1*). All of the teams studied had specific project repositories where they could store and retrieve project information. However, there were no knowledge repositories specifically for customers. Research in other fields has shown that having specifically designed customer knowledge repositories (*R2*) helps develop a better understanding and relationship with customers [17]. Given the importance of the customer to ASD, it is interesting to note that this was not considered in any of our case sites.

Organisation: Our findings suggest that agile practices give teams a large amount of control over their daily working practices as one developer says: "*The team is pretty much allowed to do whatever we feel would improve the quality of the code, the* *quality of the process, etc.*" However, the team has minimal input into decisions such as hiring new team members, what review process should be used and what methodologies, technologies and tools can be used. In both sites, however, we found that developers used statements such as "*that was decided way up*" [from a developer in Case A when asked about technologies and tools they could use] and "*that was mandated from farther up the food chain*" [from a developer in Case B when asked about why Scrum was introduced]. In both sites we found that top management support and understanding of an ASD environment (*R3*) was lacking. One project manager expressed concern that current legacy organisational processes do not show "*much understanding of personal qualities, dynamics in teams, competences built over time*" with another project manager agreeing that "*we are still not there in terms of complete buy in from the management using scrum*". In fact, in all seven projects we studied, all the project managers expressed concern about a lack of understanding of how agile teams function.

Responsibility: It was interesting to note that both of our case sites had single fixed project managers not the rotating shared leadership role promoted within the ASD literature [34]. This worked well in practice as all team members felt that their immediate management provided a supportive and enabling environment. Comments from developers such as "*the managers are generally very good, yes they do direct but they certainly wouldn't in my opinion micromanage*" and "*it is quite a supportive environment*" were the norm across both sites. Project managers themselves all agreed that they tried to create a "supportive" environment. This shows that rather than having a shared and rotating leadership, ASD teams can work well when an enabling leadership style is utilised by project managers (*R4*). Another reason why shared leadership is difficult is exemplified in Case B where consultants were used on a contract basis depending on the project needs. As the project manager points out, in this case it is difficult to have shared leadership as "*the consultants don't have the same kind of responsibility, they are responsible, but they are doing it according to a contract*".

Autonomy: There was agreement that teams had autonomy in their daily tasks, but some decisions were out of their control. Many decisions around the use of tools, technologies and methodologies, the hiring of staff, training budgets, long-term planning, etc. were made outside of the team. Developers again used phrases like *above our heads* and *at a higher level* when asked if they had input into decisions which would affect them. In Case A one developer believes that when decisions are made that affect the team, then it's explained to the team but not is a satisfactory way saying: "*Usually its explained but its explained as in its politics, and that isn't much of an explanation…it can be frustrating*". There was a feeling that the ASD teams were working within an agile bubble and they did not have sufficient support of the agile concept from mid and senior level management. Empowerment and autonomy are seen as essential components for agile development, and people feel comfortable when they have the environment and support they need [27]. By promoting an understanding of agile culture at every level of the organisation (*R5*), it will be easier to create the supportive environment ASD teams need.

Values: In both of our case sites, we found that the ASD teams operated within established flexible governance frameworks. This is facilitated in both sites by the development of project initiation plans which outlined operating guidelines within which the teams were expected to operate. These plans set out the high-level goals and milestones which the team is expected to aim for. One Scrum master in Case B highlights this by stating: "*The product owner has made an effort in creating the vision, the vision has helped a lot in regards to that*". This vision, which is incorporated into the project initiation plan, gives the teams their boundary operation conditions as well as their high-level goals, and the teams in both of our case sites found this a useful governing mechanism.

Transparency: There was a general consensus from all interviewees that the ASD methodology used provided them with sufficient information for daily operational tasks. Tools such as burn down charts, Scrum walls and the project management tools used provided the teams with accurate and up-to-date information. Teams could see at a glance the status of the project and this helped them with the management of their daily tasks. One developer from Case A recognises the value of the Scrum methodology when it comes to the team's self-management saying: "*You can see what everyone else is doing…. We update the project management tool everyday and this gives us pretty good visibility on where we are with the project at all times*".

Goals: In both sites team members felt that within the project duration, their Scrum master would informally speak with them regarding their short-term individual goals. Some senior stakeholders may have their own personal goals, but generally, the team is viewed as having a team goal. A Scrum master from Case B explains: "*The goals are the Key Performance Indicators* (*KPIs*). *Every single department has their own KPIs.*" In both case sites, the goal-setting process which involved the setting of both behavioural and technical goals was well regarded by the team members. Project roadmaps are already in place when a team is assembled and project milestones outlined. The team members see these as their main goals and within those boundaries they decide, as a team, along with the product owner their shorter-term goals.

Rewards: A reward system that rewards shared success is promoted by both the Beyond Budgeting and ASD literature. However, this was not how the reward systems worked in either of our case sites. The team members were reviewed individually by their managers and reports were then sent to either higher management or another department. In Case A the organisation worked on a bell curve reward system and one project manager found this to be problematic saying: "*It's very demotivating, I've been in meetings where I know someone should be getting an A and they've been a B even though I've given them an A*". Case B had individual contracts in place with each consultant but had only nonmonetary rewards in place for team success. Research on shared reward systems shows that when long-term coordination is required, the optimal system is one where the team is rewarded based on relative performance. Individuals are motivated through peer sanctions and teams are incentivised through team rankings [21,25]. It is surprising that while a shared rewards system (*R6*) is appropriate for ASD [5], both our case sites used individual-based reward mechanisms, thus running the risk of promoting dysfunctional behaviour and destroying intrinsic motivation [7].

Planning: The Scrum methodology used in both sites ensured that planning is a continuous and inclusive process and all of our interviewees felt that it was such. Although the teams were not part of the long-term planning process in Case A, they felt they could have some influence on long-term plans if they felt they needed to change them. Once a project got the go ahead, high-level, long-term plans were put in place. These plans were presented to the teams, and the team members were provided with the opportunity to discuss the plans and contribute ideas and suggestions. In Case B, long-term planning was not considered by team members as they were mainly consultants hired for the duration of the project. A budget was put in place and they were given a contract based on this budget. They did not know where they would be once their current contract finished. This induced myopic thinking among team members as one Scrum master pointed out: "*we only have a short horizon here; we only have a budget for the rest of the year which means we can't think any longer.*" To promote long-lasting ASD teams (*R7*), future research should examine the impact inclusive long-term planning has on an ASD team. This is particularly important when projects are staffed by consultants hired only for the duration of a particular project.

Control: In both sites, high-level project milestones were driven by the project budget. These were outlined at the project initiation phase, and using the Scrum methodology provided clear indicators as to where the project was against these high-level milestones. If these were in danger of not being met, then the project management had to make a decision about what actions to take depending on the constraints of the project. One project manager describes the process when budget costs are the main project constraint: "*So, if we have this triangle of cost, functionality and resources, we are stuck on resources because that's the main cost. We have been able to hold cost exactly at budget and quality we uphold by taking out of the box as much as possible.*" However, in some projects the budget was not the main constraint and *out of the box* was not an option. In these instances the delivery dates could be pushed back to accommodate unanticipated delays. The use of these flexible budgeting control mechanisms (*R8*) allowed the teams deliver the required functionality without compromising the quality of the system.

Resources: In both sites a fixed resource budget is assigned for each project, and while there are mechanisms in place to apply for additional funding, the ASD teams are not encouraged to do so, as one project manager points out: "*you would probably have bandwidth to do it once a year but if you have to go back looking for additional funds, you don't go back looking for a second time.*" One Scrum master shows how the fixed budget meant they had to break up a high-performing ASD team when the project budget was running out: "*They were good, they were doing so well and they were delivering excellent IT products…We really saw how good a team can be if they're allowed to stay in the same team for almost 2 years.*" Here again the use of flexible budgeting control mechanisms would allow teams to ramp up and ramp down their capacities depending on customer demand.

Coordination: In both of our case sites, the organisations provided both formal and informal communication mechanisms. Wikis, intranets, video conferencing facilities, etc. were designed to encourage coordination. The co-location of the team meant that intra-team coordination was carried out on a daily basis. This has already been shown to be beneficial to ASD teams by Mishra and Mishra [33] who found that an appropriate workspace environment had a positive impact on ASD team coordination. There was concern raised about inter-team communication and coordination (*R9*) with one Scrum master saying: "*I still think there is quite a lot that could be benefited for better coordination between teams at a team level*". According to coordination theory, actors in organisations face coordination problems that arise from dependencies that constrain how tasks can be performed [10]. In Case B a Scrum of Scrums was used as a coordinating mechanism on one project and this was found to be beneficial for coordinating and communication between teams. However, the project manager on this project stated that inter-team coordination on large projects was still one of the main areas of concern.

5 Conclusion

This study applies the Beyond Budgeting performance management model to an ASD environment. The findings suggest that contemporary thinking in budgeting resonates strongly with contemporary thinking in ISD. The Beyond Budgeting model shares many similarities with ASD with both having a distinctly agile and flexible feel. The findings suggest that ASD teams operate within environments that are affected by traditional organisational processes and structures. By using the Beyond Budgeting model as a lens to examine seven ASD teams, we explored how legacy processes have a direct impact on the daily operations of the ASD teams. We suggest ways in which ASD may be extended and improved upon to take into consideration these wider organisational influences.

Acknowledgments This work was supported, in part, by Science Foundation Ireland grant 03/CE/11855 (for CSET2) to Lero, the Irish Software Engineering Research Centre (http://www.lero.ie)

References

1. Abrahamsson P, Conboy K, Wang X (2009) Lots done, more to do: the current state of agile systems development research. Eur J Inform Syst 18(4):281–284
2. Agerfalk PJ, Fitzgerald B, Slaughter SA (2009) Flexible and distributed information systems development: state of the art and research challenges introduction. Inform Syst Res 20(3):317–328
3. Ahituv N, Hadass M, Neumann S (1984) A flexible approach to information system development. MIS Quarterly 8(2):69–78

4. Avison DE, Fitzgerald G (2003) Where now for development methodologies? Comm ACM 46(1):78–82
5. Boehm B, Turner R (2004) Balancing agility and discipline: a guide for the perplexed. Addison-Wesley, Boston
6. Bogsnes B (2009) Implementing beyond budgeting: unlocking the performance potential. Wiley, New Jersey
7. Cockburn A (2007) Agile software development: the cooperative game. Addison-Wesley, Boston
8. Conboy K (2009) Agility from first principles: reconstructing the concept of agility in information systems development. Inform Syst Res 20(3):329–354
9. Conboy K, Fitzgerald B (2004) Toward a conceptual framework of agile methods. In: Xp/Agile Universe 2004, proceedings, 3134, pp 105–116
10. Crowston K (1997) A coordination theory approach to organizational process design. Organ Sci 8:157–175
11. Dove R (2001) Response agility—the language, structure, and culture of the agile enterprise. Wiley, New York
12. Drucker P (1988) The coming of the new organization. Harv Bus Rev 66(1):45–53
13. Dugdale D, Lyne S (2006) Budgeting. Financ Manag (14719185):32–35
14. Dybå T, Dingsøyr T (2008) Empirical studies of agile software development: a systematic review. Inform Software Technol 50(9–10):833–859
15. Fitzgerald B, Hartnett G, ConboY K (2006) Customising agile methods to software practices at Intel Shannon. Eur J Inform Syst 15(2):200–213
16. Gladden GR (1982) Stop the life-cycle, I want to get off. Software Eng Notes 7(2):35–39
17. Gulati R, Oldroyd JB (2005) The quest for customer focus. Harv Bus Rev 83(4):92–101
18. Gurton A (1999) Bye bye budget. Account Int, March
19. Hansen SC, Otley DT, Van Der Stede WA (2003) Practice developments in budgeting: an overview and research perspective. J Manag Account Res 15:95–116
20. Hope J, Fraser R (2003) Beyond budgeting: how managers can break free from the annual performance trap. Harvard Business School Press, Boston, MA
21. Ishida J (2006) Team incentives under relative performance evaluation. J Econ Manag Strat 15(1):187–206
22. Jensen MC (2003) Paying people to lie: the truth about the budgeting process. Eur Financ Manag 9(3):379–406
23. Kautz K, Madsen S, Nørbjerg J (2007) Persistent problems and practices in information systems development. Inform Syst J 17(3):217–239
24. Keil M, Im GP, Mahring M (2007) Reporting bad news on software projects: the effects of culturally constituted views of face-saving. Inform Syst J 17(1):59–87
25. Knight D, Durham CC, Locke EA (2001) The relationship of team goals, incentives, and efficacy to strategic risk, tactical implementation, and performance. Acad Manage J 44:326–338
26. Larman C, Vodde B (2008) Scaling lean & agile development. Addison-Wesley, Upper Saddle River, NJ
27. Lee G, Xia W (2010) Toward agile: an integrated analysis of quantitative and qualitative field data on software development agility. MIS Quarterly 34:1
28. Lohan G, Conboy K, Lang M (2010) Beyond budgeting and agile software development: a conceptual framework for the performance management of agile software development teams. International Conference on Information Systems, St Louis
29. Maruping LM, Zhang XJ, Venkatesh V (2009) Role of collective ownership and coding standards in coordinating expertise in software project teams. Eur J Inform Syst 18(4):355–371
30. McFarland KR (2008) Should you build strategy like you build software? MIT Sloan Manag Rev 49(3):69–74
31. Miles M, Huberman A (1994) Qualitative data analysis: an expanded sourcebook, 2nd edn. Sage, London
32. Mintzberg H (2009) Rebuilding companies as communities. Harv Bus Rev 87(7/8):140–143

33. Mishra D, Mishra A (2009) Effective communication, collaboration, and coordination in eXtreme programming: human-centric perspective in a small organization. Hum Factors Ergon Manuf 19(5):438–456
34. Moe NB, Dingsoyr T, Dyba T (2010) A teamwork model for understanding an agile team: a case study of a Scrum project. Inform Software Technol 52(5):480–491
35. O'Brien R (1999) Living with budgeting. Manag Account 77:22
36. Pan G, Pan SL, Newman M, Flynn D (2006) Escalation and de-escalation of commitment: a commitment transformation analysis of an e-government project. Inform Syst J 16(1):3–21
37. Poppendieck M, Poppendieck T (2010) Leading lean software development: results are not the point. Addison-Wesley, Upper Saddle River, NJ
38. Rubinstein D (2007) Standish group report: there's less development chaos today. Software Dev Times 169:1–2
39. Sommerville I (2007) Software engineering, 8th edn. Pearson Education Limited, Essex, England
40. Wallander J (1999) Budgeting—an unnecessary evil. Scand J Manag 15:402–421
41. Walsham G (2006) Doing interpretive research. Eur J Inform Syst 15(3):320–330
42. Yin RK (2003) Case study research: design and methods. Sage, Thousand Oaks, CA

ELISA: Extensible Layer for Internet Services and Applications

Lubos Matl, Vladimir Kloucek, Viktor B. Bohdal, Jan Kubr, and Tomas Cerny

Abstract Contemporary P2P services and applications often design and implement the entire system from the base ground or builds on low-level frameworks. This leads to significant development and maintenance efforts and often reinvention of the wheel. In this chapter we propose extensible layer that does not expose low-level implementation details to client application developers, while providing them with well-known services and communication mechanisms built on efficient and scalable substrates. Analysis and design of architecture for this extensible layer are described. Special attention is given to security concerns and manycast communication approach. Our preliminary implementation is emulated and evaluated to establish functional and performance possibilities of the prototype.

1 Introduction

Existing applications communicating on Internet providewide variety of architectures and approaches. It is a common practice touse client–server architecture for systems that should be easy to locateon the Internet. Such approach provides simplicity when dealing withdevelopment, maintenance, security and localization. The architecture mayeventually become a performance bottleneck when the system demandsgrow. Alternative architecture to consider is peer-to-peer (P2P) [19]. It isoften used for collaboration of participants [18], for decentralization [1], self-scalability of applications [6] or anonymity of providers [9]. On the otherhand, this architecture

L. Matl (✉) • V. Kloucek • V.B. Bohdal • J. Kubr • T. Cerny
Computer Science & Engineering, Czech Technical University,
Charles Sq. 13, Prague, Czech Republic
e-mail: matllubo@fel.cvut.cz; kloucvla@fel.cvut.cz;
bohdavik@fel.cvut.cz; kubr@fel.cvut.cz; tomas.cerny@fel.cvut.cz

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_24,

does not naturally provide the ease of localization[18], security and maintenance of information [9]. In order to providemissing properties in P2P applications custom solutions are provided. Themain drawback of such approach is that developer often builds on lowlevel protocols or uses high level protocol and services ineffectively. Asnext, missing service is often reinvented, although, similar service alreadyexists somewhere else [12] and all maintenance efforts are placed on thedeveloper.

In this chapter we present our ongoing work on extensible layer for Internet services and applications (ELISA). It builds on generalization of our previous work on P2P approach for natural scaling of web caches [5, 6]. The aim is to simplify development of P2P applications, provide communication services such as {*uni*, *multi*, *any*}*casts* enriched with *manycasts*, in addition to natural support of *security*, minimal communication *latency* and high communication *bandwidth*. ELISA can be applied for applications that involve user collaboration, large data transfers [10], information sharing and video streaming [24], but also to improve performance of existing client–server systems with P2P self-scaling cache extension [5, 6] and similar. The goal of ELISA is to provide high-level API and framework for application development, ensuring security, providing all sort of communication methods and supporting composition of services and different types of networks. The main focus of this chapter is to present ELISA's extension mechanisms, to provide discussion and evaluation of our approach towards manycast service and security. As next, our preliminary emulation results on manycast and security are provided.

This chapter is organized as follows. In the following Sect. 2 is discussed related work. Section 3 provides analysis and design of the architecture and describes our manycast and security approaches. Section 4 provides evaluation of emulation results. Last section concludes the chapter and provide future work.

2 Related Work

The area of P2P frameworks and generic API has significant amount ofrelated work [1, 3, 11, 18, 22, 24]. Multiple existing frameworks providelow-level communication that requires deep knowledge of the underlyingimplementation, in order to write a simple client application. It is a commonpractice that security concerns are not naturally provided by frameworks and leftto client applications [2]. Multiple architectural proposals are made relative to aparticular framework [18, 22, 24], but a simple extension to its functionalityleads to framework reimplementation rather than to a simple plug-inextension [8]. Furthermore, some of existing frameworks put its effortson a single goal such as media streaming [24] or effective multicast [4], and thus provide only a subset of operations that are used in commonapplications.

In the following text we introduce multiple existing frameworks and describe their qualities, benefits or parts that are left to develop in client application. All these frameworks provide services on the top of an overlay network. An overlay network is a computer network built on top of another network. Its nodes can be thought of as being connected by virtual or logical links, each of which corresponds to a path, perhaps through many physical links, in the underlying network. Overlay networks [14] create a structured virtual topology above the basic transport protocol level that facilitates deterministic search and guarantees convergence.

First introduced framework Pastry [18] is used for building scalable, self-organizing, structured P2P overlay networks. Its nodes are assigned random IDs, objects are assigned random keys from a large ID space. Both are 128 bits length (2^b). Given a message and a key, Pastry routes the message to the node with the ID that is numerically closest to the key (key's root). Each forwarding node searches in its routing table for the node that has its ID numerically closest to the message key [18]. If such node is found then the message is forwarded, otherwise the current forwarding node is the numerically closest to the key. Expected number of routing hops is $\log 2^b(N)$. It is low level framework and developer must be aware of implementation details. No services such as anycast, multicast or security are provided. Pastry fits well as a core implementation for high-level frameworks. Multiple extensions to Pastry exist. For example Scribe [1, 3] provides scalable application level multicast. A pseudo-random Pastry key (groupID) is chosen for each multicast group. Network then uses multicast trees. Messages are forwarded from the root to the members using reverse path forwarding. Group membership is decentralized. It can support large numbers of groups, arbitrary number of group members with dynamic membership. Pastry is furthermore extended with anycast for distributed resource delivery. In order to support high bandwidths SplitStream [4] extends both Scribe and Pastry. It splits the data stream into a multiple stripes and multicasts each stripe in a separate multicast tree.

Alternative approach to SplitStream is brought by Chunkyspread [22]. It avoids nodes getting loaded beyond their capacities through its push-down and anycast operations. A child peer that requests join on a fully loaded parent initiates a push-down on that parent; it selects a child that is pushed-down in parent's descendants or alternatively sends an anycasts to the parent's group to join. Although it works well for homogeneous networks, it does not fit to heterogeneous environments as both push-down and anycast repeats often and leads to frequent peer disconnections. On the other hand, it provides good control over transmit load, better latency and responsiveness. The traditional approaches of tree-building and swarming have significant overhead in terms of data and control (to rebuild tree) and exhibit delays. Chunkyspread provides control over each members transmit load, reacts on changes, allows low latencies and has low overhead. The aim is to provide high bandwidth with low latencies, but available services are limited, and there is no support for service aggregations and security. It does not provide any simple extension mechanism. Addition of a new service would require knowledge of the entire system and recompilation.

Framework SAAR [17] introduces alternative architecture for overlay multicast and separates control and data overlay. Unstructured overlay is used for content delivery and structured for control as it allows effective neighbors discovery. Authors suggest that structured overlays are not good for stream-based overlay networks as they produce constrained and sub-optimal data distribution paths. SAAR builds on the top of Scribe and thus provides both multicasts and anycast. Similar to previous work the framework provides low-level features and does not provide extension mechanisms.

CoolStreaming/DONet [24] is a data-driven overlay network for live media streaming. The architecture extensively uses multicast for data on-demand. The system does not maintain any structure, but instead peers exchange information about the data they currently have. This framework focus on large data sharing and does not provide efficient mechanisms for small file distribution. As next, the framework is low-level exposing implementation details.

In [11] authors propose a general API for P2P overlays. They study similarities in existing P2P frameworks such as Pastry [18], Chord [21], CAN or Tapestry and suggest a three-tier abstraction. The first tier deals with the key-based routing. Second tier provides abstraction for distributed hash table (DHT), group anycast and multicast (CAST) and decentralized object location routing (DOLR). Third tier provides the high-level services and abstractions. Although the third tier provides abstraction, some applications might use the first tier directly. Most of the peers in the overlay use an uniform "peerID" from large identifier space (such as in Pastry). Application specific objects are assigned "keys". A key is mapped by the overlay to a unique live peer called key's root. To deliver a message to the root, each peer maintains a routing table (peerID and an IP address). Messages are forwarded through the overlay links to the peers whose peerIDs that are closer to the key. Similarities among existing overlays allow authors to build a common API that will fit to most of the applications. Such network can be extended by anycast implementation. Proposed API, although, does not naturally support extension mechanisms for services and applications. New functionality or service requires modification of given layer and recompilation. Suggested API is still low level. Security support is missing.

These above-mentioned frameworks provide generic and scalable substrate for P2P applications, and they provide communication services such as {*uni*, *any*, *multi*}*casts*. They can be seen as low-level frameworks and fit as a code to other systems. They expose low-level details and does not provide extension mechanisms that would allow to add new services without deep knowledge of its implementation. Below we introduce frameworks that address these disadvantages.

NEBLO (NEarly BLind Overlay) [7] is a structured overlay network which use routing tables to transfer messages with logarithmic number of hops over forwarders (such as Pastry). The aim is privacy or anonymity of participant entities and scalability. NEBLO implements Pretty Flexible API for Generic Peer-to-Peer Programming [8] that introduces a very general principles for the development of P2P applications. It provides an extension mechanism where developer registers event handlers; these provide the user necessary information needed for his client application unless knowing the implementation details of the framework.

A Framework for Constructing Peer-to-Peer Overlay Networks in Java [20] has a goal to create universal interface for applications that use P2P networks. Through the independence on concrete P2P network an application can communicate through different networks. Furthermore, it provides solutions and mechanisms beyond P2P.

Two networks are implemented in this framework; Chord [21] and FISSIONE [16]. This framework focuses on application communication through multiple networks.

From the previous text can be seen that all mentioned frameworks fail to provide support or mechanisms for security. However, certain application domains might not put high efforts on security or may provide centralized verification mechanisms [10]. Security should be core part of P2P frameworks. The reason behind this is that security for P2P systems is challenging. Proposal for P2P security mechanism is brought by Ganesh and Zhao [15]. In this work identity theft protection in structured overlays uses hash function and other nodes to verify node identity. It attempts to remove two problems of security, a *sybil attack* [13] and *eclipse attack* [2], in a very general way. Every node in the network sends periodically signed existence proofs for its prefix to randomly chosen proof managers. Signed existence proofs are created from prefixes of identifier, which was signed with node private key. These proofs are send to P2P nodes whose identifier is generated as hash of identifier prefix (from which was created the proof) and some salt. Every node can use this certification authority to verify node identifier.

3 Analysis and Design of ELISA

This section provides description on architectural design of ELISA, its extension mechanisms, security and communication services. Core components and their functionality are described with adaptation to existing P2P substrates [1, 3–6, 18]. ELISA adapts to low-level services of existing frameworks and integrates them. Its design takes into consideration simplicity of extension and provides consistent and efficient high-level services including security, service composition, etc. The consideration of extension is provided through plug-in mechanism that allows us to hide implementations details and expose only necessary details to the plug-in developer. The architecture provided in Fig. 1 reflects the goal to provide extension mechanisms. It contains three main parts: *Network API*, *Plug-in* and *Plug-in manager*.

Concrete Plug-in represents a client application. Each $plug-in$ can inherit *data message* types, which it will work with or to specify its expected format of data. It uses *Network API* for communication.

Plug-in manager component serves as a controller for plug-ins to call *Network API* or to asynchronously receive messages. It is also possible to communicate among multiple plug-ins. The interaction of components is shown in Fig. 2.

Network API is an interface used for access to the network. Its adapter called *Peer-to-Peer network* is used for communication with frameworks Pastry [18] and Scribe [3]. Developer has the possibility to customize the API to adapt to different architectures. *Network API* uses *Plug-in manager* component with access to all $Plug-ins$ to pass them events populated throughout communication. Each $plug-in$ can register its interest in certain events. *Peer-to-Peer network* provides communication services

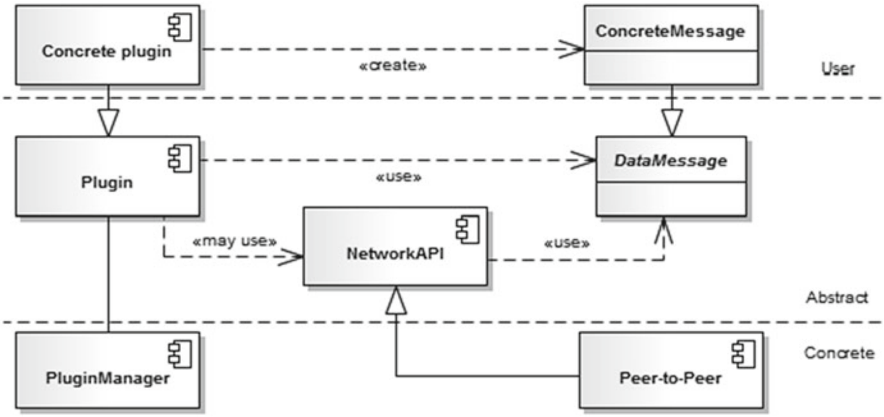


Fig. 1 Architecture schema of ELISA

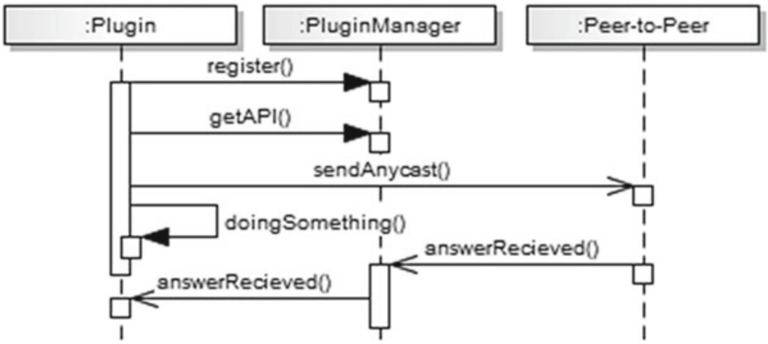


Fig. 2 Interaction of ELISA core components

such as unicast, anycast, multicast and also manycast, which we introduce later. It starts and terminates data streams among peers.

Client application in ELISA is represented as a plug-in. Such plug-in can use its data types that extends generic *Data Message*. Plug-in registers at *Plug-in manager* its interest in certain events. When the *Network API* receives a message, then all plug-ins with the registered interest are notified in specified order. Plug-in can choose specific data message identification and its type hash. Each message with given type hash can be intercepted by plug-ins registered for the particular type. The type hash is only known to developer of a given plug-in and thus other plug-ins are hard to intercept such message unless they know the message hash type.

3.1 Peer-to-Peer Network

Peer-to-peer network provides adapter to Pastry [18] and Scribe [1, 3] groups for peer cooperation. This component creates group of peers that are represented by text strings. It is possible to send three message types: anycast [3], multicast [1] and manycast. Pastry defines rules of routing messages on P2P network and Scribe provides the creation of groups, joining to group and sending anycast and multicast. Anycast is a message, which receives one of the nearest group members of Scribe group and multicast is received by all group members. Every peer that is connected to Pastry network has its own unique ID used for routing messages. Extension of *Peer-to-peer network* to the underlying frameworks is the *manycast* communication. It is such communication that can be seen as multiple anycasts to n receivers connected to a Scribe group. There are two distinct types of manycast communication: *N-anycasts* and *Tree distribution*. Both types can further consider cases when the receivers must be distinct.

N-anycasts uses n anycasts that are sent from one sender. When peer sends two anycasts at once these anycasts are often received by the identical group of members. All routing decisions are made under the same conditions to forward the message. Therefore the sender sends anycasts via its n nearest neighbors regardless of the routing table. This increases the amount of different receivers. Scribe anycast implementation ensures that messages arrive to near peer (if such peer exists) and *N-anycasts* preserves this property. The upperbound manycast group is restricted to 30 receivers.

Tree distribution uses existing Scribe tree for message distribution. Manycast sender sends only one anycast. This anycast contains information about number of receivers (n). Each anycast receiver decrements the number of receivers by one and creates m messages. This contains information about $(n-1)/m$ receivers and forwards this messages to m peers in the group tree using anycast. Each forwarder continues with this approach down to the Scribe tree, until $n>0$. The value of m can be set according to the forwarding peer load or can be set as a constant. Greater value of m means greater load of a forwarding peer, but reduces the delivery time.

Distinct receivers are not guaranteed naturally by neither of the above approaches. When a manycast sender wants to send a message to n distinct receivers, then an extension mechanism applies. Each receiver accepts the message once and all identical messages are forwarded. This reliable approach increases the number of hops for the manycast message. If not enough receivers exist, then a failure message is delivered to the sender.

3.1.1 Secure Connections

Every peer in the network needs an identifier. If we allow peers to generate their own identifiers, then significant security problem raises. This mechanism could be used by malicious users for *eclipse attack*. Therefore it is necessary to disallow

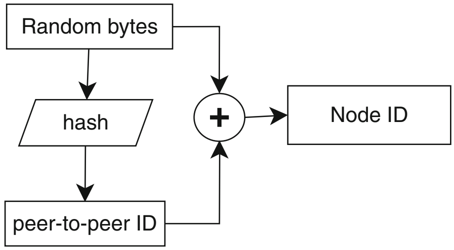


Fig. 3 Generation and validation of an secure identifier

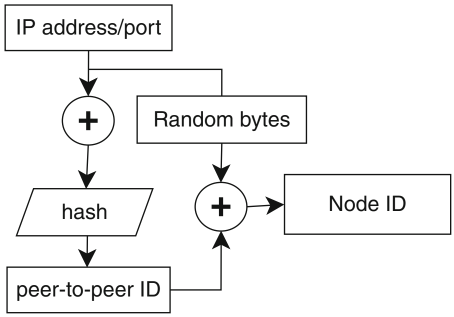


Fig. 4 Generation and validation of an secure IP identifier

peers to their generate identifiers. Instead these can be generated through authority server, but this might be a bottleneck regards scalability. Another solution [15] is to use a hash function. Peer produces random byte stream from which it creates a hash that forms an identifier. Both the hash (identifier) and the byte stream are being sent to other peers upon initial connection, so that other peers can verify correctness of the hash from the byte stream. With the hash function this approach ensures that no one can generate identifier that is near to another peer and thus it prevents *eclipse attack*. This mechanism of ID generation can be seen in Fig. 3. This solution, although, does not solve *identity theft*. To deal with this issue, peers can create an identifier as a hash of its IP address and port together with a random value, then they only send the random value with the identifier. Every peer in the network can check whether the identifier was created from the random value and the peers IP address and port. This method is described by the right side of the Fig. 4.

4 Case Study

In this section we provide comparison of implemented manycasts methods. Each manycast type might fit better to a different situation. First, we evaluate manycast efficiency in terms of number of hops over message way from sender to receiver and later we compare manycasts according to the security. The evaluations are made through emulation on Emulab [23] to test various scenarios in real-like conditions. Emulab provides large amount of physical computers, but also a limitation on the

Table 1 Average number of distinct *receivers* for ambiguous manycast

Number of receivers	5	10	15	20	25	30
Tree distribution	4.8	9.2	9.8	12.8	14.6	15.2
N-anycasts	4.4	7	10.2	13.2	13.8	14.8

Table 2 Comparison of max. number of *hops* of manycast with ambiguous and distinct receivers

Number of receivers			5	10	15	20	25	30
Ambiguous receivers	Tree distribution	(hops)	3	4	4	5	5	5
	N-anycasts	(hops)	1	1	1	1	1	1
Distinct receivers	Tree distribution	(hops)	6.2	12.8	7.8	12.8	18	31.6
	N-anycasts	(hops)	6.2	11.4	3.2	6.6	13.6	22.8

amount exists due to its extensive use for experiments. For our emulation we consider 151 computers with Linux systems. In the measurement 30 peers are connected to the group, and these peers represent manycast receivers. Additional 120 peers are forwarders and one peer is manycast sender. We evaluate both approaches with consideration of distinct and non-distinct receivers. In our scenario we consider submission of manycast with 5, 10, 15, 20, 25 and 30 receivers twice.

4.1 Manycast with Ambiguous Receivers

The average number of distinct receivers, while not considering the distinct receiver assurance, is shown in Table 1. It can be seen that both approaches have similar results. Another comparison can be seen from the number of hops to deliver a message. Table 2 shows the maximum number of hops for both approaches. The *N-anycasts* approach results in 1 hop for all cases. The *Tree distribution* provides worse results for the number of hops that grows with the number of receivers.

4.2 Manycast with Distinct Receivers

For this method the number of distinct receivers is equal to n. The comparison of both approaches regards the maximal number of hops for a manycast message is shown in Table 2. The *N-anycast approach* is more efficient. Next, we compare both approaches according to the total number of hops. With total is meant the sum of hops for all messages and different receivers sent by one manycast. The results for both methods are shown in Table 3. More details can be seen in Fig. 5, which shows that after 25 manycasts the *Tree distribution* is more efficient.

Table 3 Average number of *hops* for evaluation of manycast with distinct receivers

Number of receivers		5	10	15	20	25	30
Tree distribution	(hops)	4.2	6.2	24.4	31	67	136.2
N-anycasts	(hops)	2	2.6	24.6	48.6	82.6	146.2

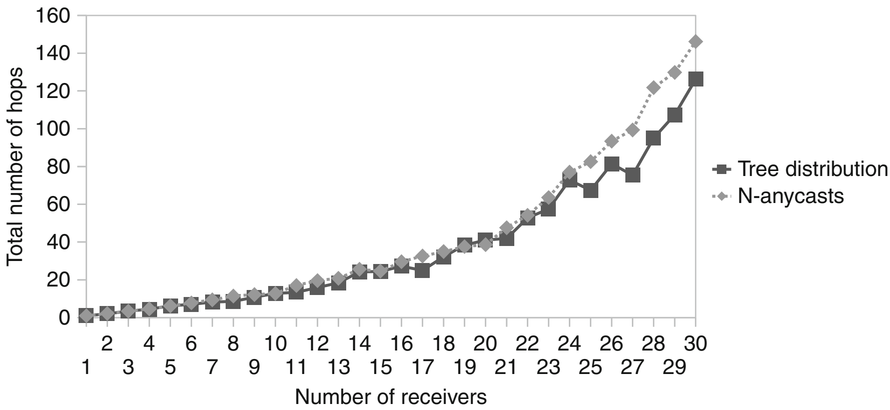


Fig. 5 *N*-anycasts—tree distribution total number of hops comparison

4.3 Security Comparison

We evaluate the network with the same settings as in previous experiment; only the number of receivers is 15. As next, we vary the percentage of malicious peers. Malicious peers change the manycasts message and we measure the number of manycast messages that are changed. We evaluated both situations where attackers are forwarders and group members. The number of attackers is set in the range of 1–50 % nodes from forwarders or receivers. You can see the results in Figs. 6 and 7. In both the situations *N-anycasts methods* (with and without distinct receivers) give better results because the number of changed messages is lower. The *N-anycasts method* is better because of using more paths from manycast sender and Scribe tree. It is not easy to capture all these messages and change them. In the case of *Tree distribution*, when an attacker receives a message on its path from sender to Scribe tree, he can change all manycasts messages.

4.4 Performance Comparison: Time to Delivery

We use the maximum number of hops from sender to receiver as the time to deliver the manycast messages to all receivers (Table 2). In the *N-anycasts method* the delivery time is the number of hops from sender to the first receiver for every message.

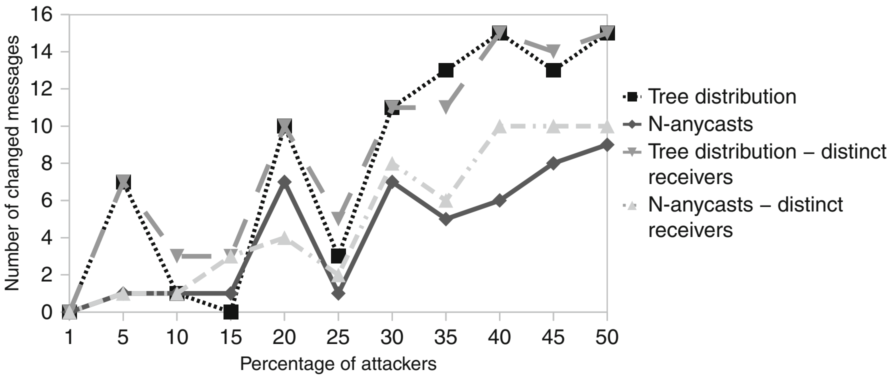


Fig. 6 Number of messages changed during transfer by attackers are among forwarders

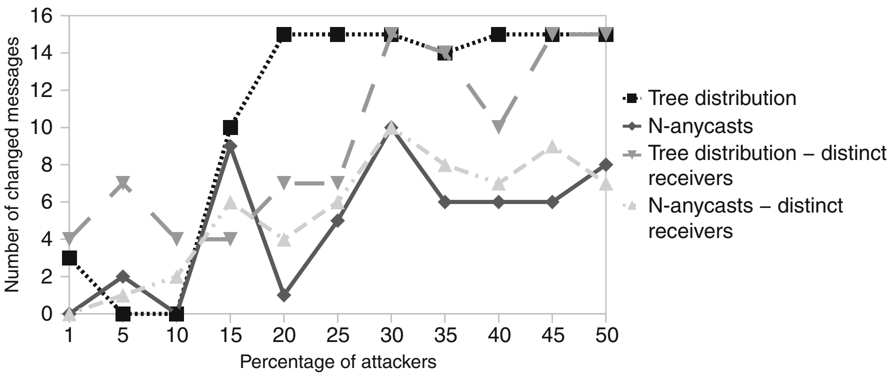


Fig. 7 Number of messages changed during transfer by attackers are among group members

The difference is that for *Tree distribution* messages are forwarded in the Scribe tree, which adds extra hops to the maximum number. *N-anycasts method* has lower number of hops for both with and without requirement of distinct receivers. Therefore time difference between the first and the last received manycast is lower for *N-anycasts method* and thus the manycast delivery is faster. To achieve fast response the *N-anycasts method* provides better results.

5 Conclusion

In this chapter we presented ELISA framework for P2P application development. It provides extension mechanisms for custom services without exposing implementation details. Client application builds on the top of plug-ins that register their interest in given communication events. ELISA provides multiple options for communication with addition of manycast approach that can address given amount of peers in the group to support collaborative task. As next, our framework provides security mechanisms to prevent eclipse attack and identity theft. Its architecture builds on the top of existing substrates Pastry and Scribe and delegates them to the lowest layers that allows simple replacement for another types of networks.

In future work we plan to extensively evaluate message security that is important for production use. For example a message integrity must be preserved while its travel through the network, intentional flood on the network should be prevented or malicious group canceling avoided. As next, we consider replacement of emulation environment due to its size limitations (amount of computers) into PlanetLab network. Regards implementation the nearest goals are to develop plug-ins demonstrating the benefits of ELISA such as cooperative web caching suggested by Cerny et al. [5, 6], media content streaming with its performance and development efforts evaluation.

References

1. Castro M, Druschel P, Kermarrec A, Rowstron A (2002a) SCRIBE: a large-scale and decentralized application-level multicast infrastructure. IEEE J Sel Areas Commun 20(8):1489–1499
2. Castro M, Druschel P, Ganesh A, Rowstron A, Wallach DS (2002b) Secure routing for structured peer-to-peer overlay networks. SIGOPS Oper Syst Rev 36(SI):299–314. Available: http://doi.acm.org/10.1145/844128.844156 [online]
3. Castro M, Druschel P, Kermarrec A-M, Rowstron A (2003a) Scalable application-level anycast for highly dynamic groups. Springer, Berlin, pp 47–57
4. Castro M, Druschel P, Kermarrec A-M, Nandi A, Rowstron A, Singh A (2003b) Splitstream: high-bandwidth multicast in cooperative environments. SIGOPS Oper Syst Rev 37(5):298–313. Available: http://doi.acm.org/10.1145/1165389.945474 [online]
5. Cerny T, Jaromerska S, Praus P, Matl L, Donahoo J (2011) Cooperative web cache. In: 18th international conference on systems, signals and image processing
6. Cerny T, Praus P, Jaromeska S, Matl L, Donahoo M (2012) Towards a smart, self-scaling cooperative web cache. In: Bielikova M, Friedrich G, Gottlob G, Katzenbeisser S, Turen G (eds) SOFSEM 2012: theory and practice of computer science. Lecture notes in computer science, vol 7147. Springer, Berlin, pp 443–455. 10.1007/978-3-642-27660-6_36. Available: http://dx.doi.org/10.1007/978-3-642-27660-6_36 [online]
7. Ciaccio G (2005) NEBLO: Anonymity in a Structured Overlay. DISI, Università di Genova, Technical Report DISI-TR-05-05, May 2005
8. Ciaccio G (2007) A pretty flexible api for generic peer-to-peer programming. In: IEEE international parallel and distributed processing symposium, 2007 (IPDPS 2007), pp 1–8
9. Clarke I, Sandberg O, Wiley B, Hong T (2001) Freenet: a distributed anonymous information storage and retrieval system. In: Designing privacy enhancing technologies. Springer, Berlin, pp 46–66
10. Cohen B (2003, June). Incentives build robustness in BitTorrent. In Workshop on Economics of Peer-to-Peer systems, vol. 6, pp. 68–72
11. Dabek F, Zhao B, Druschel P, Kubiatowicz J, Stoica I (2003a) Towards a common api for structured peer-to-peer overlays. In: Internationals workshop on peer-to-peer systems
12. Dabek F, Zhao B, Druschel P, Kubiatowicz J, Stoica I (2003b) Towards a common api for structured peer-to-peer overlays. In: Kaashoek M, Stoica I (eds) Peer-to-Peer systems II.

Lecture notes in computer science, vol 2735. Springer, Berlin, pp 33–44. doi:10.1007/978-3-540-45172-3_3 Available: http://dx.doi.org/10.1007/978-3-540-45172-3_3 [online]
13. Douceur JR (2002) The sybil attack. In: Druschel P, Kaashoek MF, Rowstron AIT (eds) IPTPS. Lecture notes in computer science, vol 2429. Springer, Berlin, pp 251–260. Available: http://dblp.uni-trier.de/db/conf/iptps/iptps2002.html#Douceur02 [online]
14. Doval D, O'Mahony D (2003) Overlay networks: a scalable alternative for p2p. IEEE Internet Comput 7:79–82
15. Ganesh L, Zhao BY (2005, November). Identity theft protection in structured overlays. In Secure Network Protocols, 2005 (NPSec). 1st IEEE ICNP Workshop on (pp. 49–54). IEEE
16. Li D, Lu X, Wu J (2005) Fissione: a scalable constant degree and low congestion dht scheme based on kautz graphs. In: Proceedings of the IEEE 24th annual joint conference of the IEEE computer and communications societies (INFOCOM'05). , vol 3, pp 1677–1688
17. Nandi A, Ganjam A, Druschel P, Ng TSE, Stoica I, Zhang H, Bhattacharjee B (2007) Saar: a shared control plane for overlay multicast. In: Proceedings of the 4th USENIX conference on networked systems design & implementation, NSDI'07. USENIX Association, Berkeley, pp 5–5. Available: http://dl.acm.org/citation.cfm?id=1973430.1973435 [online]
18. Rowstron AIT, Druschel P (2001) Pastry: scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In: Proceedings of the IFIP/ACM international conference on distributed systems platforms Heidelberg, Middleware '01. Springer, London, pp 329–350
19. Schollmeier R (2001) [16] a definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In: Proceedings of the first international conference on peer-to-peer computing, P2P '01. IEEE Computer Society, Washington, DC, p 101–102. Available: http://dl.acm.org/citation.cfm?id=882470.883282 [online]
20. Shen R, Wang J, Zhang S, Shen S, Fan P (2009) A framework for constructing peer-to-peer overlay networks in java. In: Proceedings of the 7th international conference on principles and practice of programming in Java, PPPJ '09. ACM, New York, pp 40–48. Available: http://doi.acm.org/10.1145/1596655.1596662 [online]
21. Stoica I, Morris R, Liben-Nowell D, Karger D, Kaashoek M, Dabek F, Balakrishnan H (2003) Chord: a scalable peer-to-peer lookup protocol for internet applications. IEEE/ACM Trans Netw 11(1):17–32
22. Venkataraman V, Yoshida K, Francis P (2006) Chunkyspread: heterogeneous unstructured end system multicast. In: Proceedings of the ICNP
23. Webb K, Hibler M, Ricci R, Clements A, Lepreau J (2004) Implementing the Emulab-PlanetLab portal: experience and lessons learned. Available: http://www.usenix.org/publications/library/proceedings/worlds04/tech/webb.html [online]
24. Zhang X, Liu J, Li B, Yum TSP (2005) Coolstreaming/donet: a data-driven overlay network for peer-to-peer live media streaming. In: IEEE infocom

Knowledge Networks and Innovation-Driven Collaborations: A Study in Victoria's Biotechnology Industry

Adel Moslehi, Henry Linger, and Kerry Tanner

Abstract In a rapidly changing era, organisations need to be innovative in order to survive, and they need to access external knowledge. This paper examines knowledge networks between organisations by focusing on three main elements of a network: content, knowledge process and the structure of the network. Although the influence of these three elements on innovation has been discussed, the interrelation between these elements has not been explored in detail. The subject of our study is the patent co-authorship networks of biotechnology firms in Victoria, Australia. The data were extracted via the Australia patent database (*AusPat*) from 2001 to 2010. Using social network analysis and statistical analyses, our study explores the relationship between knowledge processes, knowledge and network structure. This exploratory and ongoing research aims to extend our understanding of knowledge networking and their role in inter-organisational collaboration.

1 Introduction

The need for innovation-driven collaboration projects in the high-tech industry has been widely recognised [20]. One of the most widely cited motives for collaboration is the acquisition of knowledge and capabilities from partner firms (e.g. [25]). Any innovation, such as a new product development, needs to integrate existing and new knowledge. Although knowledge may be created internally, for example, by conducting research and development or via individual learning, knowledge is continuously changing and depreciating. Thus every organisation needs access to external knowledge [21], and the way that firms acquire such knowledge plays a very

A. Moslehi (✉) • K. Tanner
Monash University, Caulfield East, VIC, Australia
e-mail: adel.moslehi@monash.edu

H. Linger
Faculty of Information Technology, Monash University, Caulfield East, VIC, Australia

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_25,

important role in the ability of organisations to innovate [12]. In this regard the importance of knowledge acquisition and transfer to innovation-driven collaboration has been considered in the literature. By addressing global manufacturers in the high-tech industries, a recent survey by IDC Manufacturing Insights shows that 60 % of companies claim that they fail to support innovation and access to real-time information and improved collaboration has been considered as one of key criteria for successful innovation.

However, research in this area has mainly focused on structural view [28] to collaboration networks, and few studies have focused on how knowledge is managed between organisations. In this paper to get a richer picture, we also consider the properties of knowledge that is being shared in the network and the process that firms adopted to access knowledge via inter-organisational knowledge networks (k-networks). Knowledge content, knowledge processes and the structure of the network are three defining characteristics of k-networks that are discussed in this paper. Although each element has been studied separately, there is limited literature on the interaction and the integration of these three elements. In the rest of the paper, first the literature is reviewed and highlights the gap in knowledge; then research approach, findings and analysis are presented subsequently.

2 K-Networks and Innovation-Driven Collaboration

By reflecting Schumpeter's [38] claim that innovation is the new combinations of knowledge, Kraaijenbrink et al. [21] argued that any organisation needs external knowledge to be innovative since occurrence of any innovation (e.g. a new product) needs integration of existing and new knowledge. In this regard, firms interact with other companies through collaboration networks or strategic alliances [11] to acquire or access external knowledge (e.g. [14]). Here I argue that while the main focus of the knowledge management (KM) literature is on intra-organisational aspects (e.g. [32]), the focus of innovation-driven collaborations is the acquisition or access to external knowledge via k-networks. Organisational k-networks consist of nodes (i.e. firms), links (which make knowledge sharing possible) and the content (the knowledge shared through the links) (e.g. [34]). Thus k-networks can be explored by addressing three elements of interfirm k-networks [24]: content, process and structure of the knowledge network. These elements are explored in the following sections.

2.1 Content of the K-Network

In a k-network, content refers to attributes of knowledge that is being shared within the networks by nodes and through links [30]. The KM literature has argued that for an innovation project, organisations need to acquire new knowledge (e.g. [16]). This new knowledge, according to the knowledge-based view (KBV) of the firm, should possess some characteristics like tacitness and heterogeneity in order to be a source of innovation and sustainable competitive advantages over rivals (e.g. [3]). At both an individual and organisational levels, studies on innovation have shown that higher heterogeneous knowledge fosters creativity (e.g. [36]). However, very few studies have examined content from an inter-organisational perspective. Rodan and Galunic [28], for example, have studied the role of individual managers' social networks in terms of access to heterogeneous knowledge for innovation performance. Sammarra and Biggiero [30], by studying interfirm collaboration networks, showed that firms acquire heterogeneous knowledge areas for their innovation projects through their networks. Degree of tacitness is the other attribute of the knowledge content which is very important to acquire, though it is difficult to transfer (e.g. [33]).

2.2 Process of the K-Network

Within an inter-organisational k-network, nodes represent participating organisations that collaborate with other network members in order to innovate new products/services. These nodes, in the context of inter-organisational collaboration, are involved in KM processes, particularly in the knowledge-intensive industries (e.g. [35]). Drucker mentioned that a firm needs to have the capacity for generating knowledge and applying it in the form of innovation [8], and the KM process is typically presented as a catalyst for innovation [31]. These processes may aim at either enhancing exploitation (i.e. focusing on existing knowledge which is acquired, transferred and used in other similar situations) or exploration (i.e. creating new knowledge by sharing and synthesising of knowledge [23]). Although exploitation is important for innovation, it is largely exploration through knowledge sharing that allows the development of genuinely new approaches [19]. Moreover, as March and Levinthal [23] mentioned, companies need both exploration and exploitation in order to survive for the long term, but there is often a trade-off between the two approaches due to constraints on resources and the firm's strategic orientation [4].

2.3 Structure of the K-Network

At a dyad level of a k-network, links represent the collaboration ties that two organisations share with each other. Through these links, organisations share knowledge that they require. At a network level, these links comprise the whole structure of the network and influence the firm's performance [29]. Firms that are central to the network, for instance, may contribute to beneficial outcomes like power [39], decision-making [40] and individual innovation (Ibarra [41] as cited by Salman [29]).

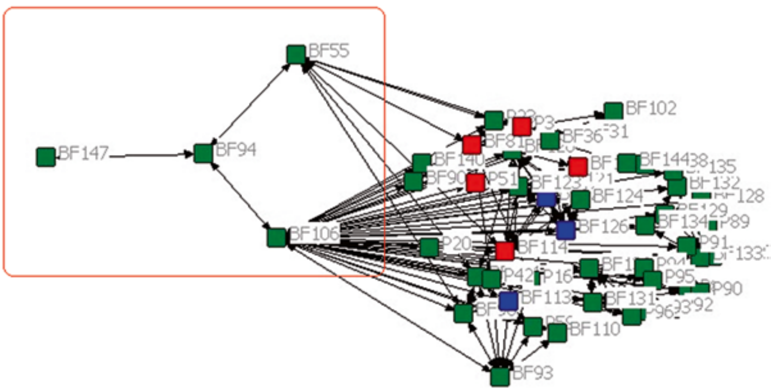


Fig. 1 Ego network of BF94: *red boxes* show partner organisations which are located in Australia but not within Victoria, while *blue* ones represent foreign partners

Dense networks provide close relations and build trust within the network. In this regard, members of dense networks are supposed to have better access to valuable strategic knowledge [9]. However, they are possibly in the structural biases [1] that makes it difficult for them to access new knowledge from a node located outside their close relations.

2.4 The Need for Integration: The Gap in Current Knowledge

Although these three elements have their own influences on innovation, the interrelation between them has not been explored in the literature (Fig. 1) except in a few research studies, for example, by Hansen [15] researching intergroup networks and Bustamante [6] using simulation techniques to research the interactions between content and structure. Exploration of the interrelations between these three elements is the means to focus on how k-networks are used in innovation-driven collaboration. In this regard, first there is a need for understanding the characteristics of these k-networks, in terms of the three elements. Second these characteristics are analysed in terms of how they contribute to the firm's innovativeness. Thus the two research questions addressed in this paper are:

- What are the characteristics of k-networks used by firms in innovation-driven collaboration in terms of structure, content and process?
- What are the interactions among content, process and structure of the knowledge network?

3 Research Approach

3.1 The Victorian Biotech Industry as the Research Domain

To study the research questions, this exploratory research has focused on the biotechnology industry as a suitable knowledge-intensive high-tech industry where innovation is highly reliant on k-networking [27]. At its simplest, biotechnology is technology based on biology [5] and includes all techniques for manipulating microorganisms [35]. According to Ernst and Young [13], Australia is "the largest presence in the Asia-Pacific region", and Victoria as one of the leading states within Australia, characterised by a large human health subsector in biotechnology industry. More than 75 % of biotechnology companies are located in the state, and in aggregate, these companies were valued at almost $24 billion [2].

3.2 Network Definition

To define the k-networks, we adopt the definition used by several authors such as Kane and Alavi [18]. A k-network refers to a collection of individuals or organisations with whom firms interact to acquire or access needed information to accomplish tasks in their innovation projects. In this research we focused on patent co-authorship networks as one of the formal resource sharing networks (e.g. [26]). These networks are characterised by legally binding contracts and licencing to protect intellectual property for exclusive partners.

3.3 Data Collection

To define any inter-organisational network, there is need to define nodes and links of the network. Here links are defined by co-authorship patenting, i.e. if two companies have registered a patent with each other, it means that they had innovation-driven collaboration, though with a 1- or 2-year delay [42, 7]. Patent documents provide data on the assignees and their locations at the time of the inventions, inventors and the international patent classification (IPC), which will be explained later. Nodes are also represented by biotechnology firms headquartered in Victoria. There is no pre-existing list, so we created our own initial list of firms based on three main sources: membership lists of AusBiotech (http://www.ausbiotech.org), the BioMelbourne portal (http://www.biomelbourne.org) and Australia stock exchange (ASX). Then by searching their patents in AusPat,[1] a database which records all Australian patents published since 1904, we found the partners (co-authors) of the

[1] http://www.ipaustralia.gov.au/auspat/index.htm

initial list. Then based on a snowball sampling method, we found other firms that are not listed in these three sources but published at least one patent during 2001–2010. Through this iteration of snowball sampling we found 126 firms located in Victoria. Also 61 partners were identified, of which 27 are located in other countries and 34 are headquartered in other states of Australia. All these companies, based on the publicly available data on these BFs, were categorised as public research organisations (PRO) like universities, pharmaceutical-bio firms (PBF), government agencies (GOV), hospitals (H) or individual innovators (II).

4 Findings

4.1 Content: Firms' Knowledge Heterogeneity

Heterogeneity of the knowledge is an essential element of knowledge content (e.g. [28]). In general, firms might need different areas of managerial, technological and organisational knowledge [30]; however, the only focus of this research is on technological knowledge. To calculate the degree of heterogeneity of a firm's knowledge, IPC[2] has been considered as a representative indicator. IPC shows the technological knowledge area which is relevant to the patents that firms have published. To calculate the degree of heterogeneity, we used the Herfindahl index and Simpson's diversity index, which have been used in a variety of fields including KM research [10]:

$$D = 1 - \sum_{i=1}^{N} p_i^2$$

where p is the proportion of particular knowledge areas (IPCs) in the ego network of firm i and N total number of all IPCs which are shared by firm i.

In this regard, knowledge heterogeneity was calculated for all 126 firms. The lowest degree of heterogeneity is 0 for firms with only one IPC. And the highest is 0.929 for the University of Melbourne.

4.2 Process: Firms' Exploration/Exploitation Process

To understand the process focus of the firms, this research has used the exploration index that shows the degree of exploration vs. exploitation [23]. Lavie and Rosenkopf [43] identified three domains of exploration and exploitation: function, attribute and structure. The structural exploration has focused on diversity of the partners, i.e.

[2] International patent classification developed by WIPO in 1971. "a hierarchical system of language independent symbols for the classification of patents and utility models according to the different areas of technology to which they pertain" http://www.wipo.int/classifications/ipc/en/

whether a firm is forming an alliance with a new partner that has no prior ties to the firm or keeps recurrent alliances with a partner that has prior ties to the firm. At this stage of the research, there is no complete data to discuss the function and attribute domains. However, the structural domain, as reviewed in the literature [22], can be studied as a representative element of the exploration/exploitation process. To calculate the exploration index of firms, we used the Herfindahl index and Simpson's diversity index:

$$D = 1 - \sum_{i=1}^{N} p_i^2$$

where p is the proportion of particular partner in the ego network of firm i and N total number of all partners which were collaborating by firm i.

4.3 Structure: Firms' Network Centrality and Density

4.3.1 Density

As mentioned earlier, the density of a firm's network (ego network of a firm) has been argued in the literature (Cumming and Cross 2004) as an influencing factor on knowledge transfer through inter-organisational relationships. In our research, ego-network analysis in UCINET6 was used to calculate the density for each node. In this method, density is measured by number of existing triangles divided by all possible triangles. The highest density is one and the lowest is zero. The density measure for both isolated firms (firms with no ties to any other partner) and for firms without any triangle relations like BF94 (Fig. 1) is zero.

4.3.2 Centrality

Centrality is another structural property that has been argued in the KM literature as an influencing factor on knowledge transfer through inter-organisational relationships (e.g. [1]). There are several measures (methods) to calculate centrality like degree, eigenvector, closeness and betweenness centrality. For instance, degree of a firm in a given network is calculated by total number of ties shared by the focal firm with other members of the network. For example, the degree of BF94 with three ties (Fig. 1) is three. In this research, all these measures were calculated by UCINET6, but for the analysis, degree of centrality is used.

4.4 Four Possible Structural Configurations

As mentioned before, centrality and density are two major measures of the structure of a network [1]. In this regard, based on the centrality of firms, whether it is high

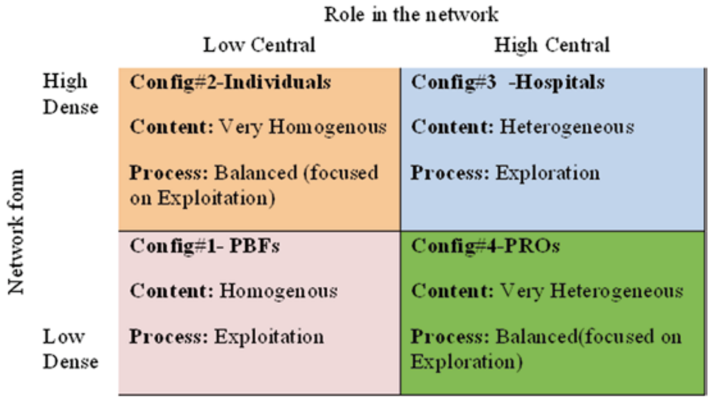


Fig. 2 A quadrant framework to represent the interactions among content, process and structure of the Victorian bioindustry from 2001 to 2010

or low, and also based on the network, whether it is dense or sparse, there are four possible configurations (Fig. 2). The four configurations were examined to see whether they have different knowledge content and process. The characteristics of the firms in each configuration are explained in the summary Sect. 4.6.

4.5 Exploring the Interactions Among K-Network Content, Process and Structure

According to a quantitative analysis of the relationships between content, process and structure of the k-networks, the following themes have emerged. The summary of the finding and corresponding statistical tests are illustrated in Table 1:

Theme 1: In bio-firms the structural exploration process is associated with knowledge heterogeneity

It seems that companies that seek more diverse and new partners are considered as companies with structural exploration process. Our analysis shows that these companies are more willing to seek new and more diverse knowledge.

Theme 2: Bio-firms with a central position in the innovation network access more heterogeneous knowledge

The more central position provides access to a broader range of knowledge. In other words, the higher degree of knowledge heterogeneity that an organisation possesses is associated with the position of the organisation in its network. The scatter plot

Table 1 Summary of the findings and corresponding statistical tests

Findings		Test	Sig level	Result
Theme 1: process and content	Exploration has a positive correlation with knowledge heterogeneity	Pearson correlation	0.003 correlation is significant at the 0.01 level (2-tailed)	Not rejected
Theme 2: content and centrality	Degree centrality has a positive correlation with knowledge heterogeneity	Pearson correlation	0.000 correlation is significant at the 0.01 level (2-tailed)	Not rejected
Theme 3: content and density	Density has a negative correlation with knowledge heterogeneity (after removing isolated 58 nodes)	Pearson correlation	0.021 correlation is significant at the 0.05 level (2-tailed)	Not rejected
Theme 4: process and centrality	Degree centrality has a positive correlation with exploration	Pearson correlation	0.000 correlation is significant at the 0.01 level (2-tailed)	Not rejected
Theme 5: process and density	Density has a positive correlation with exploration	Pearson correlation	0.000 correlation is significant at the 0.01 level (2-tailed)	Not rejected
Theme 6: process within four configurations	Level of exploration is not equal within firms with different configuration	Non-parametric—Friedman test	C3E > C4E > C2E > C1E 0.001 significant at the 0.01	Not rejected
	Firms with C3 configuration (high centrality and density) have the highest level of exploration	Non-parametric—related samples Wilcoxon signed-rank test	C3E > C1E 0.000 C3E > C2E 0.003 C3E > C4E—not supported 0.515	Weakly rejected
	Firms with C1 configuration (low centrality and density) have the lowest level of exploration		C3E>C1E 0.000 C4E>C1E 0.008 C2E>C1E 0.002	Not rejected
	Firms with C2 configuration (low centrality and high density) have the higher level of exploration compared with firms with C4 configuration (high centrality and low density)		C2E < C4E—not supported 0.139	Wealth rejected
Theme 7: content within four configurations	Level of knowledge heterogeneity is not equal within firms with different configuration	Non-parametric—Friedman test	C4KH > C3 KH > C1 KH > C2 KH 0.000	Not rejected
	Firms with C4 configuration (high centrality and low density) have the highest level of knowledge heterogeneity	Non-parametric—related samples Wilcoxon signed-rank test	C4KH > C1KH 0.008 C4KH > C2KH 0.008 C4KH > C3KH 0.008	Not rejected
	Firms with C2 configuration (low centrality and high density) have the lowest level of knowledge heterogeneity		C2KH < C3KH 0.011 C2KH < C1KH—not supported 0.345	Weakly rejected
	Firms with C1 configuration (low centrality and density) have the lower level of knowledge heterogeneity compared with firms with C3 configuration (high centrality and density)		C3KH > C1KH 0.010	Not rejected

also shows a positive correlation between these two measures. This finding is supported by the literature as well [1]. The more central position allows firms a chance of accessing various knowledge areas.

Theme 3: Bio-firms with dense networks possesses less heterogeneous knowledge

Dense networks allow a firm a chance of accessing close relationships with partners, which in the long term may cultivate trust in their relations [45] and may provide more access to strategic knowledge [46]. However, the domain of this knowledge is narrow.

Theme 4: Bio-firms with a central position in the innovation network are more likely to seek structural exploration process

The more central position provides access to a broader range of partners. In this regard, companies with high level of contacts might find new partners more easily.

Theme 5: Bio-firms with dense networks are more likely to seek structural exploration process

In the networking literature, close relations and having a dense network are associated with the concept of homophily [17]. "Birds of a feather flock together" means these partners are mainly homophile [37]. However, the high level of association between density of the firm's network and exploration index shows that bio-firms with dense networks may not necessary be limited with a small number of partners. Adding the concept of heterogeneity (theme 3), it is possible to argue that the reason for new partnership is not always about accessing new knowledge. It seems that in Victorian innovation networks, there are several companies that seek new partners while they are still working on the existing knowledge, probably for finding new markets via new partners.

Theme 6: Level of exploration is not equal within firms with different configurations

Theme 6.1: Bio-firms with a C1 configuration (low centrality and density) have the lowest level of exploration. These firms are mainly isolated or have no links to firms with a central position. On the other hand, they have a few close relationships. In this regard, these firms are not willing to find new partners for their innovation projects.

Theme 6.2: A central position in an innovation network seems more influential on structural exploration compared to density of the network. Regardless of the density of their innovation networks, firms with a high central position have a high level of structural exploration. Firms with a C3 configuration (high centrality and density) do not have the higher level of exploration compared to a C4 configuration (high centrality and low density). Although firms with a C4 configuration (high centrality and low density) have no significantly higher level of exploration compared to a C2 configuration (low centrality and high density), firms with dense networks do not necessarily have a high level of exploration (i.e. C3 firms have significantly higher level of exploration compared to C2 firms).

Theme 7: Level of knowledge heterogeneity is not equal within bio-firms with different configurations

Theme 7.1: Bio-firms with a C4 configuration (high centrality and low density) have the highest level of k-heterogeneity. As mentioned in themes 2 and 3, centrality improves access to more heterogeneous knowledge, while dense networks keep focus on a narrower domain of knowledge. So C4 bio-firms show significantly highest level of k-heterogeneity.

Theme 7.2: Central position in an innovation network seems more influential on k-heterogeneity compared to density of the network. Firms with C1 configuration (low centrality and low density) have a lower level of k-heterogeneity compared with firms with a C3 configuration (high centrality and high density). Also, there is no significant difference between the level of k-heterogeneity in C2 firms (low centrality and high density) and C1 firms (low centrality and low density).

4.6 Summary

In summary, it is possible to conceptualise the research findings on interactions of the content, process and structure of the k-network into a quadrant framework represented in Fig. 2. Organisations in C1 (mainly large public research organisations like Monash University and the University of Melbourne) have the highest degree of heterogeneity. They have access to broad range of knowledge since they locate themselves in central positions. Meanwhile they have a balanced approach for their processes. This means that they are always looking for new knowledge whether from new partners or existing ones. On the other hand, organisations in C2 have focused on a very narrow area of knowledge. They keep their focus on reusing their existing knowledge but through balanced process. It means they either work closely with their existing partners or search for new partners to reuse the existing knowledge. However, organisations with C1 and C3 seem different as they are in the middle of the continuum of knowledge heterogeneity. C3 organisations, on the one hand, have focused on moderate to high degrees of knowledge heterogeneity. But unlike C4 organisations, after they find new knowledge from new partners, they keep close relations with their partners to reuse the knowledge. Most of these companies in the Victorian biotech network are hospitals. On the other hand, C1 companies which have very low dense network with decentred positions possess a broader level of knowledge compared with C2 companies. This means that most C1 organisations work on their own R&D to acquire new knowledge and reuse it.

5 Conclusion

While the main focus of the KM literature is on intra-organisational aspects (e.g. [32]), in innovation-driven collaborations acquisition or access to external knowledge via interfirm k-networks is the main point of interest. In this regard, our research studies the characteristics of interfirm k-networks and highlights the need to consider the interplay of all the elements that constitute an inter-organisational network. In this paper, many seemingly fragmented theories including network structure theory (e.g. [15]), theory on content of knowledge (e.g. [3]) and knowledge processes [43] are utilised to propose a conceptual framework to enhance the understanding about k-networks. This framework is empirically supported by our analysis of the Victorian patent network.

Participating in a network is one of the most important challenges that an entrepreneur faces since the beginning of their business [44]. However, to manage their networking practice, this paper suggests that entrepreneur may need to consider the interplay between knowledge content, knowledge processes and the network structure as context that informs their collaborations. This content-process-structure framework can help entrepreneur to evaluate their organisational mechanisms as managerial interventions to facilitate such innovations.

This paper is based on the first exploratory phase of an ongoing research project. The conceptual quadrant framework proposed in this paper will be the focus of further in-depth studies of k-networks in the Victorian biotechnology sector to give a rich understanding of the contribution of inter-organisational collaboration to innovation practices.

References

1. Ahuja G (2000) The duality of collaboration: inducements and opportunities in the formation of interfirm linkages. Strat Manag J 21:317–343
2. Allen Consulting Group (2010) Victorian biotechnology industry skills review: Report to the Victorian Department of Innovation, Industry and Regional Development. http://www.allenconsult.com.au/publications/download.php?id=352&type=pdf&file=1. Accessed 2 Sep 2011
3. Barney JB (1991) Firm resources and sustained competitive advantage. J Manag 17(1):99–120
4. Bierly PE, Damanpour F, Santoro MD (2009) The application of external knowledge: organizational conditions for exploration and exploitation. J Manag Stud 46(3):481–509
5. Bio (2008) http://www.bio.org/. Accessed 2 Sep 2011
6. Bustamante MA (2007) Linking properties of knowledge and knowledge network topology with performance, The Graduate College at the University of Nebraska
7. Cantner U, Graf H (2006) The network of innovators in Jena: an application of social network analysis. Res Pol 35:463–480
8. Chen J, Ngaib EWT, Tonga L (2007) Inter-organizational knowledge management in complex products and systems: challenges and an exploratory framework. J Tech Manag China 2:134–144
9. Cummings J, Cross R (2004) Tie and network correlates of individual performance in knowledge-intensive work. Acad Manag J 47:928–937
10. Demirkan, I., & Demirkan, S. (2012). Network characteristics and patenting in biotechnology, 1990-2006. *Journal of Management, 38*(6):1892–1927

11. Ding HB, Peters LS (2000) Inter-firm knowledge management practices for technology and new product development in discontinuous innovation. Int J Technol Manag 20:588–600
12. Edwards T (2007) A critical account of knowledge management: agentic orientation and SME innovation. Int J Entrepreneurial Behav Res 13(2):64–81
13. Ernst and Young (2011) Beyond borders: the global biotechnology report 2011. Ernst and Young, Boston
14. Giannakis M (2008) Facilitating learning and knowledge transfer through supplier development. Supply Chain Manag Int J 13(1):62–72
15. Hansen MT (2002) Knowledge networks: explaining effective knowledge sharing in multiunit companies. Organ Sci 13(3):232–248
16. Inkpen AC, Tsang EWK (2005) Social capital, networks, and knowledge transfer. Acad Manag Rev 30(1):146–165
17. Kadushin C (2004) Introduction to social network theory. http://stat.gamma.rug.nl/snijders/Kadushin_Concepts.pdf. Accessed 16 Apr 2010
18. Kane GC, Alavi M (2005) Casting the net: a multimodal network perspective on knowledge management. Proceedings: twenty-sixth international conference on information systems, pp 233–246
19. Kim C, Song J, Nerkar A (2011) Learning and innovation: exploitation and exploration trade-offs. J Bus Res doi:10.1016/j.jbusres.2011.07.006
20. Kraaijenbrink J, Wijnhoven F (2008) Managing heterogeneous knowledge: a theory of external knowledge integration. Knowl Manag Res Pract 6(4):274–286
21. Kraaijenbrink J, Wijnhoven F, Groen AJ (2007) Towards a kernel theory of external knowledge integration for high-tech firms: exploring a failed theory test. Technol Forecast Soc Change 74(8):1215–1233
22. Li Y, Vanhaverbeke W, Schoenmakers W (2008) Exploration and exploitation in innovation: reframing the interpretation. Creativ Innovat Manag 17(2)
23. March JD, Levinthal DA (1993) The myopia of learning. Strat Manag J 14:95–112
24. Moslehi A, Linger H, Tanner K (2011) Towards a KM framework for e-business projects within SMEs. 17th Americas conference on information systems, AMCIS 2011, Detroit, Michigan, USA
25. Mowery DC, Oxley JE, Silverman BS (1996) Strategic alliances and interfirm knowledge transfer. Strat Manag J 17:77–91
26. Owen-Smith J, Powell WW (2004) Knowledge networks as channels and conduits: the effects of spillovers in the Boston biotechnology community. Organ Sci 15(1):5–21
27. Powell WW, Koput KW, Smith-Doerr L (1996) Interorganizational collaboration and the locus of innovation: networks of learning in biotechnology. Admin Sci Q 41(1):116–146
28. Rodan S, Galunic D (2004) More than network structure: how knowledge heterogeneity influences managerial performance and innovativeness. Strat Manag J 25:541–562
29. Salman, N. (2002) Networks and innovation: a social network analysis of biotechnology collaboration. Masters thesis, Concordia University
30. Sammarra A, Biggiero L (2008) Heterogeneity and specificity of inter-firm knowledge flows in innovation networks. J Manag Stud 45:785–814
31. Scarbrough H (2003) Knowledge management, HRM and the innovation process. Int J Manpow 24(5):501–516
32. Skerlavaj M, Dimovski V, Mrvar A, Pahor M (2010) Intra-organizational learning networks within knowledge-intensive learning environments. Interact Learn Environ 18(1):39–63
33. Szulanski G (1996) Exploring internal stickiness: impediments to the transfer of best practice within the firm. Strat Manag J 17(Winter Special Issue):27–43
34. Tichy NM, Tushman M, Fombrun C (1979) Social network analysis in organizational settings. Acad Manag Rev 4:507–519
35. Walker G, Kogut B, Shan WJ (1997) Social capital, structural holes, and the formation of an industry network. Organ Sci 8(2):109–125
36. Winter SG, Szulanski G (2001) Replication as strategy. Organ Sci 12(6):730–743

37. Yang XU (2006) Social capital, cognitions, and firm innovation: theoretical model and empirical studies, PhD thesis, Virginia Polytechnic Institute and State University
38. Schumpeter, J.A.. (1961). The theory of economic development: an inquiry into profits, capital, credit, interest, and the business cycle. Transaction Publishing, New Brunswick, USA.
39. Brass, D. J. (1984). Being in the right place: A structural analysis of individual influence in an organization. *Administrative Science Quarterly, 29*:518–539
40. Friedkin NE (1993) Structural bases of interpersonal influence in groups: A longitudinal case study. American Sociological Review 58:861–872
41. Ibarra H (1993) Personal networks of women and minorities in management: A conceptual framework. Academy of management Review 18(1):56–87
42. Fleming L, King C, Juda A (2004) Small worlds and innovation. Harvard Business School, Mimeo
43. Lavie D, Rosenkopf L (2006) Balancing exploration and exploitation in alliance formation. Academy of Management Journal, 49(4):797–818
44. Szarka, J. (1990). Networking and small firms. International Small Business Journal, 8(2): 10–22
45. Rice, R. E., & Aydin, C. (1991). Attitudes toward new organizational technology: Network proximity as a mechanism for social information processing. Administrative science quarterly, 36:219–244.
46. Miranda, S. M., & Saunders, C. S. (2003). The social construction of meaning: An alternative perspective on information sharing. Information systems research, 14(1), 87–106.

Global Development, ICT4D, Education for All

Anders G. Nilsson and Thérèse H. Nilsson

Abstract Global development has the mission to promote social change and renewal in developing countries on a scientific basis. For this purpose the paper presents a concept for creating five good circles for society building of poor communities. This concept is further elaborated and discussed in terms of sustainability development and sustainable information systems. For one of the good circles "active learning", we illustrate how a deliberate use of tools for ICT4D (ICT for Development) can support special education for needy children and exposed mothers. Our knowledge interest is to set up professional training spaces, called Community Children Academy Spaces (CAS), grounded on an Education for All (EFA) perspective for social inclusion of poor people in civil societies.

1 Global Development for Change

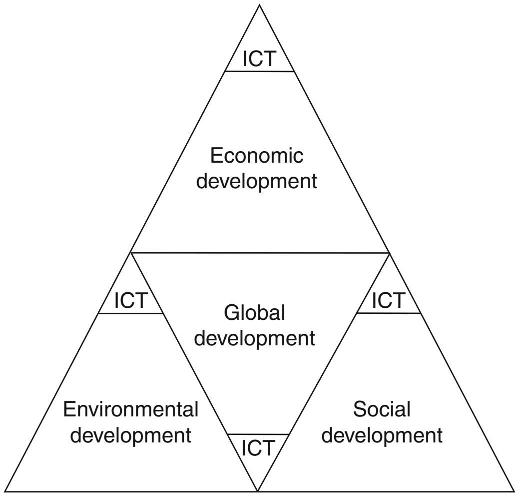
Global development has the mission to promote social change and renewal in developing countries on a scientific basis. The leading idea behind global development is to combine social changes in civil societies with economic growth and environmental (ecological) issues in order to achieve better life conditions for poor people. Development informatics or ICT4D (ICT for Development) is a field of both research and practice focusing on the application of information systems in socio-economic development [18, 20], especially the use of ICTs (Information and Communication Technologies) in developing countries.

A.G. Nilsson (✉)
Department of Computer and Systems Sciences, Stockholm University,
Stockholm, Sweden
e-mail: agn@dsv.su.se

T.H. Nilsson
Universal Child Welfare Foundation (UCWF), Karlstad, Sweden
e-mail: ucwfoundation@hotmail.com

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_26,

Fig. 1 Fostering global development with ICT for economic, environmental and social issues



Global development will use a well-known approach labelled *Base of the Pyramid* (*BOP*) which is perfectly timed for developing countries [5]. This means that the new economy of the country is growing through developing the poor population to be self-dependent, productive and respected citizens. The poor people are regarded as active business partners, value creators and demanders, strong innovators and creative entrepreneurs rather than only "victims" for potential producers or consumers. This view is in line with a service-dominant logic approach for society building using local business models [11]. Some concrete examples of BOP initiatives are micro-credits for exposed women, market-specific products for poor people and business community partnerships in developing countries [27]. Such a global development with BOP initiatives is performed through integrating social programmes with environmental and economic development of the country. The goal is to align business, earth and humanity strongly to each other. According to a BOP approach, all development areas (or change measures) must in a future modern civil society be more integrated with different types of ICT4D tools such as professional use of social media, Digital Doorways, digital maps, mobile phones and Internet search of knowledge. In Fig. 1 it is shown that ICT is needed today to support and enable each development area where global development is a combination of economic development, environmental development and social development.

2 The Main Concept: Creating Five Good Circles

For achieving an efficient global development, there is a need for a major interplay between economic, environmental and social issues for improving the life situation for poor people in developing countries. The focus here is to study the relationship

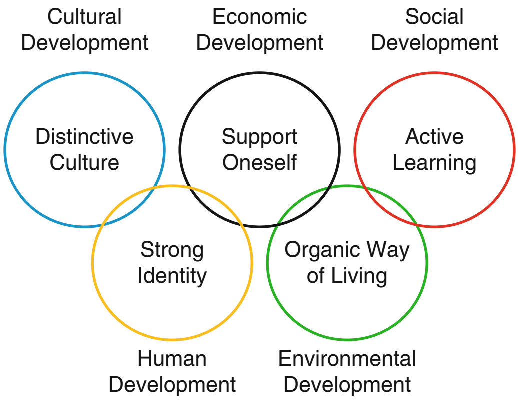


Fig. 2 Concept for global development by creating good circles for society building

between needy children and exposed mothers as a target group. By this we mean reaching vulnerable and marginalised people living in the worst environments with a great jeopardy of poverty and violence [23].

The main idea is to break vicious cycles or better to create good circles for overcoming obstacles and changing the prerequisites for poor people. The good circles represent a valuable platform for creating enablers for change in civil societies (communities) which is the aim for global development. We have identified *five kinds of good circles* for society building which is symbolised by the Olympic rings known by many people in the development world: distinctive culture, strong identity, support oneself, organic way of living and active learning (see Fig. 2).

2.1 Distinctive Culture

Promote the dominant values, ethics and morale that are prevailing in the civil societies and communities we want to understand and change. A programme for global development has to be adapted to the cultural norms characterised by the local people. Therefore it is needed to gain a deeper knowledge of the region's culture and history as a base for a social change [7]. The dominant culture in a society refers to the established language, religion and spirituality, behaviour, values, rituals and social customs [12]. These traits are often the cultural norms for the civil society or community as a whole. In a multicultural society, various subcultures are celebrated and should therefore be respected equally. The first good circle for society building refers to *cultural development* as prerequisite for global development.

2.2 Strong Identity

Promote democratic principles and human rights for poor people in the civil society [9]. An important circumstance is to create a genuine atmosphere of social justice, peace building and secure environments for the families living in their communities [14]. This represents a political dimension of society building. A programme for global development should give people such as needy children and exposed mothers a genuine dignity and strengthened self-confidence based on their strong identity. As an example we can mention to organise valid ID documents (or identity cards) as official papers for refugees and transit families living in temporary communities. The second good circle for society building refers to *human development* as prerequisite for global development.

2.3 Support Oneself

Promote economic principles for professional enterprising and business trade. A basic idea behind a programme on global development is to create occupational possibilities for mothers and families from, e.g. exploiting organic plantations (coffee, cacao, fruits and vegetables) connected to community-based children spaces. This is an application of "Base of the Pyramid (BOP)" approach described above [5]. The poor families are regarded as active business partners, value creators, innovators and entrepreneurs. The exposed mothers could be assisted by a microfinance system for starting up small businesses [27]. Creating job opportunities for poor mothers gives them the possibility to be able to "support oneself" and taking professional care of their children. The third good circle for society building refers to *economic development* as part of global development.

2.4 Organic Way of Living

Promote organic principles for social care in a creative family atmosphere. An important guiding star for a programme on global development is to promote an organic way of living for women and children in order to strengthen their physical health and mental capacity with ecological and nutritious food. According to the United Nations Environment Programme [22], organic agriculture not only can feed the world, it may be the only way we can solve the growing problem of hunger in developing countries. Biodynamics is an internationally recognised method to organic agriculture in which the farmer respects and works with the spiritual dimension of the earth's environment [16]; in other words it promotes the human's interaction with nature based on an anthroposophic tradition. We have faced some successful projects on organic farming with biodynamics such as the Rio Limpio experiment for poor people in the Dominican Republic. Some other elements for an organic way of living would be sustainable production of goods (e.g. cloths), reusing organic wastes (efficient ecocycling) and the rising supply of renewable energies. The fourth good circle for society building refers to *environmental development* as part of global development.

2.5 Active Learning

Promote experience-based principles for active learning in order to achieve a sustainable basic education for children and a preparatory vocational education for mothers [4]. A leading idea for this kind of initiative is to have a situated learning situation where we combine indoor studies with practical exercises in an outdoor environment. Waldorf pedagogy is an internationally recognised approach for active and situated learning for children at school. The Waldorf approach to education promotes problem solving, creative learning, visual stimulation, competence enhancement and reality-based applications. An important goal for global development is to work out special education programmes for needy children and exposed mothers to raise their abilities for communication and concentration based on constructive learning [15]. This approach will facilitate an "inclusive education" which means that the poor people will be supported by special education based on approaches for active and situated learning in order to include or integrate them better in the civil societies where they live—"A Society for All" (cf. [25]). The fifth good circle for society building refers to *social development* as part of global development.

3 Sustainable Development and Sustainable Information Systems

The constructs and relationships in Fig. 2 of the five good circles will be further elaborated and explained in this section. There are a lot of interactions or synergies between the different development areas for global change. As an example we can emphasise that exploitation of organic or nutritious foods (environmental development) will give possibilities for raising the learning capacity for children in school (social development) which later leads to possibilities for good employments and raised welfare in the future (economic development).

We can recognise the interactions between development areas as the relationships between cultural, human, economic, environmental and social aspects of sustainability. The five good circles as a theoretical approach (Fig. 2) is a representation of the five development areas of sustainable development for society building. The interactions between development areas can be symbolised by the interlocked connections in the Olympic rings for the good circles (all possible interactions are not shown in a convenient "Olympic ring" look). Sustainable development is thus an overall concept which embraces a holistic or helical approach for building civil societies in developing countries. Sustainable development more precisely comprises cultural development, human development and global development. Cultural and human developments are necessary prerequisites for a global development. Global development is in turn a combination of economic development, environmental development and social development. This kind of reasoning can be summarised as follows:

- Sustainable development = (holistic or helical) development
- Sustainable development = (cultural + human + global) development
- Global development = (economic + environmental + social) development

The concept of *sustainable development* was officially recognised by the Brundtland Commission [3], and the term's origin was described as follows: "Sustainable development is development that meets the needs of the present without compromising the ability of future generations to meet their own needs." The concept of "needs" was in particular directed towards the essential needs of the world's poor, to which overriding priority should be given. Later on the concept of "sustainable development" has been broken in three constituent parts: environmental, economic and sociopolitical sustainability in the Agenda 21 action plan [22]. This definition is in line with the newer concept of global development. We have broadened the concept of "sustainable development" to include important issues on cultural and human development as well as essential interactions between the five above-mentioned development areas.

The design and use of modern ICT tools is today a necessary condition for success when supporting each development area with vital and critical information as well as to enable and promote the interactions between the development areas by strategic information and communication flows. This is a main characteristic for green ICT and *sustainable information systems* (cf. [26]). In this sense the information systems should support important sustainable initiatives for strengthening of cultural values, human rights, health, work and education for poor people (cf. [13]). In other words such a green ICT is essential for all the five good circles described in Fig. 2.

4 The Use of ICT4D for Active Learning

We will here concentrate on the fifth good circle on "active learning" for society building because we have an "Education for All" perspective in this study. The goal is to create learning spaces for needy children and their exposed mothers with help of modern tools for *ICT4D* as a theoretical approach for *active learning* [6]. The *primary argument* for ICT spaces for active learning is as follows. Children in general are very excited and amused by working with computers! This is the main argument for setting up a special ICT spaces for needy children together with their mothers. When children are using computers, they have a possibility to develop both cognitive and mobility competence. The child can experiment, acquire new experiences, obtain stimulation, make independent choices, develop a power of initiative, become confident, increase innovativeness, practise communication, balance mobility and concentration as well as learn how to handle mistakes [10]. An essential outcome of an ICT space is also to give mothers possibilities through digital media to find new job opportunities so that they can take professional care of themselves and their children [5]. This is a *secondary argument* for setting up ICT spaces in developing countries. But we should be aware of the complex issues that could act as barriers to overcome for women and children in some cultures to be introduced for ICT spaces.

An effective way of for establishing ICT spaces is to use so-called multimedia studios for children. This means a computer play centre which offers pedagogical software for training and learning purposes. The primary aim is to raise children's power of concentration and ability to communicate with their environment. For all children the multimedia products gives an arena for playing activities. There exist many kinds of professional multimedia programmes from serious vendors on the market. Most of the products have a twofold goal to support the children's need for educational and playing activities. Computer games are a method of making play possible and fun while developing the cognitive abilities and independence of the child. The optimum size of the multimedia studio is around ten fully equipped computers. Besides multimedia applications, there should be installed programmes for office products and Internet applications [18].

There are also other technological possibilities for building up ICT spaces (cf. [24]). One of those is a mobile phone laboratory with image-talk technology. Another possibility is "smart home" technology for children with cognitive disturbances. A third possibility is using computers for creating art and music. A fourth possibility is the well-known model of "one laptop per child" (OLPC) for primary schools in developing countries. This concept could be extended to "one tablet per child" (OTPC) and "one mobile per child" (OMPC) for the future.

In the area of development informatics and ICT4D, there are some recent initiatives about using so-called Digital Doorways, ICT hubs and e-skills for supporting indigenous communities in developing countries, foremost in the South African culture [19]. These experiments are of great interest and utmost inspiring for creating ICT spaces for needy children and exposed mothers—as an effective tool for their empowerment and social inclusion! The leading idea behind the concept of *Digital Doorway* is to provide poor people in disadvantaged and high-need communities with freely accessible computer equipment and open source software, enabling them to experiment and learn without formal training and with minimal external input. This is based on a concept for minimally invasive education. A Digital Doorway technology could be connected to Internet access but is rather often managed through a database copy of relevant web pages for cost reasons. A Digital Doorway concept is realised and implemented by a stand-alone solution or a robust multiterminal kiosk as part of a "Living Lab" approach. Digital Doorways can be used for educational purposes—resources such as tutorial software and encyclopedias could be installed to motivate self-learning or to support classroom learning for needy children and group learning (e.g. study circles) for exposed mothers. Furthermore, a Digital Doorway approach will open up new pathways for promoting small businesses and social networking which is a valuable resource for organising job opportunities to exposed mothers.

In a research study by Barnett and Ashman [2], they propose an interesting theoretical framework for the use of ICT tools in teaching young children with a global perspective. The increasing availability of information and communication technologies (ICTs) in education opens up many new possibilities for promoting the learning processes for needy children and exposed mothers. The proposed framework consists of four types of learning processes requiring different kinds of ICT4D tools as follows:

- Access ideas and information
- ICT tools: websites, CDs-DVDs-videos, simulation games, learning objects
- Extend ideas
- ICT tools: mind-mapping software, drawing programmes
- Link to others
- ICT tools: e-mailing, online projects, Internet forums, videoconferencing
- Share ideas
- ICT tools: presentation programmes, intranet, word processing, animations

The proposed framework is a rather extensive description of necessary ICT4D use that can be applied to different education situations. It represents a constructive approach for setting up appropriate ICT spaces for children and mothers based on principles for active and situated learning. In recent times we have got access to modern and intelligent learning management systems (LMS), professional knowledge sharepoints, mobile life tools and attractive learning tablets (e.g. iPads) for efficient use to improve special education in developing countries. These *ICT4D* tools support together a blended learning approach for *active learning*.

5 Education for All and Social Inclusion

The Millennium Development Goals (MDGs) are worked out by the United Nations (UN) to be achieved from 1990 until 2015 that respond to the world's challenges for global development [23]. These goals are oriented on social transformation in developing countries regarding, e.g. children education and women empowerment. UNESCO states that education empowers people with the knowledge and skills to improve themselves and their lives. The education goals of the MDGs have been elaborated and broadened up by UNESCO through defining various goals to achieve an "Education for All (EFA)" by 2015 [21]. We will here focus on a systems approach as a theoretical foundation for reaching the *Education for All* goals regarding the need for special education as a *social inclusion* (cf. [25]) for needy children and exposed mothers together with their affected teachers as follows:

- Millennium Development Goals (MDGs):
- Goal M2: "Achieve universal primary education for children"
- Goal M3: "Promote gender equality and empower women"
- Education for All (EFA) Goals:

- Goal E2: "Provide free and compulsory primary education for all"
- Goal E3: "Promote learning and life skills programmes for young people and adults"
- Goal E6: "Improve the quality of education"
- Special Education Goals:
- Goal S1: "Achieve basic education for needy children" (goal M2, E2)
- –All children can attend the school.
- Goal S2: "Achieve preparatory vocational education for exposed mothers" (goal M3, E3)
- –All mothers can take care of their children.
- Goal S3: "Achieve professional teacher education for affected teachers" (goal E6)
- –All teachers can support the individual learning processes of children and mothers.

We have specified three different goals on special education for the target group concerned derived from the Millennium Development Goals (MDGs) and the Education for All (EFA) Goals. These special education goals are about to create the good circle on "active learning" for society building. The leading goal is that all children can attend the school and get a necessary primary or basic education in a modern civil society (goal S1). There are a lot of entrance barriers for children to attend schools in developing countries such as the following obstacles:

- The parents or mothers have negative attitudes to the children's schooling.
- The family cannot afford school costs for children's cloths (uniforms).
- Children have psychological traumas hindering them to concentrate in school.
- Shortage of professional and qualified teachers.
- Absence of a school place or learning space for the children in the community.

The *systems approach* is to overcome the entrance barriers and existing obstacles. A systems approach can be used for solving such societal problems as a base for redesigning the future [1]. For achieving this we need to have a relationship perspective between the main actors. The child will be in the centre of attraction. The child has first of all an affectionate and attached relationship to the mother [8]. Many needy children in developing countries have traumatic experiences in life and are therefore characterised as "slow learners". It is therefore necessary to design special education programmes for these disadvantaged children in order to stimulate their ability to communicate and concentrate in school [10]. Also for the mothers it is urgent to have special education programmes on preparatory vocational education for giving them job opportunities and making them more self-dependent in order to take care of their children (goal S2). This kind of special education should also contain coursework on social child care with health and nutrition, for understanding the children's needs as well as influencing mother's attitudes to promote children's schooling. It is a well-known fact that many teachers in developing countries need a professional support to give better services to needy children and their exposed mothers. Therefore it is necessary to design professional teacher education programmes focusing on how the teacher can promote the individual learning processes of children and mothers in the best way (goal S3).

A purposeful systems solution for reaching the specified goals of special education programmes for needy children, exposed mothers and affected teachers is to set up professional training spaces, which we label *Community Children Academy Spaces* (*CAS*), based on learning workshops, modern therapy forms and outdoor activities together with the use of professional tools for ICT4D [6, 24]. It is important to utilise the existing infrastructures in civil societies. The base environment is ordinarily represented by the family home and the public or private school. But the home and school environments are necessary but not sufficient conditions for implementing professional special education programmes.

Therefore it is a great need for a supplementary training space where building up civil societies through Community Children Academy Spaces (CAS) gives rich opportunities for a professional education support to the needy children and their exposed mothers. Such a community space will be progressive, constructive and full of life for supporting children and mothers in their civil societies. The purpose is to raise their power of concentration and ability to communicate with each other and the surrounding world. There are evidences from earlier research that supplementary learning spaces outside the "classroom" of traditional schools would benefit the children's physical experiences, social interaction, emotional well-being and responses. This is in line with the basic theory on education in development contexts outlined from a liberating pedagogy approach [4].

In a Community Children Academy Space (CAS), there should be room for a professional ICT space as a major strategy for the special education support to the children, mothers and teachers. Such an ICT space would be based on the principle for active and situated learning as described above. A leading motive for the community space is an organic way of living for the children and their mothers. The training programmes should therefore also provide knowledge on how organic nutrition and biodynamic food influence the balance between human health and the natural environment. A communal kitchen could therefore be connected to the space for serving much needed food and nutrition. The Waldorf pedagogy combines school education with a biodynamic and anthroposophic tradition.

In Fig. 3 we show a systems approach for *Community Children Academy Spaces* (*CAS*) as a constructive supplement to the *home* and *school* environments for the target group of needy *children*, exposed *mothers* and the affected *teachers*. The *five good circles* for society building (Fig. 2) are the basic foundation of a systems solution for CAS where *active learning* and special education is the centre of attraction. We will label this as a learning space strategy for community development focusing on *Education for All* and *social inclusion*.

According to a systems approach, it is important to start up initiatives such as Community Children Academy Spaces (CAS) in a small scale as a platform for further expansion and continual growth—"from small things, big things grow" (from a well-known indigenous aboriginal song in Australia)! A scenario of *mini spaces* for special education support could be very attractive and effective as a starting point for building up new civil societies in developing countries. This idea is close to the concept of community-based home schools run by trained community

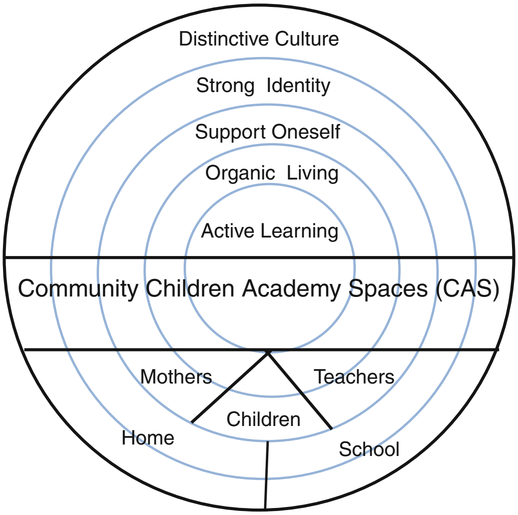


Fig. 3 A systems approach for Community Children Academy Spaces (CAS) as social inclusion

members in their family homes—maximum of six children per home school is the ideal [17]. These home schools are therapeutic units of which basic skills training, informal education and schooling form an important part.

6 Concluding Thoughts

We will end up with three concluding thoughts as a base for future research and action in the development field—in line with the title of this paper "Global Development, ICT4D, Education for All":

- Global Development
- Global development is the new venture for development aid policy. The main idea is to combine economic, environmental and social development for promoting positive change and renewal in poor countries. The globalisation creates new opportunities for vulnerable people to improve their life circumstances.
- ICT4D
- Various ICT4D tools should together support a blended learning approach. We have today got access to modern e-learning platforms, professional knowledge sharepoints, mobile life tools and attractive learning tablets for efficient use to improve special education and active learning in developing countries.
- Education for All
- Best practices in many developing countries have shown that basic education for needy children and vocational education of exposed mothers are necessary prerequisites for creating a better future life. Education and learning is a lifetime investment for poor people and opens up new "windows" of opportunities.

References

1. Ackoff RL (1974) Redesigning the future: a systems approach to societal problems. Wiley, New York. ISBN 9780471002963
2. Barnett J, Ashman G (2008) Thinking globally: global perspectives in the early years classroom. Curriculum Corporation, Australian Government, Carlton South, VIC. ISBN 9781863666404
3. Brundtland Commission (1987) Our common future. The Brundtland Report from the United Nations World Commission on Environment and Development (WCED). Oxford University Press, Oxford. ISBN: 9780192820808
4. Freire P (1970) Pedagogy of the oppressed. Continuum, New York. ISBN 0826412769
5. Hart SL (2007) Capitalism at the crossroads: the unlimited business opportunities in solving the world's most difficult problems. Pearson Education and Wharton School Publishing, Upper Saddle River, NJ. ISBN 9780136134398
6. Heeks R (2008) ICT4D 2.0: the next phase of applying ICT for international development. Computer 41(6):26–33
7. Hofstede G, Hofstede GJ, Minkov M (2010) Cultures and organizations: software of the mind. McGraw-Hill, New York. ISBN 9780071664189
8. Holmes J (1993) John Bowlby and attachment theory. Routledge, London. ISBN 9780415077309
9. Ishay MR (2008) The history of human rights: from ancient times to the globalization era. University of California Press, Berkeley. ISBN 9780520256415
10. James A, Jenks C, Prout A (1998) Theorizing childhood. Polity Press, Cambridge, UK. ISBN 9780745615646
11. Lusch R, Vargo S (2006) Service dominant logic: reactions, reflections, and refinements. Market Theor 6(3):281–288
12. Paulk JC (2008) Dominant culture and the education of women. Cambridge Scholars Publishing. Newcastle upon Tyne, UK, ISBN: 9781847185730
13. Piotrowicz W, Cuthbertson R (2009) Sustainability—a new dimension in information systems evaluation. J Enterprise Inform Manag 22(5):492–503
14. Rawls J (1999) A theory of justice. Belknap Press, Boston, USA
15. Rogers CR, Freiberg HJ (1994) Freedom to learn: a view of what education might become. Prentice-Hall, New York. ISBN 9780024031211
16. Schilthuis W (2003) Biodynamic agriculture. Floris Books, Edinburgh, UK. ISBN 9780863153976
17. Smit G (2011) Grip strategy in dealing with vulnerable ("Street") children and their families. http://www.newlifeprojects.co.za/strategy.html
18. Steyn J et al (eds) (2010) ICTs and sustainable solutions for the digital divide: vol. 1: theoretical issues and vol. 2: practical approaches. IGI-Global, Hershey, PA, IBSN: 9781615209972
19. Stillman L, Herselman M, Marais M, Pitse-Boshomane MM, Plantinga P (2010) Digital doorway: social-technical innovation for high-needs communities. Proceedings of the 4th international IDIA development informatics conference, Cape Town, 3–5 Nov 2010. IDIA, Monash University, Johannesburg, South Africa, ISBN: 9780620475907
20. Sutinen E, Tedre M (2010) ICT4D: a computer science perspective. In: Elomaa T, Mannila H, Orponen P (eds) Algorithms and applications, lecture notes in computer science (LNCS) 6060. Springer, Berlin, Heidelberg, pp 221–231. ISBN 9783642124754
21. UNESCO (2000) The Dakar framework for action—education for all: meeting our collective commitments. World Education Forum, UNESCO, Paris. http://unesdoc.unesco.org/images/0012/001211/121147e.pdf
22. United Nations (2008) Organic agriculture and food security in Africa. United Nations Environmental Programme (UNEP) and United Nations Conference on Trade and Development (UNCTAD) 2008. Report. United Nations, New York, Geneva

23. United Nations (2010) The millennium development goals report 2010. UN Department of Economic and Social Affairs, United Nations, New York. ISBN 9789211012187
24. Unwin T (ed) (2009) ICT4D, information and communication technology for development. Cambridge University Press, Cambridge, UK. ISBN 9780521712361
25. Verenikina I, Vialli W, Lysagt P (2011) Understanding, learning and development. David Barlow Publishing, Macksville, NSW, Australia. ISBN 9781921333361
26. Watson T, Boudreau M-C (2010) Information systems and environmentally sustainable development: energy informatics and new directions for the IS community. MIS Quart 34(1):23–38
27. Yunus M (2003) Banker to the poor: micro-lending and the battle against world poverty. Public Affairs, New York. ISBN 9781586481988

Towards Improving Information Systems Interoperability in Disaster Management

Ovidiu Noran

Abstract Climate change-triggered catastrophic events appear to be steadily increasing in intensity and frequency. Proper preparation, response and recovery are essential in order to survive and recover from disasters. However, for historical, geographical and cultural reasons, the organisations responsible for delivering emergency response services often underperform, with the lack of proper interoperation and collaboration of their information systems being a main culprit. This paper analyses interoperability issues specific to disaster management from an information systems perspective and proposes improvements based on advances in information systems and interoperability research, using an enterprise architecture perspective in order to provide a sustainable holistic and life cycle-based solution.

1 Introduction

History has shown that business survival and success requires a multipronged approach towards sustainability, reflecting Elkington's [1] Triple Bottom Line: one must achieve not only economic bottom-line performance but also environmental and social accomplishment. Blackburn [2] compares economic sustainability to air and environmental and social sustainability to food: the first is more urgent but not more important than the second—hence, a successful enterprise must take a whole-system approach to sustainable development [3, 4].

Although emphasis on environmental sustainability is increasing worldwide, the results will take significant time to impact on and hopefully revert the current bleak environmental trends. Climate change is nowadays seen as real and the main cause of an increasing number of disasters of growing intensity. In this context, it is

O. Noran (✉)
School of ICT, Griffith University, Nathan, QLD, Australia
e-mail: o.noran@griffith.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_27,

essential to be able to promptly and effectively prevent, prepare for, respond to and recover from catastrophic events. Governments typically respond by creating policies and agencies/organisations to tackle these aspects. The "disaster management organisations" (DMOs) thus created operate in a complex environment (history/traditions, geographic location and culture, level of governance, etc.) that often promotes heterogeneity and independent evolution. Organisational diversity, while potentially beneficial, requires additional effort to achieve proper and effective collaboration [5]. In addition, coping with large scale catastrophic events usually requires resources and capabilities beyond those of individual organisations, making effective cooperation and interoperation of DMOs essential. Failure to meet these requirements at all necessary levels and addressing all relevant aspects has dire consequences that typically include loss of property and even human life.

As an essential and ubiquitous component of every organisation [6], the information system (IS) plays an essential role in achieving cooperation and interoperability and thus organisational sustainability of the disaster management effort. This paper aims to focus on the IS aspect of disaster management interoperability and to propose ways to address them using knowledge accumulated in the IS and interoperability research areas. The analysis is performed from an enterprise architecture (EA) stance in an attempt to promote a whole-system and life cycle-based approach covering all aspects deemed as relevant for the disaster management universe of discourse.

2 Current Issues in Disaster Management

The operation of emergency services is typically legislated at state, national and international levels (see [7–10]). However, simply instructing organisations to cooperate using high-level generic directives does not bring about true collaboration and/or interoperability. The consequences are increased response times, confusion about the situation on the ground and dispute as to who, where and when is in charge. It is also difficult to coordinate with other teams' ISs due to incompatibilities in infrastructure and difficulty in filtering and validating the typical flood of information generated during disaster events. For example, inconsistency in alert notices' type and format will delay intervention and hinder warnings by fuelling a "cry wolf" situation where the population is saturated with ambiguous and/or irrelevant messages [11, 12]. This may lead to suboptimal prevention and response by intended recipients and potential property and life loss. Efforts to standardise warning message protocols are currently rather localised with low take-up rates [13].

Two main approaches are used to address the above-mentioned problems; they involve either centralisation (hierarchical command) or federalisation of emergency services. Irrespective of the approach used however, proper emergency response and cooperation has still not been achieved, as reflected in criticism expressed in various literature (e.g. [14–16]). The use of military operations-style network-enabled capabilities as the backbone of disaster management [17] is valid only as part of the overall disaster management effort and could also promote overreliance on an infrastructure that may well fail during disaster events.

Various documents, inquiries, reviews and reports ([10–12, 15, 16, 18, 19], etc.) suggest that the root causes of current shortcomings could in fact be the inadequate information and knowledge flow/quality between the participants' systems [20, 21] owing mostly to incompatibilities originating in heterogeneity, lack of trust, organisational confusion and competition fallacies. True and efficient collaboration requires the interoperability of processes, resources and organisational cultures of the participants [22, 23], all of which are reflected in their ISs [24]. Another aspect addressed to a lesser extent (albeit being an essential factor in the interoperability requirements) appears to be the life cycles of the DMOs, task forces, government agencies, legislation, service providers and disaster event(s).

Analogies with other domains help provide potential solutions. For example, the DMOs' situation resembles that of commercial enterprises that need to cope with a global business environment requiring them to tackle projects beyond their own resources and knowledge. A typical response in this case is for the companies to set up or join the so-called collaborative networks [25]. Another analogy is that of allied armed forces that prepare to cooperate in crisis situations by employing standardised agreements (to the extent possible) and joint exercises [26].

DMOs may have full time/militarised and voluntary/reserve staff components depending on the geographical location and local legal and administrative situation. Thus, concepts from both commercial and military areas can be used, provided proper customisation is performed so as to fit the specific scenario.

3 Disaster Management Interoperability Approach and Aspects

Tackling DMO interoperability requires some clarification: What is the required interoperability extent? What components and/or aspects need to interoperate? How can we ensure that all *necessary* aspects are covered and especially that interoperability is preserved over time as all participants evolve? Each disaster event is unique; thus, there is no "one size fits all" (crisis situations) DMO interoperability level. At a minimum, the participating organisations' systems should be *compatible*, so at least they don't hinder each other's operations (see Fig. 1).

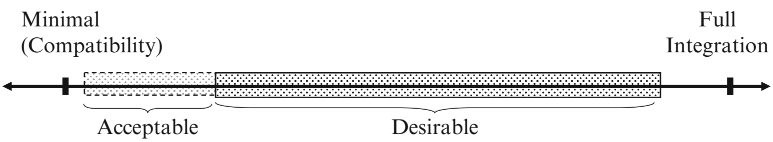


Fig. 1 Acceptable and desirable IS interoperability levels in disaster management

Full integration (if at all achievable) is not desired in this case as it would imply that the DMOs cannot fully function independently. In an emergency situation, some task force members' (information and other) systems may be affected and even cease to function. The other participant DMOs should be able to continue without significant performance loss (see e.g. the ARPANET resilient network concept [27]) and ideally even "cover" for the participant(s) that failed or is ailing.

Even if a central point of command (e.g. an "Emergency Command Centre") was secure and unaffected by the disaster event(s), the coordination provided by it could be severely impeded by communication breakdown. The remaining task force participants should be able to carry on their duties, at least for a predetermined amount of time. This quasi-independence can be achieved in the form of pre-agreed procedures and shared knowledge, set up well in advance and then continuously monitored and maintained for relevance and consistency.

Referring to the interoperability body of knowledge, ISO14258 [28] establishes several ways to achieve (IS and general) interoperability: integrated (common format for all models), unified (common format at *meta* level) and federated (participants negotiating an ontology "on the fly" so as the meaning of models is understood the same way). In the case of DMOs, full integration and federalisation did not seem to succeed and achieve the desired results due to organisational heterogeneity of DMOs and the impossibility to properly negotiate in the limited time available in the case of a disaster event. The seemingly more suitable unified approach assumes that ontology is negotiated in advance; although there have been numerous ontology integration efforts and advances, in disaster management currently there seems to be no effective substitute to DMOs "spending time together" in order to agree on meanings associated with the concepts used to exchange knowledge. Once that is achieved, proper semantic interoperability should be faster and easier to achieve in the task forces formed by the participant DMOs.

Of all the interoperability aspects to be considered in disaster management, data and process areas appear to be the most *urgent* (see Fig. 3, right). Thus, the ability to extract and exchange data from heterogeneous sources (delivering high volume and often unreliable data during disaster events) is paramount to being aware of the conditions on the ground and avoiding potentially life-threatening situations for emergency crews and population. Prior agreements on data format and especially on its *meaning* are essential. Process interoperability here concerns the capability to perform joint operations but also to "take over" and perform processes on behalf of a disaster management task force participant that may have been temporarily or permanently disabled.

The *pragmatic* interoperability aspect as described by Whitman and Panetto [5] relates to the willingness and capacity of the participants to interoperate and suggests once again that the human component of the IS needs attention *prior* to task force formation as to allow gaining trust and knowledge of the other participants.

Organisational interoperability is an important aspect in disaster management as task force participants may often exhibit significant organisational structure diversity. The issues identified by Chen [29] based on the Enterprise Interoperability Framework (EIF), namely, responsibility, authority and type of organisation, can all impact heavily on the functionality of the disaster management task force. In a crisis situation, the roles (mapping of the human resources onto the decisional structure) and hierarchy must be clear to everyone from the start so that the task force can focus on managing the disaster event rather than spend critical time figuring out its own modus operandi (who does what, who is in charge, etc.).

Finally, cultural interoperability [5] appears to be one of the hardest obstacles to overcome. The only current working solution appears to be regular immersion of the participant organisations in each other's cultures (e.g. army joint exercises and the expatriate experience of most proficient language translators'). This facilitates the transfer and conversion of tacit and explicit knowledge between the participants, which should be one of the most important goals of interoperability.

The leitmotif running through all aspects analysed above is the "co-habitation" of the organisations that are expected to form disaster management task forces, as a prerequisite towards the achievement of IS interoperability—whether functional, informational, organisational or cultural. Therefore, this paper proposes the adaptation (in a life cycle context) of the commercial-based collaborative network and associated concepts to assist IS interoperability for disaster management.

4 Collaborative Networks for Disaster Management

Collaborative networks (CNs) are created in order to act as "breeding environments" (BEs) for virtual organisations (VOs). A VO is a group of companies that act as one for a limited time in order to bid for, win and complete projects requiring combined resources and know-how; at the end of the project(s), the VO usually dissolves. BE partners may take part in one or several VOs at any given time. A CN "lead partner" may also be present—elected on size, influence, resources, etc.

The CN concept requires some customisation to make it suitable for the disaster management area. Thus, the time available for set-up of a VO (more suitably hereafter called disaster management task force—DMTF) is significantly shorter than that available for a project bidding process. In addition, the "Disaster Management" Collaborative Network (DMCN), its participants and the DMTF(s) produced will operate under tighter, legislated operational guidelines set by the relevant governments and national/international frameworks.

The commercial and competitive motivations of the typical CN participants will translate in this case into an efficiency and interoperability motivation reflected in lives and property rescued. The usual create/join/remain/leave the CN decision processes would be mandated from outside (or by the lead partner) for most participants. Reference models (such as patterns) are to be created from lessons learned in past disasters prevention/relief efforts and used to refine future DMTFs. The reference models should be stored in a web-enabled, shared repository and classified on type and location (e.g. flooding, tornadoes, wildfires, severe storms in Australia [30]) and customised to allow for specific intensity, duration, side effects and consequences.

The DMTF(s) produced by the DMCN will not bid for projects but rather be set up for and assigned a specific one—the management of a particular disaster event (or combination thereof). The IS management, communication infrastructure and other organisational interoperability issues would have been sorted out in advance within the DMCN, ensuring a prompt and appropriate task force response and thus addressing a frequent weakness of past crisis management efforts.

Human-related aspects requiring time such as trust and local/organisational culture (including recognition which features prominently in volunteer-based organisations [31, 32]) could be also tackled using the concept of an "emergency services academy" with local branches providing training based on a unique national/interstate curriculum (see [33] for an early initiative).

5 The Life Cycle Context and Enterprise Architecture

All entities, including those involved in a disaster management effort, have gone (and will also go) through a series of life cycle phases, possibly several times (e.g. during re-engineering). According to [39], these phases form the entities' *life histories*. IS interoperability requirements are inherently linked to life history as they will vary during each IS and business life cycle phase; therefore, it is essential that the analysis and search for interoperability improvements is performed *in a life cycle context*. It is hereby argued that an optimal way to integrate the life cycle aspect into IS interoperability is by using an "enterprise architecture" (EA) approach.

While there are many definitions for EA, the author takes the commonly accepted view that it expresses *the ontology of enterprise change*. Thus, EA provides a holistic approach to business evolution and agility "by creating, communicating and improving the key requirements, principles and models that describe the enterprise's future state" [34]. Importantly, "[…] EA comprises people, processes, information and technology of the enterprise, and their relationships to one another and to the external environment" [Ibid.]. This EA definition is in agreement with the view of IS as a socio-technical system [35] with voluntaristic people [38] in a complex organisational, political and behavioural context [36, 37]. As such, it encourages the use of EA as an integrated, multi-perspective approach for analysing the impact of the interactions between the disaster management participants on their individual and task force IS interoperability.

To illustrate the way EA artefacts and methods can guide and enrich the analysis and improvement of DMO interoperability, we have selected a generic architecture framework (AF) subsuming and abstracting several other mainstream AFs that is defined in Annex A of ISO15704 [39] and called Generalised Enterprise Architecture and Methodology (GERAM). The modelling framework (MF) of GERAM's reference architecture (GERA) contains a rich repository of aspects (including human) that can all be represented in a life cycle context. GERA's MF has been previously used in IS management modelling and other tasks (e.g. see [40, 41]).

Subsets of the GERA MF can be used to build life cycle-based formalisms such as shown in Fig. 2, top. These can then be utilised in the creation of business models

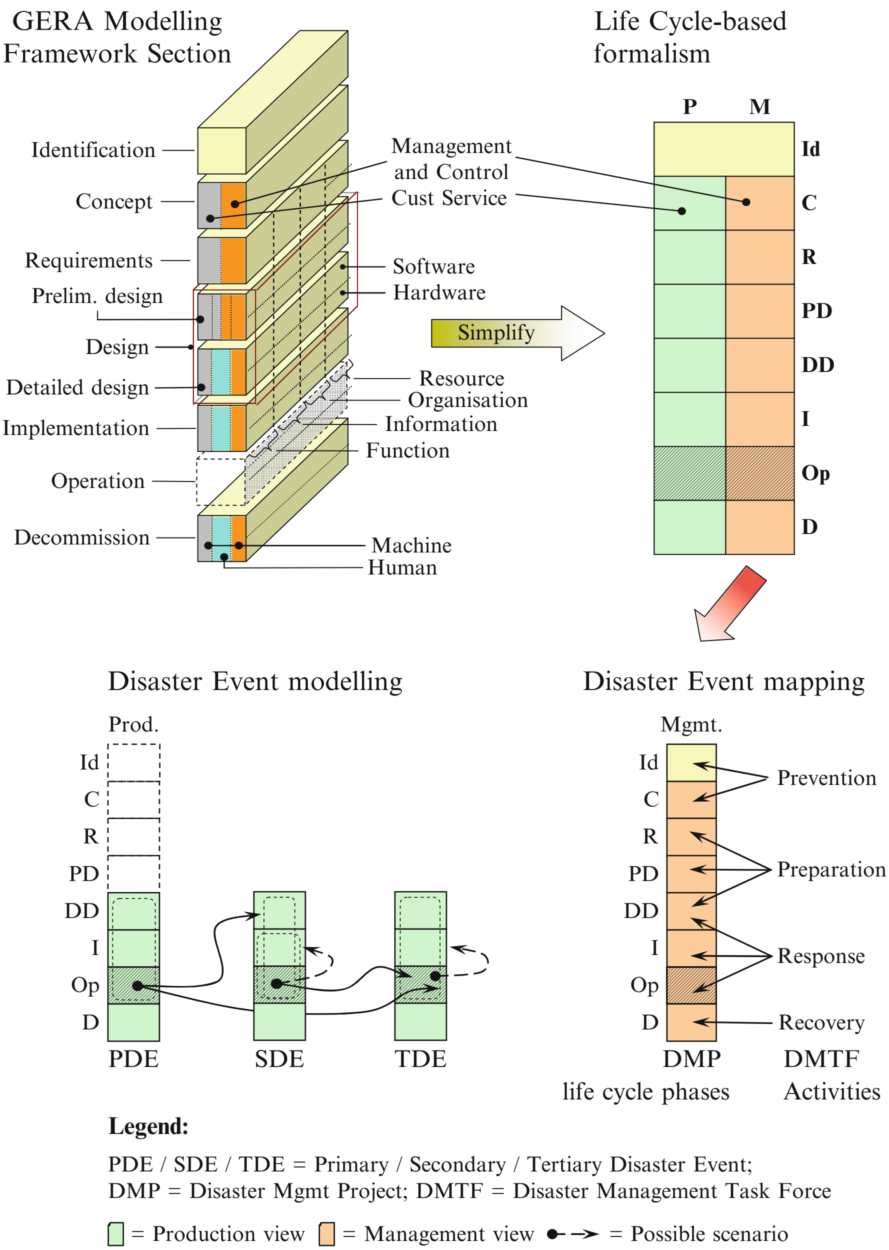


Fig. 2 Disaster event modelling and mapping using the GERA life cycle formalism

requiring a life cycle background. Thus, aspects previously identified as significant in tackling IS disaster management interoperability (function, information, organisation, human aspect) can be represented together as shown in Fig. 2 upper left. Aspects can also be separated (e.g. for clarity) using a “flattened” representation;

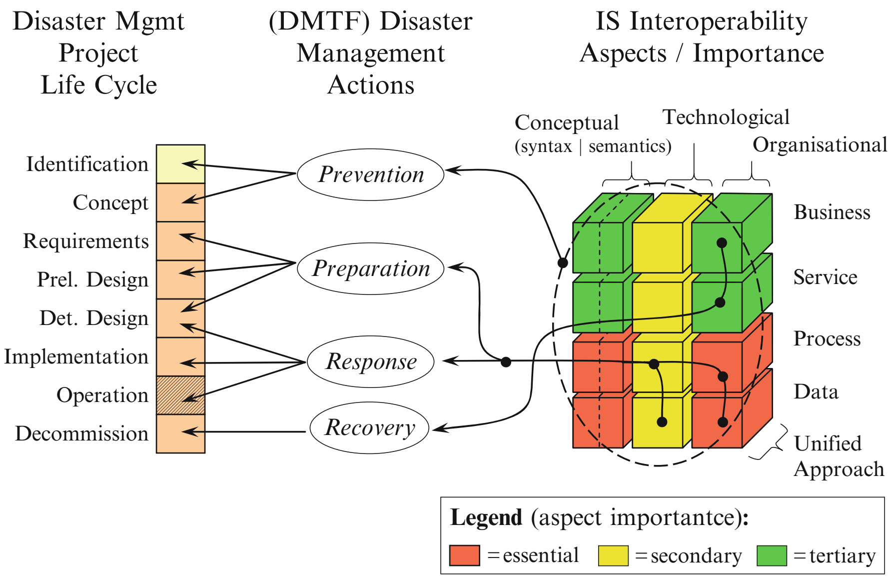


Fig. 3 IS interoperability requirements mapped on disaster management actions and the disaster management project life cycle, using Chen's EIF- [29] and GERA's MF-based formalism

for example, the 2-dimensional structure shown in Fig. 2 upper right can be used to separate diagrams for information, function and so on.

Figure 2 lower left uses the GERA-based modelling formalism to show a high-level representation of a disaster event, taking into account its relevant life cycle phases and possible relations to other events. Thus, it can be conveyed, for example, that a primary disaster event (PDE) can trigger/influence other events (SDE, TDE). For example, an earthquake event (PDE) can trigger a tsunami (SDE) that can in turn trigger a partial nuclear meltdown (TDE). However, PDE can also influence TDE directly. Events such as chain reactions can also be shown (arrows from operation to implementation within same entity). Here, the GERA MF's "Detailed Design" and "Implementation" life cycle phases refer to features of the event—e.g. earthquake time, epicentre and duration or tsunami spread, wave speed, height.

The lower right hand side of Fig. 2 shows how the life cycle phases of a disaster management project (DMP) can be mapped to the typical disaster management activities [8] performed by the DMTF that sets up and operates that project. Such diagrams are useful to help stakeholders grasp a common understanding of the disaster event and management aspects, thus overcoming some IS interoperability semantic barriers that may delay DMP kick-off and thus hinder DMTF response.

Figure 3 uses a modified version of Chen's EIF (used in Sect. 3 to identify types, aspects and barriers to interoperability) and GERA's MF-based life cycle modelling formalism to show how the relevance and applicability of the IS interoperability aspects depend on the specific life cycle phases of the disaster event and the DMTFs' actions to address them. The figure shows that different IS interoperability aspects take precedence during various disaster event life cycles.

For example, during the response phase, data and process IS interoperability aspects are paramount as accurate, and fresh data is required, and processes may need to be performed interchangeably due to potential disability of DMTF participants (or need to involve replacement DMCN members). During the recovery, which typically takes place over an extended period of time and requires less real-time responsiveness, business and service IS interoperability aspects take precedence.

6 Application to IS Data Interoperability: A Possible Scenario

Although very helpful in setting up the task force and DM project, Fig. 3 shows only the influence of the disaster event, management project and task force life cycles on the interoperability requirements. As a matter of fact, the DMOs, DMTF and DMP interact with a plethora of entities and artefacts during their life history. These interactions must also to be analysed in order to gain a whole-system view of the IS interoperability endeavour.

Using the GERA-based formalism shown in Fig. 2, several diagrams may be built for the aspects identified in Sect. 3 as important to IS interoperability. For example, Fig. 4 shows interoperability-related interactions for the information

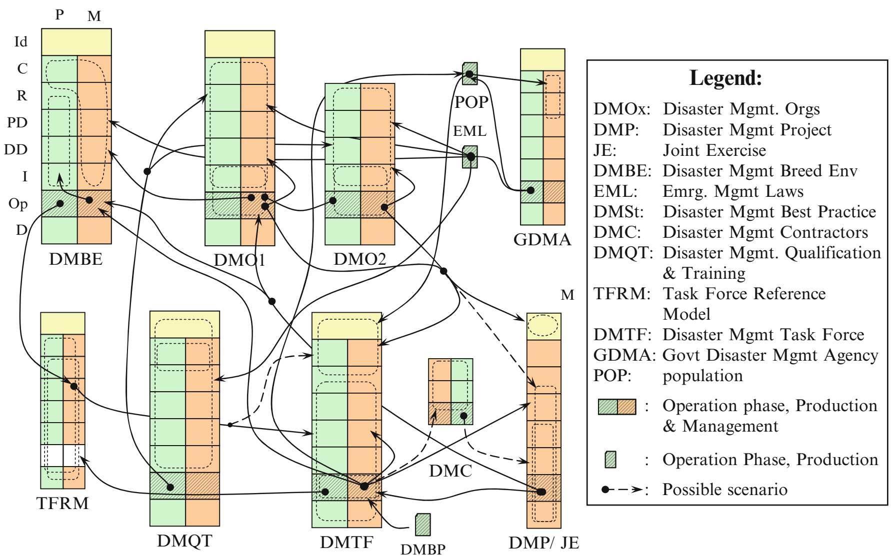


Fig. 4 High-level IS data interoperability requirements in the context of a Disaster Management Breeding Environment and Task Force solution (details omitted for clarity)

(data) aspect of the GERA MF, in the previously described scenario of a breeding environment (DMBE) creating task forces (DMTFs) supervising disaster management projects (DMPs) that coordinate disaster event responses.

The arrows in the figure show data interoperability requirements and influences. For example, the participating DMOs' ISs need to interoperate during their operation life cycle phase; however, this may also imply some redesign (line between DMO1 and DMO2 linking operation but also "upper" life cycle phases). The IS of the DMBE must be able to interoperate with the IS of the DMTF it creates and DMPs managed by them in their operation life cycle phases, with influences on other phases in case of subsequent redesign (details omitted for clarity).

In another example, data "interoperation" of the population POP with the government's disaster management agencies (GDMAs, such as [42]) is important as it may result in changes to legislation—hence in the DMO and DMBE interoperability requirements. However, data interoperability between the ISs of DMOs participating in a DMTF and between the DMTF's IS and POP is paramount as it will directly influence the extent of lost property and casualties. Past (and sometimes tragic) experience and feedback from subsequent enquiries [11, 12, 30] has shown that two major goals of the disaster management interoperability improvement effort (partly reflected in Fig. 4) should be a) whether POP receives, understands, believes and *acts* on DMTF warnings and directives and b) that DMTF participants can properly interoperate during the disaster event. Thus, the IS of the DMTFs (and implicitly DMOs) should be also designed to avoid ambiguity and maximise focus in relation to the local specific semantic interoperability requirements (language, technology type, habits, etc.). For example, the message format and distribution in a densely populated and developed area would most likely differ from the one used in a sparsely and/or underdeveloped region—at least until efforts to standardise warnings are successful, widely implemented and observed.

Importantly, some organisations shown in Fig. 4 should be able to redesign themselves to a certain extent (arrows from operation life cycle phase to upper phases within same entities e.g. in DMBE, DMO, DMTF). This reflects an essential capability to adapt (and thus be *agile* and resilient) in the face of changes in the situation and environment that are likely to occur briskly and unexpectedly during disaster events.

The Disaster Management Qualification and Training organisation (DMQT, see e.g. [33]) can assess the suitability of organisations to enter a DMBE by requiring (and providing training if necessary towards) IS data interoperability between DMO applicants and the DMBE. This requirement may also go beyond the operation life cycle phase should training/redesign of the participants be performed.

The validity and effectiveness of the DMBE concept can be tested by joint exercises (JEs) simulating disaster events; this will also allow practical and potentially hidden IS data interoperability problems to surface and be tackled. Importantly, the resilience of DMTFs created can be also assessed in JEs through various scenarios and proper corrective action be taken to improve DMOs' preparedness.

7 Conclusions and Further Work

Disaster management interoperability-related issues are best addressed in advance and in a holistic manner so that when a disaster event occurs an efficient, collaborative task force can be promptly put together. Therefore, in order to tackle the IS interoperability aspects relevant to disaster management identified and prioritised using state-of the-art research in the domain, it is proposed to adopt the commercial breeding environment and virtual enterprise concepts, customised for disaster management. In addition, as the life cycle of the disaster management participants plays a central role in the IS interoperability requirements, it is argued that (and shown how) an EA approach can provide an integrated, life cycle context to the entire disaster management IS interoperability improvement effort.

Further research will concentrate on testing, verifying and validating the findings with DMO stakeholders within several past, present and future case studies.

References

1. Elkington, J. (1998). Partnerships from cannibals with forks: The triple bottom line of 21st-century business. Environmental Quality Management, 8(1):37–51
2. Blackburn WR (2007) The sustainability handbook. EarthScan Publishers, Cornwall, UK
3. UN World Commission on Environment and Development (1987) Our common future (Brundtland Report). Oxford University Press, Oxford
4. Noran O (2011) Integrating environmental and information systems management: an enterprise architecture approach, in information systems development: Asian experiences. In: Song W et al. (eds) Proceedings of the 18th international conference on information systems development (ISD 2009), Springer, Nanchang, China, pp 123–134
5. Whitman L, Panetto H (2006) The missing link: culture and language barriers to interoperability. Annu Rev Contr 30(2):233–241
6. Lehtinen E, Lyytinen K (1986) Action based model of information system. Inform Syst Res 11(4):299–317
7. Government of South Australia (2004) Emergency Management Act 2004. http://www.legislation.sa.gov.au/LZ/C/A/EMERGENCYMANAGEMENTACT2004.aspx. Accessed 30 Mar 2011
8. Australian Government (2011) Attorney's General's Office—Emergency Management in Australia. http://www.ema.gov.au/. Accessed 30 Mar 2011
9. Federal Emergency Management Agency (2011) National response framework. http://www.fema.gov/pdf/emergency/nrf/about_nrf.pdf. Accessed 30 Mar 2011
10. United Nations International Strategy for Disaster Reduction Secretariat (UNISDR) (2011) Hyogo Framework for Action 2005–2015: building the resilience of nations and communities to disasters. http://www.preventionweb.net/files/1037_hyogoframeworkforactionenglish.pdf. Accessed 30 Mar 2011
11. Queensland Floods Commission of Enquiry (2011) Submissions to the enquiry. http://www.floodcommission.qld.gov.au/submissions/. Accessed Jul 2011
12. Victorian Bushfires Royal Commission (2009) Submissions to the enquiry. http://www.royalcommission.vic.gov.au/Submissions/View-Submissions. Accessed Jul 2011
13. OASIS (2005) Common Alerting Protocol v1.1. http://www.oasis-open.org/committees/download.php/15135/emergency-CAPv1.1-Corrected_DOM.pdf. Accessed 2012
14. Clark JL (2006) Practical aspects of federalizing disaster response. Crit Care 10:107–113

15. Wiese CR (2006) Organizing Homeland Security after Katrina: is adaptive management what's missing? Publ Admin Rev 66(3):302–318
16. Hart P et al (2005) The politics of crisis management: public leadership under pressure. J Homeland Security Emerg Manag. Cambridge University Press, Cambridge, UK
17. von Lubitz DKJE, Beakley JE, Patricelli F (2008) Disaster management: the structure, function, and significance of network-centric operations. J Homeland Security Emerg Manag 5(1):Art. 42
18. Brewin B (2011) Tsunami response reveals poor radio interoperability. http://www.nextgov.com/nextgov/ng_20110415_3972.php. Accessed Apr 2012
19. Igarashi Y et al (2011) Anatomy of historical tsunamis: lessons learned for tsunami warning. Pure Appl Geophys 168:2043–2063
20. Wickramasinghe N, von Lubitz DKJE (2007) Knowledge based enterprises: theories and fundamentals. J Homeland Security Emerg Manag. Vol. 5. IGP Publishing Group, Hershey PA
21. Prizzia R, Helfand G (2001) Emergency preparedness and disaster management in Hawaii. Disast Prev Manag 10(3):163–172
22. Kapucu N, Arslan T, Demiroz F (2010) Collaborative emergency management and national emergency management network. Disast Prev Manag 19(4):452–468
23. Trakas A (2012) Interoperability—a key requirement for emergency and disaster management. http://www.un-spider.org/book/5143/4c-challenge-communication-coordination-cooperation-capacity-development. Accessed 2012
24. Stohr E, Konsynsky BR (1992) Information systems and decision processes. IEEE Computer Society Press, California
25. Camarinha-Matos L et al (2009) Collaborative networked organizations—concepts and practice in manufacturing enterprises. Comput Ind Eng 57(1):46–60
26. NATO (2006) Interoperability for joint operations. http://www.nato.int/docu/interoperability/interoperability.pdf. Accessed Jul 2011
27. Heart F et al (1978) In: Bolt, Beranek, Newman (eds) ARPANET Completion Report. Burlington, MA
28. ISO (2005) ISO14258 Industrial automation systems—concepts and rules for enterprise models
29. Chen D (2006) Framework for enterprise interoperability. http://www.fines-cluster.eu/fines/jm/Download-document/53-Framework-for-Enterprise-Interoperability-Chen.html. Accessed Jul 2011
30. Ellis S, Kanowski P, Whelan R (2004) National inquiry into bushfire mitigation and management. Commonwealth of Australia, Canberra
31. Esmond J (2011) Report on the attraction, support and retention of emergency management volunteers. http://www.ema.gov.au/www/emaweb/emaweb.nsf/Page/Volunteers2. Accessed 30 Mar 2011
32. McLennan J (2008) Issues facing Australian volunteer-based emergency services organisations: 2008–2010. http://www.ema.gov.au/www/emaweb/emaweb.nsf/Page/Volunteers2. Accessed 30 Mar 2011
33. Queensland Government (2011) Queensland combined emergency services academy (QCESA). http://www.safecom.sa.gov.au/site/emergency_management.jsp. Accessed 30 Mar 2011
34. Gartner Research (2012) IT glossary. http://www.gartner.com/technology/it-glossary/enterprise-architecture.jsp. Accessed 2012
35. Pava C (1983) Managing new office technology, an organisational strategy. Free Press, New York
36. Keen PGW, Scott Morton M (1978) Decision support systems: an organisational perspective. Addison-Wesley, Reading, MA
37. Iivari J (1991) A paradigmatic analysis of contemporary schools of IS development. Eur J Inform Syst 1(4):249–272
38. McGregor D (1960) The human side of enterprise. McGraw-Hill, New York

39. ISO/IEC (2005) Annex C: GERAM, in ISO/IS 15704:2000/Amd1:2005: industrial automation systems—requirements for enterprise-reference architectures and methodologies
40. Noran O (2008) A meta-methodology for collaborative networked organisations: creating directly applicable methods for enterprise engineering projects. VDM Verlag Dr. Müller, Saarbrücken
41. Noran O (2009) In: Doucet G, Gotze J, Saha P (eds) Engineering the sustainable business: an enterprise architecture approach, in coherency management: architecting the enterprise for alignment, agility, and assurance. International Enterprise Architecture Institute. pp 179–210
42. Government of Queensland (2012) Department of community safety—structure. http://www.emergency.qld.gov.au/about/. Accessed Apr 2012

Enterprise Architecture Modelling for Corporate Sustainability

Malgorzata Pankowska

Abstract The chapter aims to validate the usability of enterprise architecture (EA) development methodologies to support enterprise sustainable development. The main thesis of the chapter is that EA modelling supports the sustainable development of business organisation as well as the sustainable governance of IT in the organisation. Therefore, the first part of the chapter covers discussion on what sustainability is, particularly in the context of EA development. The second part comprises analysis of EA modelling methods and explanations of their usefulness for sustainable development. The third part includes presentation of the EA development principles as important for corporate sustainability.

1 Introduction

In socio-economic literature, sustainable development is the practice of meeting the needs of society today without compromising the ability of future generations to meet their own needs [1]. The term in its environmental usage refers to the potential longevity of vital human ecological support systems, such as the climatic systems, systems of agriculture, industry, forestry and fisheries and the various systems, on which they depend in balance with the impacts of their unsustainable or sustainable design. In each information communication technology (ICT) project, project sponsors expect results for prolonged periods of time and expect answer to the questions of what results will be available in the future, how project beneficiaries want to ensure project results sustainability and what project results arrive after the project financial support finishes. Sustainable development is to be proactive about change.

M. Pankowska (✉)
University of Economics, Katowice, Poland
e-mail: pank@ekonom.ae.katowice.pl

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_28,

The move towards the enterprise engineering is usually strongly based on economic considerations, i.e. development and maintenance costs reduction, decrease of human efforts and power energy usage and improvement of the quality of the resulting EA components. The EA engineering approaches are to support the integrity, reliability, effectiveness, efficiency and actuality of the EA components as well as large-scale reuse during the systems' development.

2 Sustainability Development

Sustainability is a characteristic of a process or state than can be maintained at a certain organisational level. It is understood as the simultaneous effort of balancing economic, social and environmental goals for a corporation [2]. So, sustainability is a metaphor for describing corporate social responsibility, corporate citizenship or ethical business conduct.

In 1987, the World Commission on Environment and Development (WCED) related sustainability to corporations and the economy by defining the sustainable development as development that meets the needs of the present without compromising the ability of future generations to meet their own needs [3, 4]. Sustainable development is proposed as a strategy to improve the quality of human life.

The etymology of "sustainable" carries interesting and important implications for the way the word is used as it includes several contradictions. The word "sustain" is derived from the Latina "sub-tenere", meaning "to uphold". This carries as passive connotation in it and gives the image of stability, persistence and balance. "Sustainable" is used in a more active sense together with "development". Development means change, progress and growth. Hence, "sustainable development" means a progress which is being uphold or defended at the same time as it implies movement and improvement [5]. The idea of "sustainable development" includes a normative and active meaning. In this sense, the sustainable development assumes a certain equilibrium among the strategic factors and available resources.

Shrivastava notices the incentives for organisations to support ecologically sustainable development activities include the following: the decrease of operating costs, creating a competitive advantage with green ICT, establishing a legitimate sustainability present with the public and stock markets, reducing the long-term risks associated with resource depletion, improving the ecosystems and community environment and positioning their organisations ahead of the regulations [6].

Corporate governance for sustainability is a setting of rules, their application and enforcement [7]. Proponents of economic liberalism prefer market mechanisms domination and they do not see the state playing a major role in supporting sustainability. Models of governance for sustainability need to concentrate more on change than stability. The existing rules, customs, practices and rights are seen more as the subject to be influenced, than as the main business of governance [8].

In this chapter, the sustainability is considered in three aspects. In economic aspect, the sustainability means long-term growth, ICT resources availability, efficiency and minimisation of human efforts and decrease of consumption of electricity. The economic dimension of sustainability concerns the organisation's impacts on the economic conditions of its stakeholders and on economic systems at local, national and global levels. The economic indicators cover flow of capital among different stakeholders and main economic impacts of the organisation throughout society. The impact indicators comprise economic performance measures, market presence and indirect economic impact indicators.

From the environmental point of view, the sustainability means maintaining lower energy human ecosystems. The environmental dimension of sustainability concerns an organisation's impact on living and nonliving natural systems. They cover performance related to biodiversity, environmental compliance and other relevant information such as environmental expenditure and the impacts of products and services.

In the social and cultural aspect, sustainability is identified with the sociocultural diversity, including multi-stakeholder opinions, justice, equity respect and transparency of ICT systems. The social dimension of sustainability concerns the impacts an organisation has on the social systems within which it operates. Social sustainability occurs when the formal and informal processes, systems, structures and relationships actively support the capacity of current and future generations to create healthy communities [9].

Sustainability of an ICT activity is strongly influenced by the technology used. Despite the current emphasis on economic sustainability, it is too difficult to determine sustainability criteria for ICT projects. Although many such projects use cost-recovery mechanisms, mostly they declare that project results will be exploited for the established project duration period according to the predetermined plans. Sustainability in the context of ICT is addressed through the Green IT approach. However, ICT specialists have got a certain dilemma. On one side, they create an environment of IT that is sustainable in itself (sustainable ICT), and on the other side, they foster the sustainability throughout the organisation (sustainable through ICT) that requires the integration, stability and communication inside and outside an organisation [10, 11].

In this chapter, sustainability is perceived to provide added value to the entire information system life cycle by establishing architecture to enable its implementation and rules for ownership and usage, resulting in benefits to the end user. Particularly, the enterprise architecture has abilities to ensure transparency, credibility, comprehensiveness and consistency for corporate sustainability. Transparency is enabled in the processes of the EA modelling and execution, based on scientific methods or international conventions and standards. Credibility is verified by the end users and stakeholders of the EA development. The enterprise architecture is communicated to the entire company. The focus of the communication is rather on new mutual insights and action orientation. Comprehensiveness refers to the consideration of all aspects i.e. economic, social and environmental, as well as to the identification of potential benefits. Only the enterprise architecture is able to ensure the comprehensive view, which is so important for the corporate sustainability. Consistency concerns the harmony among all parts of the process.

3 Enterprise Architecture Modelling

The main aim of the modelling exercise is to go beyond the concrete description and to fetch the abstract view of a modelled object. Modelling refers to systematic activities undertaken to describe and visualise abstract phenomena in a structured or formal way. The enterprise modelling is to describe enterprise objectives, activities, information resources, processes, actors, products, requirements and the relationships between those entities [12].

Although the enterprise architecture development is expected to convey semantic unification, nowadays the enterprise modelling approaches do not offer mutually agreed languages. Therefore, the lack of unified approaches prevents from providing meaningful information outside the scope of the enterprise. Unfortunately, every EA framework establishes its own techniques, artefacts, schemes and vocabularies. Companies have to customise and adapt recognised EA frameworks to meet their requirements, so perhaps they have no opportunities to reduce costs, efforts, and work time. Eventually, there is lack of transparency and credibility of EA.

In American National Standards Institute and Institute of Electrical and Electronics Engineers (ANSI/IEEE) Standard 1471–2000, an architecture is the fundamental organisation of a system, embodied in its components, their relationships to each other and the environment and the principles governing its design and evolution. The architectural framework is a specification of how to organise and present architectural models. Because the EA is an all-encompassing discipline, and because the enterprises it describes are often large, it can result in very complex models. The extension of the conceptual framework of IEEE 1471 is presented in Fig. 1.

The IEEE 1471 definition explains that architecture is described by one or more architecture descriptions, which are composed of one or more stakeholders, viewpoints, views and models. The architectural framework is a conceptual structure related to a certain system type and consisting of areas of concern and a necessary and sufficient set of design domains. In the extension of the ANSI/IEEE definition, the business strategy and sustainability are included. A strategy is perceived as something an organisation needs or uses in order to win or establish its legitimacy in a world of rivalry [13, 14]. Strategy is both a plan for the future and a pattern from the past; it is the match an organisation makes between its internal resources and skills and the opportunities and risks created by its external environment [15–17].

The historian Alfred Chandler has formulated the thesis that structure follows strategy [18]. He described strategy as the determination of long-term goals, the adoption of courses of action and associated allocation of resources required to achieve strategic goals. He defined structure as the design of the organisation through which strategy is managed. Changes in an organisational strategy lead to new enterprise architecture (see Fig. 1). Therefore, the sustainability strategy formulation must be followed by enterprise engineering and utilisation of feedback for the strategy reformulation. The business strategy can be identified with a selected way of ensuring corporate sustainability and creating a fit between external environment, internal resources and capabilities (see Fig. 1). In this context, ICT strategy is

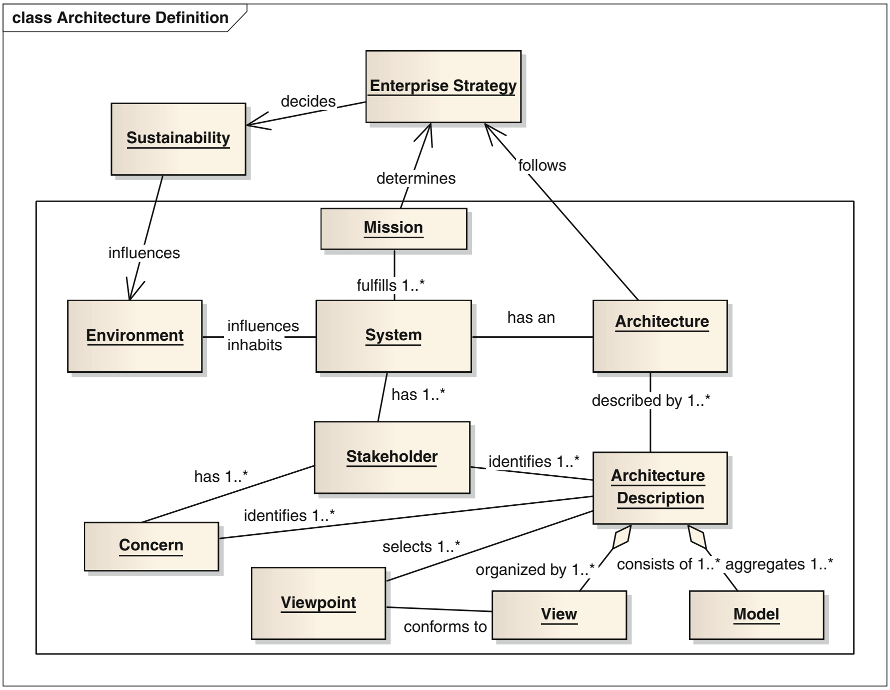


Fig. 1 Extension of the conceptual framework of IEEE 1471

a general plan or a direction of IT application in the enterprise to achieve strategic business goals. ICT strategy is a discipline that seeks to explain why organisations do what they do and how they can be changed to achieve a certain demanded purpose. The EA's role for organisation strategic development is presented in literature [19].

In this chapter, the EA model is clearly a communication tool. The EA model communicates a compelling vision of usage of ICT within an organisation and within its contacts with the business environment, to coordinate organisational strengths with environmental opportunities, to guide and coordinate supporting activities, to generate more benefits than costs and to explore new opportunities and respond to new user requirements. The same as above, questions are formulated in the EA framework e.g. Zachman Framework. Ross et al. [20] argue that strategy and architecture are relatively analogous terms. However, an architecture is often thought more of as a static picture, and a strategy is more like putting the architecture into motion, defining not only what is to be accomplished but how it is going to be achieved.

The EA as a strategic management discipline creates a holistic view of the business processes, systems, information and technology of the enterprise. As such, it is a manifestation of an organisation's mission and enables that mission (see Fig. 1). The EA consisting of ICT systems is to ensure adaptability and agility of the enterprise to respond more adequately to the dynamics of the business environment.

The EA is expected to give faster delivery of new functionalities and modifications of the legacy systems, as well as an easier access to higher quality, more consistent and more reliable information. The business organisation should eventually have a bigger consistency of business processes and information across business units. The enterprise architecture is to identify opportunities for integration of inconsistent processes and information and support the reuse of them.

Unfortunately, there are no clear and transparent guidelines that enterprise architecture framework is to support the corporate sustainability.

3.1 The Open Group Architecture Framework

In The Open Group Architecture Framework (TOGAF), architecture has two meanings:

- A formal description of a system, or a detailed plan of the system at component level to guide its implementation
- The structure of components, their interrelationships and the principles and guidelines governing their design and evolution over time [21, 22]

There are four architecture domains:

- The business architecture that defines the business strategy, governance, organisation and key business processes
- The data architecture that describes the structure of an organisation's logical and physical data assets and data management resources
- The application architecture that provides a scheme of the individual application, their interactions and their relations to the core business processes
- The technology architecture that describes the logical software and hardware capabilities that are required to support the deployment of business, data and applications services

The purpose of enterprise architecture is to optimise across the enterprise legacy systems into an integrated environment that can be responsive to business strategy and corporate sustainability through reduction of ICT resources incompatibilities and costs. The EA development is to ensure lower business operation costs, more agile organisation, lower change management costs and improved business productivity. Although it is not clearly defined in this method, improvement of interoperability, reduced complexity of the business processes and the ability to procure heterogeneous, multi-vendor open systems are the ways to ensure environmental and economic sustainability of the enterprise. With TOGAF the sustainable development of the enterprise is supported by the major components within an Architecture Repository that are as follows:

- The Architecture Metamodel describing the organisationally tailored application of an architecture framework
- The Architecture Capability defining the parameters, structures and processes that support governance of the Architecture Repository

- The Architecture Landscape which is the architectural representation of assets deployed within the modelled enterprise [22]

TOGAF guides the selection and integration of specific services to create an architecture useful for building reusable solutions across a wide number of relevant domains. The reuse of building components is a typical activity for sustainable development of EA and for corporate sustainability.

3.2 Zachman Framework

According to Zachman, his framework is a model or ontology for understanding and managing a change of enterprise [23]. Zachman assumes that architecture is the result of work of several actors, who present diverse interests and have to negotiate the final state of the enterprise architecture. So, the EA is an integrated and transparent representation of all enterprise interests. The Zachman Framework (ZF) brings attention to stakeholders' integration, transparency of their opinions and alignment challenges associated with the EA. The Zachman's model provides deep insights into the descriptions of EA and interrelationships among them. The Zachman enterprise architecture framework promotes a top-down approach to development. Within the framework, the considerations are conducted on six levels. The ZF differentiates between the levels: scope model (contextual, planner view), enterprise model (conceptual, owner view), system model (logical, designer view), technology model (physical, builder model) and detailed representation (subcontractor, user model). Each of these views is presented as a row in the matrix. The lower the row, the greater the degree of detail of the level represented. The model works with six aspects of the enterprise architecture: data (what), function (how), network (where), people (who), time (when) and motivation (Why). Each view (column) interrogates the architecture from a particular perspective. Taken together, the matrix cells create a complete image of the enterprise. Such holistic visualisation allows for establishing a certain equilibrium among the views of different stakeholders. The ZF establishes a common vocabulary and set of perspectives for defining and describing complex enterprise systems. The key role is that of the business enterprise architect, who is responsible for documenting, analysing and designing the business processes and functions, products, business units and interactions among them. Governance approach towards the EA development is to understand the business better, to provide an overview of all the units within the enterprise described from different viewpoints, to ensure collaboration between all the units and to add value to the users through continuous enhancement of business processes.

3.3 Federal Enterprise Architecture Framework

The Federal Enterprise Architecture Framework (FEAF) promotes interoperability and sharing of information among US federal agencies and other governmental entities [24]. The FEAF components of an enterprise architecture are as follows: architecture drivers, strategic direction, current architecture, target architecture, transitional processes, architectural segments, architectural models and standards. The FEAF supports the establishment of the scope of the enterprise architecture similarly as it is in the Zachman Framework. The FEAF method also accepts the actor-oriented approach, including Planner, Owner, Designer, Builder and Subcontractor Perspective and demanding analysis of Data, Application and Technology Architecture from that five viewpoints. So, the holistic model of EA is the result of negotiations and compromises of the different stakeholders.

3.4 C4ISR Architecture Framework

The Command, Control, Computers, Communications (C4), Intelligence, Surveillance, and Reconnaissance (ISR) framework covers three views [25]. The operational view describes and integrates the operational elements, tasks and activities and information flows required to accomplish mission operations. The system view describes systems and their performance to the operational view. The technical view describes the minimal set of rules governing the arrangement and interdependences of system components. The framework aims to ensure that the architecture is the description, from different perspectives, of the integrated, interoperable and cost-effective capabilities in the field.

3.5 Treasury Enterprise Architecture Framework

The Treasury Enterprise Architecture Framework (TEAF) provides guidance and template for development and evolution of information systems architecture. The TEAF's functional, information and organisational architecture views allow for modelling the organisation's processes and business operations. The enterprise architecture description is a matrix, with columns being views (functional, information, organizational and infrastructure) and rows being perspectives (planner, owner, designer and builder) [21]. The matrix supports the realisation of the transition strategy to new environment and the establishment of the sustainability of the enterprise and its architecture.

3.6 The Ministry of Defense Architectural Framework

The Ministry of Defense Architectural Framework (MODAF) is the UK Government specification for architectural frameworks for the defence industry.

The framework consists of seven viewpoints, i.e. acquisition, strategic, operational, system, service-oriented, technical and all view viewpoint [26]. All these viewpoints are interrelated and integrated to ensure long-term balance of the EA components and further improvements within the assumed scopes.

3.7 CIMOSA Framework

The Computer Integrated Manufacturing Open System Architecture (CIMOSA) is assumed to produce a formal, executable model that may be used to simulate and operate the enterprise [25]. The CIMOSA framework emphasises the necessity to transfer the executable model from the enterprise engineering environment to the operational environment. The use of two separate environments supports the implementation of parallel and concurrent processes of the EA development. The CIMOSA modelling framework is based on four abstract views (function, information, resource and organisation views) and three modelling levels (requirements definition, design specification and implementation description). The four modelling views are provided to manage the integrated enterprise model [27]. The argument for the enterprise sustainability is that CIMOSA considers enterprise integration to be a continuous process, which requires that the enterprise modelling activities should be realised simultaneously with the normal operation of the enterprise. Beyond that, the CIMOSA guidelines include building block selection from a catalogue, customisation of selected building blocks, adding variables at execution time to reduce costs and efforts of the EA development.

3.8 Dynamic Architecture Model

The principles of the Dynamic Architecture (DYA) model assume that enterprise architecture aims at achieving coherence and cohesion. Architecture investments have a chance to be approved if they are an integral part of the investment necessary to attain important business objectives. By providing a clear insight into the relationships between various architectural objects (processes, information, applications) and various architectural levels (strategic, tactical and operational) within an organisation, the transparent relationships are defined and the risk of uncontrolled growth of noncompliant solutions will be reduced [28]. The enterprise software application maintenance may demand on development. The DYA model assumes clear defining of independent software components to make adaptations and implementation easier in the future. The EA internal cohesion is achieved because of the anticipative strategy and the ICT governance to coordinate the activities in such a way that they contribute towards achieving business objectives.

4 Enterprise Architecture Principles for Corporate Sustainability

Although there are much more frameworks of EA, e.g. Generic Enterprise Reference Architecture and Methodology (GERAM), Purdue Enterprise Reference Architecture (PERA), Lightweight Enterprise Architecture (LEA), Nolan Norton Framework, Extended Enterprise Architecture Framework (E2AF) or Technical Architecture Framework for Information Management (TAFIM), the role of the EA as a support factor for corporate sustainability development is not explained directly [21, 29]. However, generally, the established EA frameworks allow to develop ICT projects that are expected to be cost-effective and shared solutions that aid the enterprise in the long term. They are coherent with the standards and industry regulation compliant. The beneficiaries of ICT projects within the EA should be responsible for the correctness of financial and economic analyses, cohesion of information included in the project and the EA documentation, correctness of cost qualifications, technical, organizational, legal and financial feasibilities as well as each project and the whole EA stability and energy effectiveness of proposed ICT solutions. All the criteria are important for corporate sustainability from the ICT projects' point of view. The EA principles can be positioned as instruments to articulate an enterprise's future direction and its sustainability, while serving as a coordination and steering mechanism towards the actual transformation of the enterprise. The basic EA principles need to drive the behaviour within the enterprise. Therefore, the elements of the EA should be understandable, robust, sufficiently defined and precise to support consistent decision making, complete, consistent, and agile to accommodate changes. The EA is the bridge between the business strategy and organisational design, but it must be perceived as the normative restriction of design freedom [30].

Particularly important to users is the capability of integrating the information among software applications and across data warehouses and data marts. By understanding the enterprise architecture, they can develop a standard data dictionary and develop metadata standard to minimise data inconsistency. The EA ensures the traceability between business processes, data, user roles, applications and infrastructure. The traceability and integration support corporate ICT sustainability. Otherwise, i.e. if an organisation does not have a clear model of its business application and technical infrastructure, they are not monitored and maintained consistently in the long term. The better monitoring of ICT assets provides a greater understanding of the interrelated issues of business-ICT alignment and support electric energy and human efforts reduction. The development of the EA enables complex preparation for new technologies, smarter project realisation and the reuse of the EA components and best practices. Standardisation for the EA drives ICT procurement efficiencies, because of the opportunities of economies of scale, reduced skills maintenance and trainings.

For architecture development the conceptual integrity is an important consideration in system design [31]. Problems that enterprise integration can solve include the following: information aggregation, single point of data entry, process efficiency, web channel integration, supplier integration and supply chain optimisation.

5 Conclusion

Concluding, the EA is the principal mechanism for establishing the fundamentals in enterprise design, managing the knowledge of the enterprise and long-term integration of the technology into the enterprise. The EA implements risk-monitoring mechanisms and generates technical guidelines of how the service delivery function makes optimal use of ICT assets, thereby maximising cost-effectiveness. So far, the enterprise architecture frameworks are poorly supported by the principles and tools to control the enterprise sustainability. The EA modelling should be the formal or formalised approach for addressing the organisational complexity and sustainability and the realisation of a unified and integrated design. The corporate sustainable development should be determined by the EA that aims at ensuring the communication between users and developers, as well as at the validation and verification of implemented systems, including material for reuse repositories.

References

1. Stavris JM, Sprangel JR (2008) SOAR from the mediocrity of status quo to the heights of global sustainability. In: Wandel CH, Stone JAF (eds) Innovate approaches to global sustainability. Palgrave Macmillan, NY, pp 11–38
2. Bondy K, Matten D, Moon J (2007) Codes of conduct as a tool for sustainable governance in MNCs. In: Benn S, Dunphy D (eds) Corporate governance and sustainability. Routledge, London, pp 165–186
3. Hilty LM, Seifert EK, Treibert R (2005) Information systems for sustainable development. IDG, Hershey NY
4. Unnerstall H (2005) Sustainable development as legal term in European Community Law, making it operable within the habitats directive and the water framework directive, Department Environmental and Planning Law 16/2004, July. UFZ Centre for Environmental Research Department Environmental and Planning Law, Leipzig
5. Sunden S, Wicander G (2005) ICT in developing countries: to be sustainable or not – is that the question? In: ISD'2005 proceedings of the fourteenth international conference on information systems development: pre-conference, 14–17 August. University of Karlstad, Karlstad, Sweden, pp.103–115
6. Shrivastava P (1995) The role of corporations in achieving ecological sustainability. Academy of management review 20(4):936–960
7. Mette Kjaer A (2004) Governance. Polity, Cambridge
8. Martin A, Benn S, Dunphy D (2007) Towards a model of governance for sustainability. In: Benn S, Dunphy D (eds) Corporate governance and sustainability. Routledge, London, pp 94–121
9. Truex D, Olsson L, Lindblad-Gidlund K, Sefyrin J, Larsson A, Nilsson O, Anderson K, Borglund E, Asproth V (2011) Position statement: sustainable information and information

systems (SIIS). In: Nuttgens M, Gadatsch A, Kautz K, Schirmer I, Blinn N (eds) Governance and sustainability in information systems, managing the transfer and diffusion of IT. Springer, Heidelberg, pp 306–310
10. Skou Petersen G (2011) Sustainability and IS research – An interaction design perspective. In: Nuttgens M, Gadatsch A, Kautz K, Schirmer I, Blinn N (eds) Managing the transfer and diffusion of IT. Springer, Heildeberg, pp 310–316
11. Ricardo RJ (2008) Organization design, a practical methodology and toolkit. HRD, Amherst, MA
12. Albertsen T, Sandkuhl K, Seigerroth U, Tarasov V (2010) Modelling network-based defense, success and failure of an enterprise modelling endeavour. In: van Bommel P, Hoppenbrouwers S, Overbeek S, Proper E, Barjis J (eds) The practice of enterprise modeling. Springer, Berlin, pp 121–130
13. Coad A (2005) Strategy and control. In: Berry AJ, Broadbent J, Otley D (eds) Management control, theories, issues and performance. Palgrave Macmillan, New York, pp 167–191
14. Mintzberg H, Quinn JB (1991) The Strategy process, concepts, contexts, cases. Prentice Hall, Englewood Cliffs
15. Gottschalk P (2005) E-business strategy, sourcing and governance. Idea Group Publishing, Hershey
16. Lee KJ (2004) Business models and governance strategy of policy knowledge service for national knowledge management. In: Wimmer MA (ed) Knowledge management in electronic government, 5th IFIP international working conference on knowledge management in electronic commerce, Krems, Austria, proceedings. LNAI 3035:27–37
17. Neef D (2001) E-procurement from strategy to implementation. Prentice Hall, Upper Saddle River
18. Chandler AD (1962) Strategy and structure: chapters in the history of the American industrial enterprise. MIT Press, Cambridge, MA
19. Radeke F, Legner CH (2012) Embedding EAM into strategic planning. In: Ahlemann F, Stettiner E, Messerschmidt M, Legner C (eds) Strategic enterprise architecture management. Springer, Berlin, pp 141–168
20. Ross JW, Weill P, Robertson DC (2006) Enterprise architecture as strategy. Harvard Business School Press, Boston
21. Minoli D (2008) Enterprise architecture A to Z, frameworks, business process modeling, SOA, and infrastructure technology. CRC, London
22. Open Group Standard (2011) TOGAF Version 9.1, ISBN: 9789087536794 http://pubs.opengroup.org/architecture/togaf9-doc/arch/
23. Zachman J (2007), Architecture is architecture IS architecture, EIM Insight Magazine 1(1), March online at http://www.eiminstitute.org/library/eimi-archives/volume-1-issue-1-march-2007-edition/architecture-is-architecture-is-architecture, 5 January 2012
24. General Accounting Office, Office of Management and Budget (2001) A practical guide to Federal enterprise architecture, version 1.0. http://www.gao.gov/bestpractices/bpeaguide. pdf. Accessed Mar 2012
25. Noran O (2003) A mapping of individual architecture frameworks (GRAI, PERA, C4ISR, CIMOSA, ZACHMAN, ARIS) onto GERAM. In: Bernus P, Nemes L, Schmidt G (eds) Handbook on enterprise architecture. Springer, Heidelberg, pp 65–212
26. Holt J, Perry S (2010) Modelling enterprise architectures. The Institution of Engineering and Technology, London
27. Spadoni M, Abdmouleh A (2007) Information systems architecture for business process modelling. In: Saha P (ed) Handbook of enterprise systems architecture in practice. Information Science Reference, Hershey, PA, pp 366–382
28. Wagter R, van den Berg M, Luijpers J, van Steenbergen M (2005) Dynamic enterprise architecture, how to make it work. Wiley, Hoboken, NJ
29. Bernus P, Nemes L, Schmidt G (2003) Handbook on enterprise architecture. Springer, Berlin
30. Greefhorst D, Proper E (2011) Architecture principles, the cornerstones of enterprise architecture. Springer, Berlin
31. Lam W, Shankararaman V (2007) Dissolving organisational and technological silos: an overview of enterprise integration concepts. In: Lam W, Shankararaman V (eds) Enterprise architecture and integration: methods implementation and technologies. Information Science Reference, Hershey, NY, pp 1–23

Understanding IS Risk Emergence from Escalation Perspective

Nipon Parinyavuttichai and Angela Y. Lin

Abstract This paper investigates IS risks from an emergent perspective. The analysis of project escalation unearths the emergence of IS risks and the situations within which IS risks are produced and reproduced. To gain a deeper understanding, this study examines an unsuccessful IS project which was overrun and over-budget. The results show that project escalation plays an important role in risk emergence; therefore, to better understand and manage risks, one must understand causes of escalation. In this study the causes of project escalation were mainly psychological and communicational, and the escalation could have been avoided if they were recognised and managed properly before more resources were committed to wrong course of actions. The study offers contributions to both literature and practice.

1 Introduction

Organisations have been making significant investments in information systems (IS) projects as they increasingly rely on IS for products and services development, production, delivery and distribution [24]. Many IS projects however are reported unsuccessful as they are poorly managed, over-budget and overrun [1, 11]. The concern over project success rate has led to a stream of research on causes of IS project failures [4]. One frequently noted reason is mismanagement of IS risks where project teams fail to identify, recognise and understand risks, and subsequently they manage risks poorly or not manage them at all [12, 16].

N. Parinyavuttichai (✉)
Rajabhat Rajanagarindra University, 422 Marupong Road, Cha-choeng-sao, Thailand
e-mail: Nipon.par@csit.rru.ac.th

A.Y. Lin
The University of Sheffield, Information School, Regent Court, 211 Portobello, Sheffield, S1 4DP, UK
e-mail: A.Lin@Shef.ac.uk

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_29,

IS risks can be defined as the uncertainties that arise in IS development which prevent the success of the project teams in fulfilling user expectations and delivering the envisioned system on schedule and within the expected budget [19]. As commonly understood, IS risks can be identified and defined in advance and they are static in nature. That is, the content and quality of the risks are time and context independent and will be the same throughout a project. Such understanding of IS risks, to an extent, helps project teams avoid obvious project risks and prepare contingency plans to minimise potential damages brought by the risks. Nevertheless, in practice not all IS risks can easily be identified and defined prior to a project and many in fact emerge during the project. Risks are dynamic because they are formed and reformed in situ and through social interactions. In other words, the content and quality of IS risks will change as the situation and project develop. Parinyavuttichai [16] hence argues that in order to properly manage IS risks, it is critical to recognise that risks are dynamic and often emerge from project situations. He continues to argue that the conventional risk management that assumes risks are static and predictable cannot sufficiently address the IS risks.

The literature on the notion that IS risks are dynamic and emergent is limited. Thus this paper intends to contribute to this gap in the literature and is organised as follows. The next section explains what escalation theory is and how it can contribute to the examination of the emergent process of IS risks drawing on escalation theory. This is followed by a description of the research strategy including case study method and data collection and analysis techniques. To gain a deeper understanding, a case study was carried out to investigate an unsuccessful IS project which was overrun and over-budget. Next, the findings are presented and it is followed by a discussion. Finally, the conclusion and implications of the research are discussed.

2 Theoretical Background

Risk management in the context of IS project is regarded as an effort to develop and implement solutions to encounter all possible threats to the project success. And it is believed that by implementing the appropriate risk management countermeasures can mitigate IS risks and even turn around a failing project to a successful one [19]. It has been suggested that IS risk management involves a number of linear processes including identifying, planning and analysing risks and implementing risk management strategies [5, 17]. Identifying and defining potential risks is the first and foremost process as it enables a project team to anticipate and prepare contingency plans to mitigate the occurrence of risks and contain the damage when risks become manifest [12, 14]. This linear approach to IS risk management seems to take a static view and assumes that risks are independent from time and context and can be understood as they are. Such approach overlooks the dynamic features of IS risks. First, not all occurrence of IS risks can be anticipated and many emerge from situations during the project. Second, IS risks are both consequences and causes of other risks. Third, the consequences of poor risk management can become risks for the project. Therefore IS risks should be understood and managed as emergent phenomena which are formed and reformed throughout a project.

The notion of escalation has been used in IS research to describe and explain how and why erratic decisions are made to commit additional resources to a seemly failing project [2, 3, 8, 16]. Escalation of commitment can cause subsequent project complexity and even leads runaway project. Escalation is a complex phenomenon which is influenced by many different factors. Staw and Ross [23] suggest that the determinants affecting escalation in IS settings can be grouped into four perspectives: project, psychological, social and structural. The project perspective concerns about objective features of a project and escalations occur when individuals keep on their actions due to the concerns over the objective values, e.g. financial and nonfinancial values. For instance, a decision maker may keep allocating resources to a project because he/she considers that the project will benefit the organisation in the long run [8] or because he/she believes that the project is getting close to complete [10]. The psychological perspective concerns over decision makers' relationship with the project and escalations arise because of emotional attachment that decision makers have with the project [2, 22]. For instance, a project team may intentionally digest only the information which matches its view and understanding of the project and ignore other aspects [22]. The social perspective considers the features of decision makers' social surroundings, and escalations occur because individuals are unwilling to lose their public status, and hence they would try everything to make themselves look good in front of the public [6, 8]. For example, a project team may decide to commit additional resources on a failing course of action because they do not want to reveal their mistakes to others or because they are not ready to take the responsibility for the project failure [16]. Finally, the structural perspective considers the contextual conditions surrounding the project, and escalations arise because of the organisational, cultural and managerial reasons, e.g. continuation of project champion, strong support from the top management of the organisation and administrative inertia.

In addition to the above four perspectives, communication problems are suggested as frequently quoted reasons for project escalation [3, 4]. Two commonly encountered errors in project communication that cause project escalation are deaf effect and mum effect. Deaf effect refers to a situation that a decision maker refuses to hear the report of bad news from members of the project team or external auditors [9]. Project managers tend to ignore or even reject warning messages that would help them to avoid project escalation when the bad news reporter is not regarded as reliable or the possibility of risk occurrence is low [3]. On the contrary, mum effect explains the situations where members of a project team are not willing to report bad news to the project manager because their assessment of the ongoing project situations (in conjunction with personal responsibility to the bad news) suggests that it is more logical not to report the bad news [21].

The above escalation perspectives provide a useful taxonomy of factors that explain why projects escalate. They accentuate erratic decisions made by decision makers and people involved in the project and how these decisions lead to the emergence of risks. Guided by the notion of escalation, this study examines how risks emerge from project escalation and how they structure the further risks.

3 Research Methodology

This study used case study method to explore the phenomenon of IS risk emergence. The case study method was chosen because it enabled us to understand the phenomenon in its context and understand the phenomenon from participants' point of views [26]. Being able to examine the phenomenon in its context is important to this study because it allows us to identify the trajectory of risk emergence which cannot be achieved through a quantitative research method such as survey questionnaire. Moreover, case study allows us to collect a variety of evidence to ensure the validity of participants' statements as well as develop a holistic picture through rich descriptions [25].

The case study is concerned with a centralised accounting system project, named 3D (pseudonyms) launched in a university in Thailand. This 3.5 million baht (Thai currency) project was to replace the existing cash-based accounting system based on a new national accounting standard. The system was designed to be used by all departments in the university to process financial and accounting transactions and for the executives of the university to support their decision making. The project was first launched in 2003 and aimed to be delivered and implemented in 2006, but only 50–60 % of the project were completed and implemented in 2007. In 2010, the project manager announced the project delay and promised that the system will fully operate in 2012.

The primary data sources of this case study were semi-structured interviews and project documentation. The data collected from the semi-structured interviews provides rich description of the project from the participants' point of views. A set of standard questions was prepared before the interviews, and additional questions were asked to follow up the interviewees' answers. Four key members of the project team were invited and involved in the interviews including project manager, project leader, system analyst and project developer.

Documentation was collected to enhance the credibility of this research. Documentation is generally used to validate the data obtained from the interview and to cross-check spelling of the project name, name of the people involved in the project and technical terms given by the interview participants [20]. In this study, project documents include the university's profile, project background, system design and diagrams and meeting minutes.

The data collection of this research followed the ethic procedure in the researchers' institution. An ethic application form for data collection, information sheet about the project and consent form for the interviewees were prepared and submitted to the research committee within the academic department for approval before collecting data. All interview scripts and documentation were treated as confidential and kept in a safe place only the researchers were able to access.

Thematic data analysis method was used. Generally, this analytical approach is used to identify specific issues of interest or themes which lead to the ability to systematically examine the issues of interest [6]. For this research, it helped the researchers to identify and group the data obtained from the field into (but not limited to) the predefined themes, i.e. IS risk emergence due to escalation behaviours. Once these themes were identified, further investigation and analysis of the collected data could be made.

4 Findings

The main reasons for 3D project escalation are organised under two headings: psychological and communication.

4.1 Psychological Dimension

Psychological factors contribute to escalations in 3D project mostly including errors in information processing and framing and decision makers' emotional attachments. Errors in framing and information process led to requirement risks and resulted in project escalation that the project team continued to commit with inaccurate information. With an assumption that user requirements will be the same across the university, the project manager made a decision to randomly select representatives from user departments that he was familiar with. Another reason for the project manager's decision to collect requirements from a few user departments was his method to reduce the complexities involved in collecting user requirements.

> The selection criteria were entirely up to us. The project manager made suggestions to the (university) president of how we would select representatives from user departments (Project leader).

Relying on the requirements made by only a few and randomly selected user departments especially by the staff in finance department, the project team understood the issues and framed the system requirements partially. The project team's bias towards the information collected from certain user departments implied that the overall system requirements were not understood properly or were even misunderstood. Unaware of the errors made in framing the situations and user requirements and bias in information process, the project proceeded to the next stage of the project: designing the system. Since the system design was based on a partial view, it failed to live up to user expectations and was rejected by the users.

Emotional attachment to the project was observed. A senior system analyst who had been working for the project from the beginning of the project was getting overconfident and unwilling to listen to others' suggestions especially regarding the system design. Conflict hence arose between the system analyst and the experts from the computer science department who were invited to provide their help during the requirement and collection and analysis phases.

> The key system analyst was overconfident with her knowledge [...]. She never listened to any suggestions about the database design given by those from computer science (Project leader).

Even when the system analyst's ego had got in the way and affected the performance of the project team, the project leader was reluctant to communicate with the analyst directly regarding the problems because he was concerned about his collegial relationship with the analyst in the long run. As a result, the expert from the computer science department decided to withdraw from the project. The emotional attachment to her method and way of doing things explains why the analyst continued designing the system the way it was despite the advisors had criticised the design. The result of this was the inaccurate system design and further complexities of the project.

> Later we discovered that the way in which the system analyst used to design the system did not follow the principle of ISD. For instance, her coding was too difficult for others to understand and modify; and data dictionary used in the database was impractical and wrong [...]. Hence, it required us more time and efforts to resolve the problems (Project leader).

4.2 Communication Dimension

Ineffective communication and lack of communication within the project team and between the project team and users were the sources of project escalation. The communication problems within the project team led to various risks especially personnel risks and escalation of project.

It was reported that project problems were not brought to the decision makers' attention because the project team believed that the management would not respond to the problem.

> I decided not to inform the project leader the problems such as insufficient number of testing server and system tester, and requirement creep, because he usually failed to respond to my request. (Project developer)

One of the problems that was communicated but not managed was a shortage of personnel involved in the project. The reasons for the short of project personnel were twofold. First, the initial recruitment policy was unrealistic so that the project could not recruit people who could meet the selection criteria. Second, the project lost its personnel because of job uncertainty. It is interesting to note that the short of qualified personnel was the cause as well as the outcome of risk. This is because each time the project lost a member, work would naturally fall on those who stay on the project, and the more people left the project, the heavier the workload became and less incentive for people to stay. This became a vicious circle; hence in order to solve the problem, the recruitment policy was revised downward: to lower the selection criteria. In so doing however did not help the situation as new staff had less experience and knowledge of system development.

> I believe that the project manager might have thought that we had a lot of spare time. I heard that there were 33 applicants applying for the available positions in the project team, but none of them met their criteria. How could that be? [...]. At one stage, I ended up taking all responsibilities for the system design as no one else helped me (System analyst).

The project leader admitted that he was busy with giving lectures, doing research and taking care of other administrative roles and paid little attention to the project. The lack of constant project monitoring created opportunities for the project team to hide and not report the problems to the leader. For example, the project team hid some vital information about the project progress from the project leader and tried to solve the problems themselves. But instead of solving the problems, the project became more complex, for example, the uncertainty of user requirements was accentuated by the changing requirements.

> They (system analysts and system developers) allowed some users to change their requirements without asking me for the permission. They should have asked me before they allowed the changes because I might be able to prevent undesirable consequences […]. What frustrated me was that they dug themselves a hole. The project could not move on to the next phase and I could not see the future of this project either (Project leader).

5 Discussion

This study examines the process of risk emergence with an aim of having a deeper understanding of risks from an emergent perspective. The study was informed by the notion of escalation and argues that risks and project failure are often the products of project escalation which occur during a project and therefore are not predictable. The key reasons for escalation can be viewed from five perspectives: project, psychological, social, structural and communicational. In this study the reasons that trigger the process of project escalation were mainly psychological and communicational.

It is evident that psychological reasons especially framing and emotional attachment to the project can result in partial view and cloud people's judgment and hence affect the direction of the project. Framing issues such as understanding of the scope of a project and user requirements and expectations of the project have been discussed in the literature [6, 8]. The study shows that to ensure correct understanding of the scope of a project, user requirements and problems is important and can help avoid committing time and resource to wrong course of action further down the line. The study also shows that decision makers' emotional attachment to a project can create bad atmosphere in the project team which impedes its overall productivity. The emotional attachment can also cause the decision makers to turn a blind eye on to obvious problems [6]. The conventional IS risk management strategies have assumed that decision makers are rational, and they always weigh up the cost and benefit of each decision; however it is not always as shown in the study, e.g. emotional attachment, bias interpretation of the project and information process. Therefore IS risks management should also take into account psychological issues at both individual and project level.

The study also shows how ineffective communication and lack of communication can lead to project escalation. In this study both mum effects and deaf effects were observed and proved to be damaging to a project. Deaf effect was found when the system analyst turned a deaf ear to expert opinions, project leader turn a blind eye to the project problem and project team chose to listen to few users and ignore others. The literature suggests that the causes of deaf effect can be gender differences, risk perception and societal collectivism [7, 20], and this study shows that deaf effects can also be psychologically, politically and organisationally motivated

[9]. For example, psychologically the system analyst's emotional attachment to the project and her own way of doing things resulted in her turning a deaf ear to expert opinions, politically the project leader turned a deaf ear to the complaints about the system analyst, and organisationally the decision makers did not respond to problems reported. The project escalation could have been avoided if project team, system analyst and the project leader were prepared to listen and communicate problems with others actively.

Mum effect was also observed in the study where there was a tendency in the project that negative information was not communicated with the authority. Mum effect can be a result of deaf effect in the case where people rather not to report the problems and try to solve problems themselves instead of reporting to the manager because they believe the management will not respond to problems. This is known as deaf–dumb–blind condition where mum and deaf effects are both strong, and bad news will neither be reported nor heard [9].

6 Conclusion

This study offers two major contributions to current literature. First, unlike many of the current studies, this study argues and proves that occurrence of IS risks cannot be identified in advance as they often emerge from situations during the project development. This argument provides an alternative view to look into what become of risks and their impacts on the subsequent events. Second, the study shows how escalation theory can be applied in IS risk research to examine the process of IS risks emergence through occurrences of escalation. The study also contributes to risk management practice by highlighting the weakness of linear approach to project management and suggest that in order to effectively manage IS risks attention should be paid to the situations that are likely to encourage project escalation.

The discussions of the paper are based on an in-house IS project within a university; therefore the future research can consider investigating the same topic in different project settings such as implementing package software, outsourcing projects and organisational settings, e.g. government organisations and business organisations.

References

1. Boehm BW (2000) Project termination doesn't equal project failure. IEEE Computer 33(9):94–96
2. Brooks L, Eke B (2010). IS project escalation in developing countries. Americas conference on information systems, August 12–15, 2010, Lima, Peru
3. Cuellar MJ (2009). An examination of the deaf effect response to bad news reporting in information systems projects. PhD thesis, Georgia State University
4. Cuellar MJ, Keil M, Johnson RD (2006) The deaf effect response to bad news reporting in information systems projects. E-Serv J 5(1):75–97

5. Dey PK, Kinch J, Ogunlana SO (2007) Managing risk in software development projects: a case study. Ind Manage Data Syst 107(2):284–303
6. Drummond H (1998) Is escalation always irrational? Organisation Stud 19(6):911–929
7. Gelfand MJ, Bhawuk DPS, Nishii LH, Bechtold DJ (2004) Individualism and collectivism. In: House R, Hanges P, Javidan M, Dorfman P, Gupta V (eds) Culture, leadership and organizations. Sage, Thousand Oaks, CA, pp 437–512
8. Keil M (1995) Pulling the plug: software project management and the problem of project escalation. MIS Quarterly 19(4):421–447
9. Keil M, Robey D (2001) Blowing the whistle on troubled software projects. Commun ACM 44(4):87–93
10. Keil M, Bernard CY, Wei K-K, Saarinen T, Tuunainen V, Wassenaar A (2000) A cross-cultural study on escalation of commitment behavior in software projects. MIS Quarterly 24(2): 299–325
11. Keil M, Cule PE, Lyytinen K, Schmidt RC (1998) A framework for identifying software project risk. Commun ACM 41(11):76–83
12. Kwak YH, Stoddard J (2004) Project risk management: lessons learned from software development environment. Technovation 24(11):915–920
13. Meissonier R, Belbaly N, Houzé E (2006) In: The 11th international conference of the association information and management *(AIM)* Luxemberg, pp. 318–329
14. Project Management Institute (2004) A guide to the project management body of knowledge (PMBOK guide). Project Management Institute, Newtown Square, PA
15. Schmidt R, Lyytinen K, Keil M, Cule P (2001) Identifying software project risks: an International Delphi study. J Manag Inform Syst 17(4):32
16. Silverman D (2006) Interpreting qualitative data. Sage, Thousand Oaks, CA
17. Smith HJ, Keil M (2003) The reluctance to report bad news on troubled software projects: a theoretical model. Inform Syst J 13:69–95
18. Snow AP, Keil M (2002) The challenge of accurate software project status reporting: a two-stage model incorporating status errors and reporting bias. IEEE T Eng Manag 49(4): 491–504
19. Staw BM, Ross J (1987) Behavior in escalation situations: antecedents, prototypes, and solutions. Res Organ Behav 9(40):39–78
20. Tiwana A, Keil M (2004) The one-minute risk assessment tool. Commun ACM 47(11): 73–77
21. Walsham G (1995) Interpretive case studies in IS research: nature and method. Eur J Inform Syst 4:74–81
22. Yin RK (2009) Case study research: design and methods. Sage, Thousand Oaks, CA

Profiles in Professional Social Networks

Jaroslav Pokorný

Abstract This paper discusses user profiles as an important component of online social networks (OSN). A special category of OSN includes professional social networks, where so-called professional profiles are significant. They enable to connect not only people but also projects to people, courses to students, etc. A powerful tool for representing profiles is ontologies, particularly various classification hierarchies. A contribution of this paper is a matching framework able to consider profiles, whose some features are described by concepts from classification hierarchies. Moreover, users can assign weights to these concepts and influence an associated similarity measure. We discuss the notions of similarity and compatibility of such profiles and show some new possibilities how to tackle the matching problem.

1 Introduction

Online social networks (OSN) are emerging as a new type of application on the Internet, which can be considered as a natural extension of Web applications that establishes and manages explicit relationships between users [10]. OSNs have reached a great development and popularity for users to connect as well as express content and share it. For example, MySpace (http://www.myspace.com) with over 275 million users, Facebook (http://www.facebook.com) with over 845 million users in February 2011 and Orkut (http://www.orkut.com) with over 100 million users mainly from India and Brazil are examples of popular networks used to find and organize contacts. A special category of OSN includes professional social networks (PSN), e.g. well-known LinkedIn (http://www.linkedin.com) and Academia. edu (over 1.2 million academics in 2012)—a free social networking Website and

J. Pokorný (✉)
Department of Software Engineering, Charles University of Prague, Prague, Czech Republic
e-mail: pokorny@ksi.mff.cuni.cz

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_30,

collaboration tool aimed at academics and researchers from all fields. The other examples of PSNs include HR.com (0.194 million users in 2010)—the largest social networking site dedicated to the human resources professionals—and ResearchGate (http://www.researchgate.net) with 0.9 million users. We are reminded also one of the oldest network communication systems USENET (http://www.slyck.com/ng.php) developed by the academic community. For ICT community in the Czech Republic, there exists the social network SoSIReČR (http://www.sitit.cz). A more detailed overview of PSN specialized on ICT can be found, e.g. in [6].

OSNs adapt real world social structures to online channels, both web and mobile. Their members construct personal profiles with the information they want others to know, share interests through recommendations, links or documents and build lists of people with whom they are connected to. PSNs use not only personal profiles but also profiles of other actors, like companies, teams and projects.

Most of OSN sites carry out a type of virtual community. When people join social networking sites, they allow users to create personal profiles viewable to anyone in a given network. Users can enter "friendship" relationships with other registered users and share content, e.g. photo albums that can be linked to the profiles of those present in a picture. In PSNs, such content includes scientific documents, research projects, university courses and PhD thesis topics.

Regarding the definition of profiles, these usually comprehend both structured information in the form of key-value pairs (features) and unstructured or semistructured information, mainly text-free fields or uploaded contents. Structured information provides basic descriptors about actors. For a person some descriptors are name, age, gender, schools attended, geographical location, interests or Web page. Features describing employee's expertise usually occur in PSNs.

PSNs are often business-oriented social networks where many core services such as recruiting, job seeking, expert/profile search and item recommendation rely on successful identification of similar actors. A similar approach can be found in educational systems which adapt the learning material to the knowledge of a student, display personalized help texts or tailor descriptions to the technical background of an actor. The PSNs based on these general ideas are used to be supported by a web portal containing a browsable repository.

A particular problem of OSN is user's profiles matching required in several scenarios. One of them consists of linking data corresponding to the same actor in the same or different data sources (see, e.g. [11]). The other one concerns matching two actors inside one OSN. In the context of PSN, the latter includes, e.g. coupling a project and appropriate researchers and searching for suitable job seekers based on an advertisement described as a professional profile.

Our contribution in this paper is a novel matching framework able to consider profiles, whose some features are described by concept names from a classification hierarchy. The framework is based on distances between topics and their matching. Moreover, users can assign weights to the features and influence an associated similarity measure (score). As a main result of the paper, we consider our approach to profile matching, particularly a measure called profile compatibility. Its values enable, e.g. to rank candidates for a position in an organization with respect to a given requester profile.

Section 2 describes some necessary notions concerning OSN. In Sect. 3 we present an overview of profile management issues and approaches. Section 4 is devoted to classifications usable for profile construction. In general, ontologies present an appropriate framework for this task today. A method for profile matching is described in Sect. 5. First, we will introduce the notion of concept distance in a hierarchy, different types of matching two concepts in a hierarchy as well as the notion of profiles compatibility. As examples we will use ACM Computing Classification System [1] containing hierarchies of ICT topics. Finally, in Sect. 6 we suggest current and future plans in profile processing research.

2 Towards Professional Social Networks

We start with an intuitive definition of a social network.

Definition 1
Given an undirected graph G (V, E) representing a *social network*, where V, $|V|=n$, is the nonempty set of vertices representing *actors* and E, $|E|=m$, is the set of edges representing the *relationships* among them. Let v_i and v_j be two vertices from V; if $e(v_i, v_j) \in E$ then v_i and v_j are *neighbours*.

To take *actors' profiles* into account, let F_V be the set of *features* describing the actors of the social network, which can be represented by a matrix of size $n \times |F_V|$. Note that the features in this definition represent single-valued attributes. In practice, a multi-valued attribute is a more real case, e.g. hobby = [reading, chess, music]. We will use the employee's expertise expressed by a set of topics. Also each feature could be optionally an aggregation of single-valued attributes (e.g. date of birth = [day, month, year]).

The relationships between actors depend on the type of society. Between companies, the relationship could be a business contract of supply. Between people in a company, it could be the hierarchical relationship if we are considering the organizational structure, or it could be the sending of e-mails in a network of relationships between friends. The relationships define a *social graph*. Depending of the social network, relationships can be qualified or not. They are often bidirectional and may express, e.g. relationships between persons, users and user groups.

Technically, the functionalities of OSN can be usually described in seven blocks supporting the actors' activities: identity, profiles, presence (neighbours, friends), relationship types, messaging, repository, calendar and events. We may distinguish the identity of an actor and his different profiles representing the facets that he exposes (private, professional profile, etc.).

However, social networks are more than a graph; they have an interesting amount of information derived from its social aspect, such as profile information, content sharing and annotations, among others. Suppose, e.g. an OSN of the employees from enterprises. Based on information retrieval methods and graph processing, we can use the OSN relationships and explore whether the OSN actors have sent messages to each other or not and derive the type of projects each employee has been involved in. Thus, we can search communities of people which have met before and have worked in similar projects. Those communities of cooperation could be used to find experts or to create work teams in the enterprise.

We will also consider the employees' expertise explicitly and discover special communities for practical purposes. In this case, we talk about PSNs.

3 Profile Management

One of the most important components of an OSN is the profile page. For example, Orkut profile pages basically consist of 68 features. The profile page reflects a single user or a company. Every user on the social network can access this information, and the company can specify and maintain exactly what information it makes available to the user. Usually the profile page is managed and administrated via a portal. In OSN, the profile page's management and administration are carried out by reading and writing comments.

We are interested in so-called professional profiles (PP). Informally speaking, a PP is the machine-readable description of an extent of knowledge in an area. A little more formally, a PP is an instance of a structured data type, whose part is a hierarchy of categories (usually a classification tree). We will formally define PP in the following way.

Definition 2
A *professional profile* is a set of couples $\{(T_1, w_1),\ldots, (T_m, w_m)\}$, where T_i are the terms (topics) that describe the actor and w_i denote the importance of T_i in describing the actor.

PPs are enabling to express the professional focus of a given actor (e.g. person, research team or project) in a structured way. For example, what is taught on which level in a course or what ICT knowledge on which level a person has. Also relationship types in PPs can be special ones, e.g. *supervised_by, coauthor*. In more advanced approach to PPs, organizations are modelled. They are represented by several groupings of people with common research interests.

The actor profile may not be entered directly. Extended user profiles are built by extending the basic explicitly defined user's profiles with data inferred from user

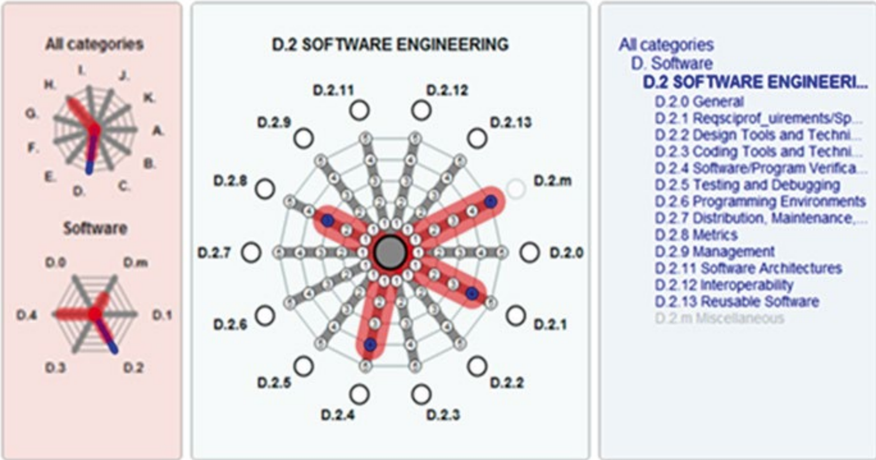


Fig. 1 Visualization of a professional profile

content analysis and user interactions with each other and with the tool itself. Actor description fields and other text-based fields can also contain implicit data regarding actor interests and preferences that can be extracted by content analysis tools. The authors of [9] developed a tool for implicit groups based on extended user profiles built by extending the basic explicitly defined user's profiles with data inferred from user content analysis and user interactions with each other and with the tool itself. User profiles can be extracted using various types of corpora, e.g. utilizing knowledge about experts from Wikipedia or analysing public expert's documents (papers, technical reports). Authors of [13] generate extended user profiles by spreading activation networks derived from ontologies. User profiles are also processed in recommendation systems.

An important feature of each profile is its visualization. For example, in project SoSIReČR [14], the classification nodes are depicted as a bar chart, arranged in a circle (the resulted graph in Fig. 1 is often called a *cobweb graph*).

4 The Role of Ontologies in Social Networks

A powerful tool for representing profiles is ontologies [7]. Ontologies are used to represent the formal specifications of the notions involved on a domain of interest and the relations between these terms [5]. These approaches can be:

- Corpus-based
- Knowledge-based

Corpus-based methods use well-known methods from information retrieval area, like vector space model, statistical model and latent semantic analysis. These methods use matching techniques well known from information retrieval area (cosine, similarity measure, Dice's coefficient and Jaccard's index). In context of user profiles, these approaches are rather naive. As the profiles are short bags of words, they often do not determine a semantic inexact match when here is no overlap of words of participated profiles.

Knowledge-based approaches are based on predefined taxonomies or ontologies, like WordNet or bibliographical classifications used in context of digital libraries or Web portals. Typically, classification tree(s) are used in these ontologies. There are various ways of ontologies organization. Some of them are hierarchical taxonomies based on ISA relationships; the other uses more relationship types, most often BT/NT (broader/narrower term), organized even not only in hierarchies but in general graphs.

Representing actors by profiles is also used in content-based filtering where the system selects rank-ordered items according to user profiles. Both items and user profiles are described either by bags of words or by concepts from ontology. For example, in the newspaper domain the authors of [12] used the ontology NewsCodes[1]—a subject classification hierarchy with three levels and seventeen categories in its first level.

A similar idea will be used here with PPs describing actors from ICT community. We focused on the knowledge-based ACM Classification Scheme [1], i.e. a typical schema that is universally accepted standard classification of ICT disciplines appropriate. Its category space is also organized in three level hierarchies. That is, the classification has a forest structure with trees of a fixed maximum depth. The ACM Classification Scheme consists of eleven major partitions (first-level subjects). These are subdivided into 81 second-level topics, which are further subdivided into third-layer topics; see, e.g. the category *Software Engineering* in Fig. 1. An uncoded fourth level contains subject descriptors, e.g. Computer-aided software engineering (CASE), Decision tables, Evolutionary prototyping, Flow charts, in category D.2.2 Design Tools and Techniques.

For example, all publications issued by ACM are annotated by this classification. There are many applications of this classification scheme for describing expertise in ICT scientific community, e.g. in recommender systems [2], in the context of a digital library [15] as well as in representing research organizations [8]. The classification was used even for determining the semantic similarity in matching software practitioners' needs and software research activities [4]. The so-called subject clusters describing groupings composing an organization are also represented by sets of ACM topics [8].

The concepts representing a topic from the profile are necessarily not the most specific ones in a path of the hierarchy. For example, if somebody is an expert in *Formal Definitions and Theory*, it does not mean necessarily that he/she knows all about *Programming Languages*. Thus it is not possible to understand ACM classification as a set of usual ISA hierarchies but rather of BT/NT hierarchies.

[1] http://www.iptc.org/cms/site/index.html?channel=CH0088

5 Profile Matching

A usual method belonging to a profile management in OSN is comparing profiles by a profile matching. Profile matching is based on the notion of similarity. It reflects closeness and interaction between actors. For purposes of this paper we omit interactions and focus mainly on static PPs. Various techniques for profile matching can be divided into two main categories:

- Syntactic-based approaches
- Semantic-based approaches

While the former provide exact or approximate lexicographic matching of two concepts, the latter use semantics for similarity definition. Here, as an example, we will use ACM classification where semantics is determined by hierarchies of topics. Obviously, in general, we can consider arbitrary concept hierarchies.

Consider first Web pages representing profiles. Technically they are part of the Web, but their data representations are different from general Web pages. In OSN, Web pages describing profiles are automatically generated and not authored by any person. Thus methods for comparing such profiles are relatively simple. Syntactic-based approaches like vector space model can be used in the case of single-valued features. For example, in Orkut, out of 68 features 20 features are considered for similarity measurement. In this case a classical notion of similarity is used in which a similarity function *sim* (a, b) is reflexive, i.e. given two PPs P_1 and P_2, in general, we have that $sim\ (P_1, P_2) = sim\ (P_2, P_1)$. Then, e.g. a group is composed from actors they are mutually similar.

In context of PSNs, a more useful case than this (*full*) *social similarity* is usage of only *partial similarity*. Suppose as actors a *requester* and a *candidate*, both expressed by respective profiles. Then a more appropriate *sim* function is not necessarily reflexive. A partial similarity is defined as the level of matching between a requester and candidates, e.g. a company finds out a new employee, a project requires a team of researchers and a student with his/her interests chooses a new study program. These examples take into account the fact that, while the "perfect" candidate seldom exists in the profiles base, profiles are often available which provide the desired requirements "to some extent". Here we use the term *compatibility* (rather than similarity) due to asymmetry in the classification schema: the functionalities of children nodes are also the functionalities provided by their ancestor(s) in the BT/NT hierarchy, while the reverse usually does not hold.

5.1 Distances Between Topics and Their Matching

Now we define the notion of distance between topics based on the forest structure [3]. Let there be t trees $(f_1, f_2, \ldots, f_t)$ in the forest F whose nodes represent topics.

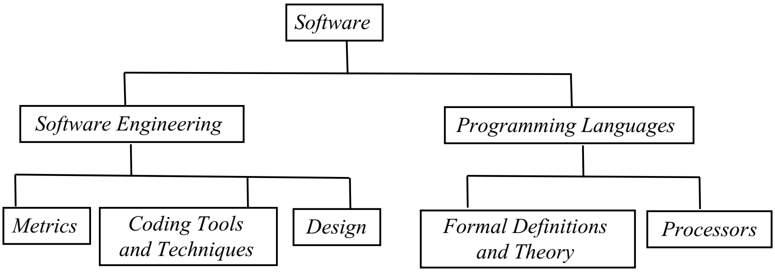


Fig. 2 A part of category D2 software engineering in the ACM hierarchy

Consider two topics T_a and T_b such that both of them belong to the same tree of *F*. Let *LCA* be the least common ancestor of T_a and T_b. Also, assume d (LCA, T_a) to be the depth of T_a from the *LCA*.

Definition 3
If T_1 and T_2 are two topics, then the *distance*, D (T_1, T_2), between them is given as:

$$D(T_1,T_2) = \begin{cases} d_{LCA}(T_1,T_2) & \text{if } T_1,T_2 \in f_i \\ \infty & \text{if no such } f_i \text{ exists} \end{cases},$$

where d_{LCA} (T_1, T_2) = max(d(LCA, T_1); d (LCA, T_2)). We do not suppose that more than one such f_i exists in classifications considered.

If T_1 and T_2 are in f_i and f_i, respectively, and $i \neq j$, then D (T_1, T_2) is ∞. Consider a part of category D2 in Fig. 2. For example, when T_1 = *Programming Languages* and T_2 = *Data structures* then, since T_1 and T_2 are in different trees, D (T_1, T_2) = ∞. Considering Fig. 2, if T_1 = *Coding Tools and Techniques* and T_2 = *Programming Languages* then LCA = *Software* and d (LCA, T_1) = 2, d (LCA, T_2) = 1 and, consequently, D (T_1, T_2) = 2. Any two topics with a common parent node have their distance equal to 1. For example, D (*Formal Definitions and Theory*, *Processors*) = 1. If $T_1 = T_2$, then D (T_1, T_2) = 0.

The Definition 2 helps us to determine the distance, when the topics belong to a single tree. Clearly, in n-level classification hierarchy, the useful distances are from interval <0, $n-1$>.

Now we will consider a pair of profiles P_r and P_c, belonging to the requester and the candidate, respectively. In our retrieval model, we exploit different types of matching between the set of features associated to the requester and those associated to available candidates. Let $\mathrm{T}_r = \{T_{r1},\ldots,T_{rm}\}$ and $\mathrm{T}_c = \{T_{c1},\ldots,T_{cn}\}$ be their associate sets of profile topics. For any two topics $T_{ri} \in \mathrm{T}_r$ and $T_{cj} \in \mathrm{T}_c$ that lie on the same path in a tree from F, we can distinguish at least three types of matches between them:

- *Perfect match*, if $d(T_{ri}, T_{cj}) = 0$
- *Close match*, if $d\ (T_{ri}, T_{cj}) = 1$
- *Weak match*, if $d\ (T_{ri}, T_{cj}) \geq 2$

The third variant of matching is relevant for n-level hierarchies with $n > 3$. So for ACM classification we obtain $d\ (T_{ri}, T_{cj}) = 2$ for weak match.

Consider $\mathrm{T}_r = \{$*Software*$\}$ and $\mathrm{T}_c = \{$*Coding Tools and Techniques, Processors*$\}$. Then four occurrences of weak matching exist: two from BT (*Software*) to NT (*Coding Tools and Techniques, Processors*) and vice versa.

How to express a similarity (score) between topics contained in the requester and candidate profiles? We denote it by *score*(T_{ci}, T_{rj}) and talk about the *score value* of the candidate topic T_{ci} with respect to the associated requester topic T_{rj}. For perfect matching it is easy, e.g. *score* (*Metrics*, *Metrics*) can be defined as 1. It is not the case for close and weak matches. For example, *score* (*Software Engineering*, *Metrics*) should be not the same as *score* (*Metrics, Software Engineering*). The former shows that the candidate is more general and in the latter case the candidate is more specific. For example, for finding an employee who is an expert in software engineering, the value of *score* (*Metrics*, *Software Engineering*) should be greater than *score* (*Software Engineering*, *Metrics*) in which the candidate is "universal" software engineer. That is, a specialist in metrics probably knows fundaments of software engineering. The same asymmetry holds for couples of weak matched topics. In the context of content-based filtering, the authors of [12] use score values 2/5 and 2/3 for close match and ½ and 1/3 for weak match. Thus, *score* (*Metrics*, *Software Engineering*) = 2/3 and *score* (*Software Engineering*, *Metrics*) = 2/5. For the weak match case, we obtain *score* (*Metrics*, *Software*) = 1/2 and *score* (*Software*, *Metrics*) = 1/3.

The restriction to matching topics lying only on a tree path might seem too restrictive to somebody. Other relationships, e.g. siblings, can be relevant for a requester. Consider *Formal Definitions and Theory* and *Processors* contained in requester and candidate profile, respectively. Although distances between siblings are equal to 1 by Definition 3, their matching is not considered here. In a spreading activation method, both profiles could be extended towards the common parent node *Programming Languages* with assignment of weights according to an influence function. How to quantify this influence offers an opportunity for a future research.

5.2 Profile Compatibility

In [14] we extended PP profiles by *weight* information added to ACM topics reflecting the actor's expertise. The actor is asked to select a number of topics of any layer

Table 1 Examples of profiles and their compatibility

No_c	P_c	No_r	P_r	$Comp(P_c, P_r)$
1	Metrics, processors	1	Software engineering (2)	0.66
2	Software		Programming languages (3)	0.4
3	Metrics, processors, formal definitions and theory	2	Programming languages (3) Processors (4)	0.86
		3	Processors (4)	1

of the ACM forest and assign each with a number between 0 and 5 expressing degree of the actor's activity to the topic. These weights can be used for ICT specialist's finding out an appropriate job. The same can be done in the case of a company creating job possibilities for ICT people. Concerning comparing of respective profiles, it is clear that the ICT specialist can be both requester and candidate. The same holds for a company.

We start with a simple situation when no candidate's topic weights are taken into account. We will suppose only nonzero weights w in requester profiles. Compatibility between candidate profile and requester profile can be defined as follows. Let P_c, P_r be profiles of a candidate and requester, respectively. P_r contains m profile topics from which n, $n \leq m$, lie on the same paths in F as some topics from P_c. Let s_i be a *score* value of a candidate's topic with respect to the associate requester topic on these particular paths. The compatibility between the candidate P_c and P_r is defined as follows:

$$Comp\left(P_c, P_r\right) = \frac{\sum_{i \in <1,n>} w_i \times s_i}{\sum_{j \in <1,m>} w_j}$$

In Table 1 we show some examples of profiles both of candidates and requesters. The numbers in () denote associated weights. The difference in compatibility between requests 2 and 3 for the candidate 3 is because of the explicit requirement of Programming Languages. The requester calls for more expertise than only Processors.

Note also that the topics of candidate profiles used in *Comp* are the most specific ones in a path of the hierarchy. Releasing this condition, we have various possibilities how to calculate *Comp* function. Suppose a number of T_{ci} lying on a path. The most natural solution is to consider such T_{ci} that is the closest to its parent or ancestor from T_r on the same path. Assume, e.g. $T_c = \{$*Metrics*, *Software Engineering*$\}$ and $T_r = \{$*Software Engineering*(2), *Processors*(3)$\}$. Then we would choose rather *Software Engineering* and not *Metrics* from T_c.

There is a question why candidate's weights are not used in *Comp* function. In extension of our approach, we will take them into account and change the formula for *Comp*. The resulted weights will approximate both types of weights. Let

Table 2 Examples of weighted profiles and their compatibility

No_c	P_c	No_r	P_r	$Comp(P_c,P_r)$
1	Metrics (1), processors (3)	1	Software engineering (2)	0.53
2	Software (4)		Programming languages (3)	0.4
3	Metrics (2), processors (3), formal definitions and theory (2)	2	Programming languages (3)	0.61
			Processors (4)	
		3	Processors (4)	0.75

$T_{ck}(v_k)$ and $T_{rj}(w_j)$ be weighted topics of respective profiles P_c and P_r. There are two possible cases:

- $D(T_{ck}, T_{rj}) = 0$

  If $v_k < w_j$, we will use w_i as the resulted weight. For example, an expert in *Software Engineering* (3) contributes by only three into *Comp* with requester topic *Software Engineering* (5). In the case $v_k > w_j$, then the candidate is too highly expert for the requester and can contribute only by v_j. Thus the resulted weight is *min* (v_k, w_j).

- $D(T_{ck}, T_{rj}) = 1$ or $D(T_{ck}, T_{rj}) = 2$.

  The same consideration can be used.

  Then the resulted function *Comp* can be expressed by

$$Comp\left(P_c, P_r\right) = \frac{\sum_{i \in <1,n>} \min\left(v_i, w_i\right) \times s_i}{\sum_{j \in <1,m>} w_j}$$

We can check the approach on examples from Table 1. In Table 2 we see that candidate weight 4 for *Software* has no influence in the resulted *Comp* value. On the contrary, small weights of *Processors* and *Formal Definitions and Theory* decrease the *Comp* values for requesters 2 and 3.

There are possibilities to take into account various other observations. For example, in ref. [3] the authors follow the assumption that semantic differences among upper-level concepts are bigger than among lower-level concepts. This fact then influences the development of similarity measure.

6 Conclusions and Future Work

We have described a method of how the comparison in PPs is used in PSN. A particular application behind the research was finding experts for professional activities. Obviously, the problem of finding experts on a given set of topics is important for many lines of business, e.g. consulting, recruitment and e-business. Our method supposes hierarchical ontologies to be at disposal and a simple technique of weighting. A substantial difference of the method from those used in friendship-like OSNs lies in certain asymmetry in estimating score of similarity between requester's and candidate's topics. We have seen that it is possible to explore more sophisticated techniques to measure compatibility or similarity of PPs.

The method can be used in any enterprise portal which is a part of a social network. For example, the selection of business partners and collaborators described by a PP can be a helpful feature of such portal. In combination with business intelligence, the methods of PPs management can contribute for the realization of dynamic business ecosystems in near future. To evaluate the method requires its use in an environment with real users, particularly in an appropriate application context, e.g. a portal of a career site.

Concerning future research, there is a lot of associated topics:

- Scores for similarity of candidate and requester topics can depend on the application domain and/or on ontologies used.
- Both requester and candidate profile can be also dynamic. Their changes could be easily controlled or derived from actor's behaviour.
- Often nonhierarchical relationships are added to hierarchical ontologies. Their consideration can significantly extend semantics of matching of profiles.
- Clustering actors according to their PPs.

Another direction of future work involves evaluating methods of comparing PPs with those used in associated areas like recommender systems and other enterprise tagging systems.

Acknowledgments This research has been supported by the grant GACR No. P202/10/0761.

References

1. ACM computing classification system. http://www.acm.org/about/class. Accessed 14 Apr, 2012
2. Afzal MT, Latif A, Saeed AU, Sturm P, Aslam S, Andrews K, Tochtermann K, Maurer H (2009) Discovery and visualization of expertise in a scientific community. In: Proc. of FIT'09, CIIT, Abbottabad, Pakistan, 16–18 Dec 2009, pp.43
3. Bhattacharyya P, Garg A, Wu SF (2011) Analysis of user keyword similarity in online social networks. Social Netw Analys Mining 1(3):143–158. doi:10.1007/s13278-010-0006-4
4. Feather M, Menzies T, Connelly J (2003) Matching software practitioner needs to researcher activities. In: Proc. of the 10th Asia-Pacific software engineering conference (APSEC'03), IEEE, Chiang Mai, Tahiland, 6–16
5. Gruber T (1993) A translation approach to portable ontology specifications. Knowl Acquis 5:199–220. doi:10.1006/knac.1993.1008
6. Kubalík J, Matoušek K, Doležal J, Nečaský M (2011) Analysis of portal for social network of IT professionals. J Syst Integr 2(1):21–28
7. Katifori A, Halatsis C, Lepouras G, Vassilakis C, Giannopoulou E (2007) Ontology visualization methods – a survey. ACM Computing Surveys 39(4):10.1–10.42, 10.1145/1287620.1287621
8. Mirkin B, Nascimento S, Pereira LM (2007) ACM classification can be used for representing research organizations. DIMACS technical report 2007–13

9. Pais MR, Morgadob C, Cunha JC (2011) Implicit groups in web-based interactive applications. In: Proc. of the 3rd int. conf. on computational aspects of social networks (CASoN). Salamanca, Spain, 175–180
10. Pallis G, Zeinalipour-Yazti D, Dikaiakos MD (2011) Online social networks: status and trends. In New Directions in Web Data Management 1 (pp. 213–234). Springer Berlin Heidelberg
11. Raad E, Chbeir R, Dipanda A (2010) User profile matching in social networks. In: Network-Based Information Systems (NBiS), 13th International Conference on (pp. 297–304). IEEE
12. Shoval P, Maidel V, Shapira B (2008) An ontology – content-based filtering method. Int J Inform Theor Appl 15(4):303–314
13. Thiagarajan R, Manjunath G, Stumptner M (2008) Finding experts by semantic matching of user profiles. In: Proc. of the 3rd expert finder workshop on personal identification and collaborations: knowledge mediation and extraction (PICKME 2008). Karlsruhe, Germany, pp 7–18
14. Vojtáš P, Pokorný J, Nečaský M, Skopal T, Matoušek K, Kubalík J, Novotný O, Maryška M (2011) SoSIReČR – IT professional social network. In: Proc. of the 3rd int. conf. on computational aspects of social networks (CASoN), 108–113
15. Wang T, Desai BC (2007) Document classification with ACM subject hierarchy. In: Proc. electrical and computer engineering, CCECE, 792–795

XML Query Adaptation as Schema Evolves

Marek Polák, Irena Mlýnková, and Eric Pardede

Abstract In this chapter we study the impact of *XML schema evolution* on related XML queries when the evolution can affect the result of those queries. We provide a novel approach and present preliminary solution to the problem. In our approach, we define changes in the schema and propagate these changes to the queries. We focus on a subset of XPath queries and show how particular changes in data structure can be propagated to them, either automatically or with user interference. The implementation of our approach is incorporated into a general evolution framework called *DaemonX* and it enables us to provide a proof of the concept.

1 Introduction

As XML [17] has become a de facto standard for data representation, there exists a large number of XML-based applications. Since most applications are dynamic, sooner or later the structure of the data in these applications needs to be changed and other related issues have to be changed also to preserve consistency. This fact has raised the issue of *evolution* and *adaptability* of XML-based applications. One particular issue that has received much attention is the evolution of XML schemas.

Current approaches involve techniques to propagate changes from selected XML schema such as DTD [1] or XSD [11] to the respective data, and vice versa [6]. Other approaches consider changes at an abstraction of a schema through a graphical representation [5, 9] and the propagation of the changes to the data. We observe that schema change can cause not only *data inconsistency* but also *query inconsistency*.

M. Polák (✉) • I. Mlýnková
Charles University in Prague, Prague, Czech Republic
e-mail: polak@ksi.mff.cuni.cz; mlynkova@ksi.mff.cuni.cz

E. Pardede
La Trobe University, Melbourn, VIC 3086, Australia
e-mail: E.Pardede@latrobe.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_31,

The XML queries over the original schema and over the new schema may return different results. The result can also be an error caused by non-existing element, as one example. These possibilities are not desirable and should be recognized and corrected. At the time of writing, we only find one related work [13] that investigates backward compatibility of queries over evolving schema.

The aim of this chapter is to propose an approach to deal with the stated problem. First, we define a model of XML schema and XPath queries and respective edit operations. Next, we study the impact of the schema changes on the queries. We show cases when the query can be directly adapted, when user interaction is required and when the adaptation cannot be decided. As a proof of concept, the approach is implemented as a part of a general evolution framework called *DaemonX* [4].

The chapter is structured as follows: In Sect. 2 we briefly introduce our XML evolution framework. In Sect. 3 we describe the model of XML schema and XPath queries and their mutual mapping. In Sect. 4 we introduce the edit operations and respective evolution algorithm. In Sect. 5 we provide a proof of the concept and we conclude the chapter in Sect. 6.

2 XML Evolution Architecture

In a previous work, we introduced a general framework for XML evolution that comprises five levels, each of which represents a different view of an XML application and its evolution [15]. The framework is depicted in Fig. 1. The lowest level, called *extensional level*, represents XML documents. Its parent level, called *operational level*, represents operations over XML documents, i.e. XML queries. And the level above, called *schema level*, represents XML schemas that describe the structure of the XML documents.

In practice, there are usually multiple XML schemas applied in a single XML-based application. The schemas might have evolved through time. Considering only the three lower levels of the framework in Fig. 1 leads to a separate evolution of each affected XML schema which is time-consuming and error-prone. Therefore, we introduced two additional levels that allow us to establish and preserve relationships among all parts of the system [15]. These levels follow the *MDA principle* [12] which is based on modeling the problem domain at different levels of abstraction. The topmost level, called *platform-independent level*, represents the whole problem domain. It consists of a conceptual model that specifies the problem domain independently of its representation in the XML formats. The level below it, called *platform-specific level*, represents mappings of the problem domain to particular XML formats. For each XML format it comprises a model of mapping of a selected part of PIM to XML element and attribute declarations.

The proposed architecture was implemented as a general framework *DaemonX* [4] which focuses on propagation between the data levels (except for the operational level) and supports any kind of data format. In this chapter we focus on one of the parts of the framework, which is the propagation of changes from XML schemas to the respective XML queries.

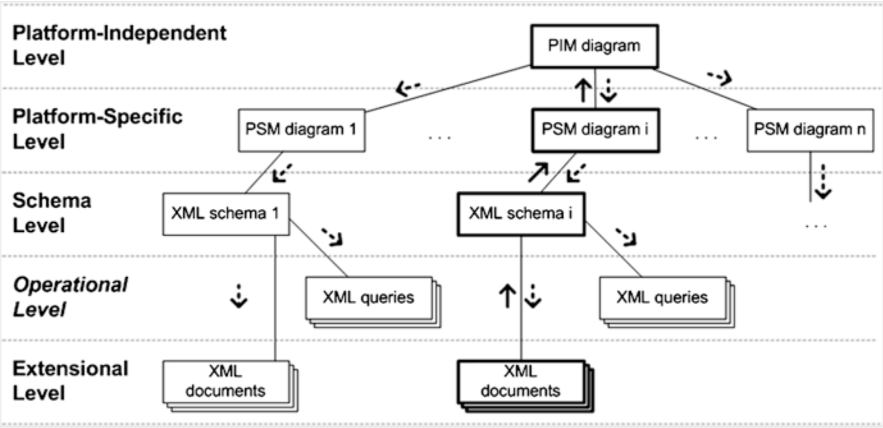


Fig. 1 Five-level XML evolution architecture

3 Models for XML Schema and XPath

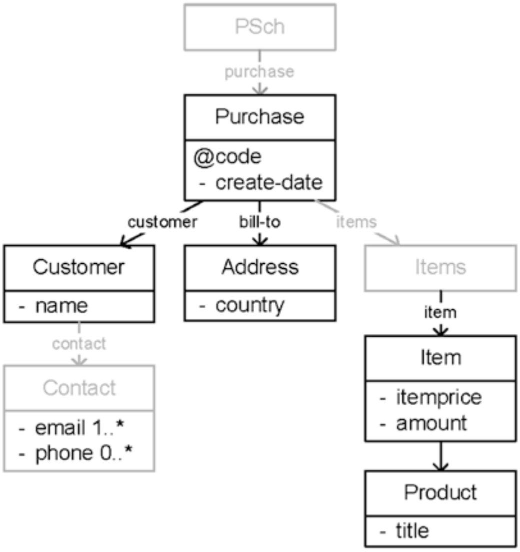
The *platform-specific model (PSM)* of the platform-specific level enables us to specify how a part of the reality modeled at the platform-independent level is represented in a particular XML schema. In addition, the designer works in a UML-style way which is more user-friendly than editing the XML schema. The model we use is called *XSEM* [14].

Definition 1

A *PSM schema* is an 8-tuple $\mathcal{S}' = (\mathcal{S}'_c, \mathcal{S}'_a, \mathcal{S}'_r, \mathcal{S}'_e, C'_{\mathcal{S}'}, content', class', participant')$. $\mathcal{S}'_c$, $\mathcal{S}'_a$ and $\mathcal{S}'_r$ are sets of *classes*, *attributes* and *associations*, respectively. $\mathcal{S}'_e$ is a set of *association ends*. An association is an ordered pair $R' = (E'_1, E'_2)$, where E'_1, E'_2 are different association ends. Any two associations are disjoint. $C'_{\mathcal{S}'} \in \mathcal{S}'_c$ is a *schema class* of $\mathcal{S}'$. Function *content′* assigns a class C' with an ordered sequence of all associations with C' as the parent.

A PSM schema is displayed as a UML class diagram in an ordered tree layout which reflects the hierarchical structure of XML data. Note that we omit names, types and cardinalities from the definition for simplicity. We do not cover all the schema constructs—they are covered in the full definition of our model [14]. An actual XML schema can be automatically generated from our PSM schema and vice versa. A sample self-explanatory PSM schema is depicted in Fig. 2.

For the purpose of evolution of XPath queries related to XML schemas, there must exist a mapping between an XML schema and an XPath query. Since the full




```xml
<?xml version="1.0"?>
<purchase code="1234">
  <create-date>9/13/2010</create-date>
  <customer>
    <name>Martin Necasky</name>
    <contact>
      <email>necasky@...</email>
      <phone>+420254...</phone>
      <phone>+420777...</phone>
    </contact>
  </customer>
  <bill-to>
    <country>Czech Republic</country>
  </bill-to>
  <items>
    <item>
      <itemprice>10</itemprice>
      <amount>3</amount>
      <title>White book</title>
    </item>
    <item>
      <itemprice>55</itemprice>
      <amount>1</amount>
      <title>Green book</title>
    </item>
  </items>
</purchase>
```


Fig. 2 Sample PSM schema

XPath syntax is too extensive, we use its subset based on the *Positive Core XPath* [8] with some modifications. Our syntax at the current stage does not consider predicates and we add the operator except to the definition.

$$\mathbb{X} \equiv \mathbb{X} \,|\, \mathbb{X} \;||\; /\mathbb{X} \;||\; \mathbb{X}/\mathbb{X} \;||\; (\mathbb{X}) \;||\; \mathbb{X} \;\; except \;\; \mathbb{X} \;||\; \mathbb{A} :: \mathbb{L}$$

$$\mathbb{A} \equiv self \;||\; child \;||\; parent \;||\; descendant \;||\; ancestor$$
$$||\; preceding \;||\; following \;||\; descendant-or-self \;||\; ancestor-$$
$$or-self \;||\; preceding-sibling \;||\; following-sibling$$

where $\mathbb{X}$ denotes *location path* and $\mathbb{A}$ represents an *axis*. As we can see, the only one node test is possible—*name test*, denoted $\mathbb{L}$.

The original Positive Core XPath definition contains predicates, but it can only be used to test for element/attribute occurrence. A query using predicates can be rewritten to a query without them and still returns the same *result set* [3]. This solution has only one problem—the query is transformed to a complex form not transparent for the designer at the first sight. In the defined syntax, it is also possible to use all classical XPath abbreviations for axes, such as "$*$" for all child elements, " " for the child axis, "." for the self axis, ".." for the parent axis and "//" for $/descendant-or-self :: node()/$.

To be able to map XSEM PSM diagram to XPath query, an *XPath model* must be defined. We propose a model that follows ordered tree structure of the XPath query;

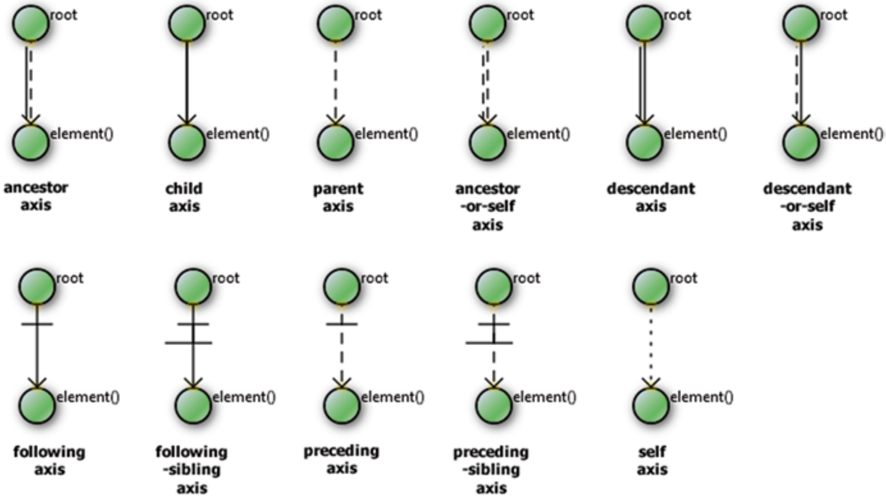


Fig. 3 XPath axes and nodes

it results from the presented syntax and it visualizes its textual representation. The components of the model can be divided into two parts—nodes that represent nodes in the location path and edges that represent axes. An edge and a node together comprise a *location step* of the XPath query. The model contains the following components:

- *Node* (E)—representing node test, or name test if name is specified
- *Axes* child (L_{ch}), descendant (L_d), descendant-or-self (L_{dos}), parent (L_{pa}), ancestor (L_a), ancestor-or-self (L_{aos}), following (L_f), following-sibling (L_{fs}), preceding (L_{pr}), preceding-sibling (L_{prs}) and self (L_s)
- *Expression node* (E_{ex}) representing *disjunction* (denoted by " | ") and *except* operators. Its first output edge represents the first part of the expression, the second output edge represents the second part of the expression. The third edge represents the following part of the query in the sense of:(*first_expression operator second_expression*)/*third_expression*

Nodes and axes are visualized in Fig. 3; an expression node is visualized in Fig. 4. In particular, we use the notation we have proposed in [16] and implemented in [4].

Since the XSEM PSM schema has a tree structure and the XPath query follows a tree structure, it is straightforward and natural to map XSEM PSM to a location path. An example is shown in Fig. 5; its formal definition is provided in [16]. As we can see, an axis can intervene not only a single node in the schema tree but also a part of a tree. We say that the part of the tree is *hit* by the location step. When the schema evolves, the query is gradually evaluated and the hit parts are compared with the previous version. If a difference is discovered, the evolution algorithm is launched.

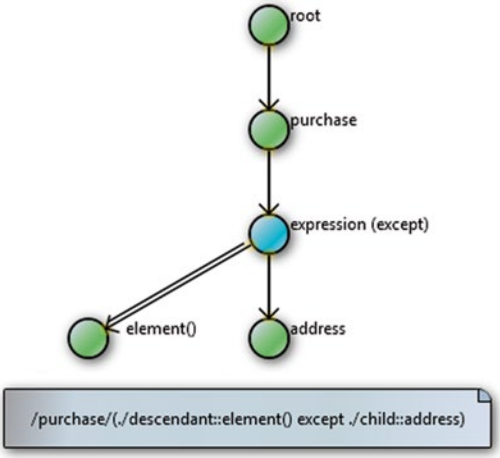


Fig. 4 XPath expression node

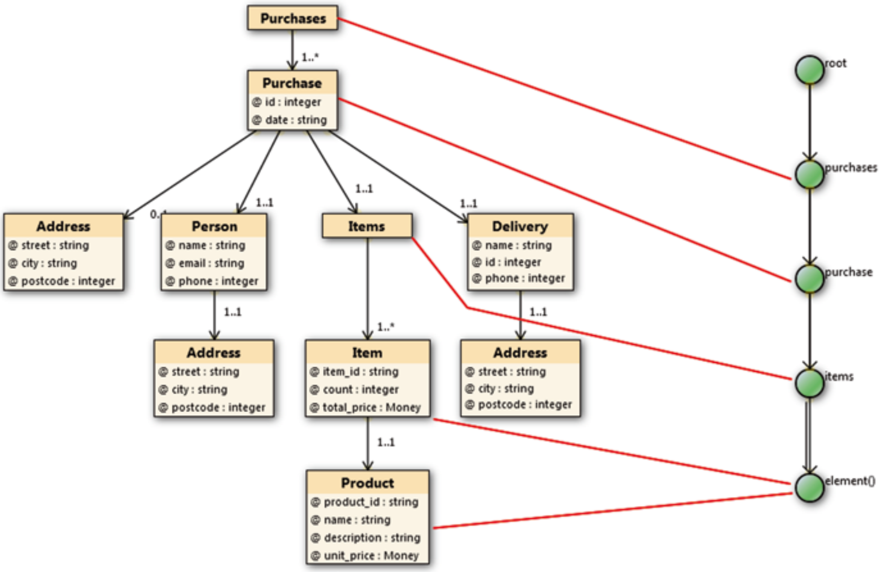


Fig. 5 Mapping between XSEM and XPath models

4 Evolution Algorithm

Every change in the source XSEM PSM can cause changes in multiple location steps of the XPath model. All operations which change the source XSEM PSM schema are *atomic*. We use the subset of operations for query adaptation identified in [10] (see Table 1). Naturally, from the atomic operations any *composite* and more user-friendly operations can be created.

Table 1 Operations for the XSEM model

Operation	Description
$\alpha(C)$	Adds a new PSM class C as the root into the XSEM PSM diagram
$\alpha(C, C_p)$	Adds a new PSM class C into XSEM PSM diagram as a child of parent class C_p. This operation creates an association A between these classes
$\rho(A)$	Removes a PSM association A from XSEM PSM diagram
$\rho(C)$	Removes a PSM class C from XSEM PSM diagram, s.t. C is a leaf node of the schema tree. When a PSM class C is deleted, all associations connected to C are deleted too
$\delta(C, name)$	Sets a new name to PSM class C
$\psi(C, direction)$	Moves a PSM class C to the left or to the right in the sequence of its siblings
$\mu(C, C_p)$	Reconnects a PSM class C as child of PSM class C_p

Table 2 Operations for the XPath model

Operation	Description
$\alpha(E_{root})$	Adds a new root node E_{root} into XPath diagram
$\alpha(E, E_p)$	Adds a new node E into XPath diagram as a child of node E_p
$\alpha(L, E_p, E_{ch})$	Adds a new axis edge L between two nodes, child node E_{ch} and parent node E_p
$\rho(L)$	Removes axis edge L from the diagram
$\rho(E)$	Removes node E and all related axis edges
$\delta(E, name)$	Sets a new name to node E

Following the set of operations in Table 1, we established similar set of operations for the XPath model (see Table 2).

Formally, let Q be the original query over the original schema S, Q' be the adapted query over the evolved schema S', $R = Q(S)$ be the result set of Q over S and $R' = Q'(S')$ be the result set of Q' over S'. Let AO_{XSEM} be an atomic operation done in an XSEM PSM schema (from Table 1) which should be propagated and let $O\ S_{XPath}$ be a sequence of atomic operations in an XPath model (from Table 2) which was generated from AO_{XSEM} to preserve the same results of the queries with the original and the new XSEM PSM schema. Then $R = R' = Q(S) = Q'(S') = OS_{XPath}(Q)(AO_{XSEM}(S))$ if there exists an appropriate propagation algorithm which generates OS_{XPath}.

Due to space limitations, we will use some simplifications. (For full description see [16].) We will consider changes with a single class corresponding to element x (to be added, deleted, etc.). Also, if not specified otherwise, all considered elements are in a *sequence* element. In all presented situations we suppose that there exists no two sibling elements of the same name in S and in S'. In the description we will use functions with self-explanatory names, such as *parent()*, *descendant()*, *absolute_path_to()* and *absolute_path_to_previous_sibling()*.

In the following text we will consider cases when query consistency is violated (i.e. $R \neq R'$) and Q' needs to be adapted accordingly. Since each query can be divided into separate location steps, we can consider only one location step of the query Q.

Table 3 Refinement—adding element *x*

Axis	Description
Ancestor (-or-self), parent	Since x can be added only as a child element (see Table 1), adding x as ancestor/self/parent of p to the root will be solved in another location step
Child	If $Q = p / child :: *$, then $Q' = p / child :: *\ except\ absolute\_path\_to(x)$ (see Example 1) *Note:* If x is inserted into choice or when *minOccurs* of x is 0, then $Q' = Q$, i.e. no change propagation is needed
Descendant (-or-self)	If $Q = p / descendant :: *$, then $Q' = p / descendant / :: *\ except\ absolute\_path\_to(x)$. *Note:* This modification is possible only if there exists no sibling element $q \in S'$, s.t. $name(q) = name(x)$. Otherwise, we should use function *position* which in combination with different values of *minOccurs* disallows precise selection of x. Therefore, we assume no sibling elements in S and S' with the same name If $Q = p / descendant :: str$, then revalidation is needed only when $name(x) = s\ t\ r$
Following (-sibling)	If $Q = p / following :: *$, then $Q' = p / following :: *\ except\ absolute\_path\_to(x)$
Preceding (-sibling)	If $Q = p / preceding :: *$, then $Q' = p / preceding :: *\ except\ absolute\_path\_to(x)$
Self	Solved in another location step

4.1 Adding

This operation adds an element x as child of an existing element in S. In the current location step we consider context element $p \in S$ (Table 3).

Example 1

Consider S in Fig. 6 on the left and XPath model of $Q = /vehicle / child :: * / registration\_number$ in Fig. 7 on the left. If element *motorcycle* is added as a child of element *vehicle* (see Fig. 6 on the right), sub-query $/vehicle / child :: *$ will return all elements including *motorcycle*. Hence, the location step is updated from $child :: *$ to $child :: *\, except\ /vehicle / motorcycle$ and $Q' = /vehicle / (child :: *\ except\ /vehicle / motorcycle) / registration\_number$. The model of Q' is shown in Fig. 7 on the right.

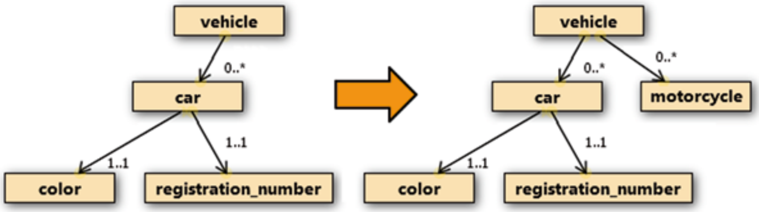


Fig. 6 Schema example for adding

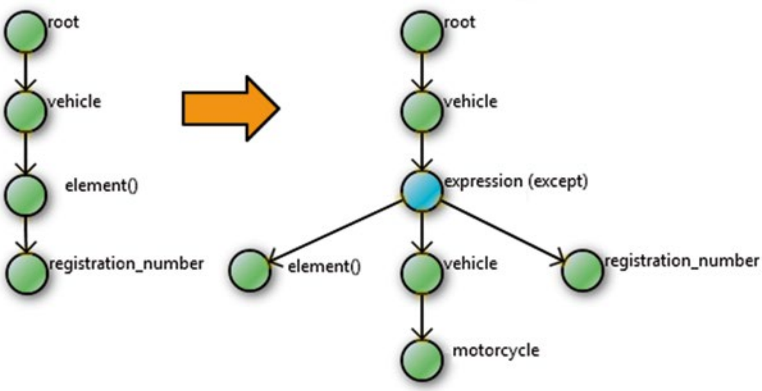


Fig. 7 Query example for adding

4.2 Removing

This operation removes element x from S. According to Table 1, the removed element must be a leaf of the schema tree. If we want to remove whole sub-tree, it can be done by its iteration. Let $p = parent(x)$, the particular cases are discussed in Table 4.

Example 2

Consider S in Fig. 8 on the left and $Q = //Address/following::*$ in Fig. 9 on the left. If element $/Purchase/Person/Address$ is removed (Fig. 8 on the right), Q is updated as depicted in Fig. 9 on the right.

Table 4 Removal—removing element x

Axis	Description
Ancestor	If $Q = x / ancestor :: *$, then $Q' = absolute\_path\_to(p) / ancestor-or-self :: * \mid x/ancestor :: *$
Ancestor-or-self	If $Q = x / ancestor-or-self :: *$, there is no possibility to preserve query compatibility, because $R' \subset R$. In other cases, no update is needed
Child, Descendant (-or-self)	If $Q = x / child :: *$ and x is a leaf node, then $R = \phi$ If $Q = p / child :: *$, it is not possible to preserve query compatibility, because the removed element x cannot be hit
Following	If there are more occurrences of element x (i.e. elements with the same name) in S and x is hit, then: If x is not the only child of p and not the last one, then $Q = x / following :: *$ is updated to $Q' = x / following :: * \mid absolute\_path\_to\_next\_sibling(x) / (descendant-or-self :: * \mid following :: *)$ If x is the only child of p or the last one with this name in S, then $Q = x / following :: *$ is updated to $Q' = x / following :: * \mid absolute\_path\_to\_next\_element\_right(x) / (descendant-or-self :: * \mid following :: *)$ (see Example 2) *Note:* Next element right is the first element returned by the following axis
Following-sibling	If x is not the only child of p, then $Q = x / following-sibling :: *$ is updated to $Q' = absolute\_path\_to\_next\_sibling(x) / (self :: * \mid following-sibling :: *) \mid x / following-sibling :: *$
Preceding	Symmetric situation to the following axis.
Preceding-sibling	Symmetric situation to the following-sibling axis.
Parent	$Q = x / parent :: *$ is updated to $Q' = absolute\_path\_to(p) / self :: * \mid x/parent :: *$.
Self	$Q = x / self :: *$ cannot be updated, since x does not exist anymore.

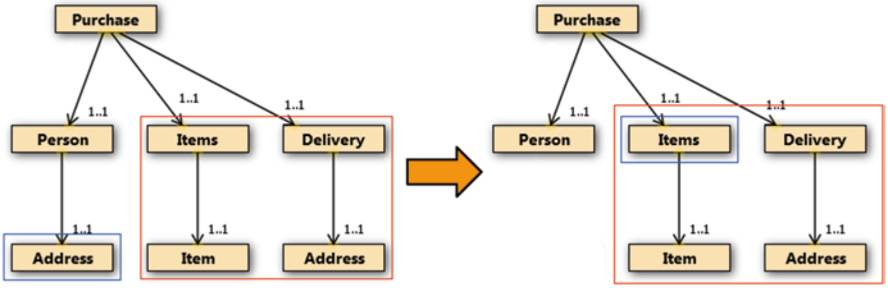


Fig. 8 Schema example for removal

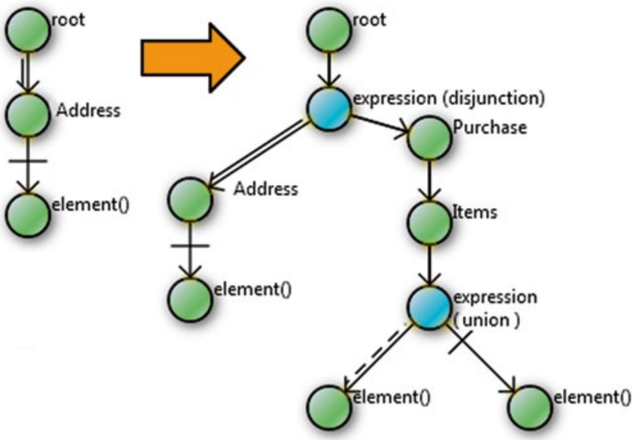


Fig. 9 Query example for removal

4.3 Renaming

Renaming operation changes the name of a selected element x. Change of the name can cause a change of the result set R. Possible situations are similar for all axes, so we do not provide the respective table. Update of Q must be done only if x is hit by the name test. If the location step uses name test with $*$ or *element*(), no change is needed. There are two cases how the change of the name can affect the result of the query. Let the new name of x be y, then:

- If more elements are in the result set ($R \subset R'$), the location step must be extended with *except absolute_path_to_element*(y)
- If less elements are in the result set ($R' \subset R$), the location step must be extended with $|$ *absolute_path_to_element*(y) (see Example 3)

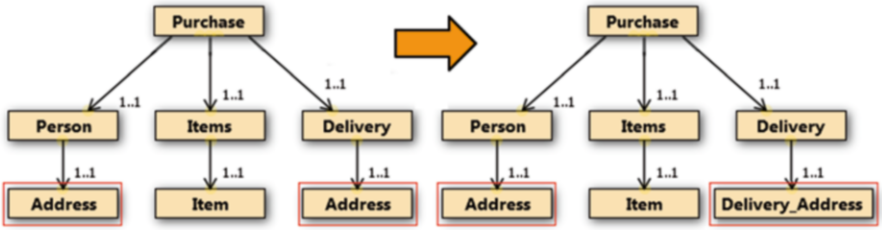


Fig. 10 Schema example for renaming

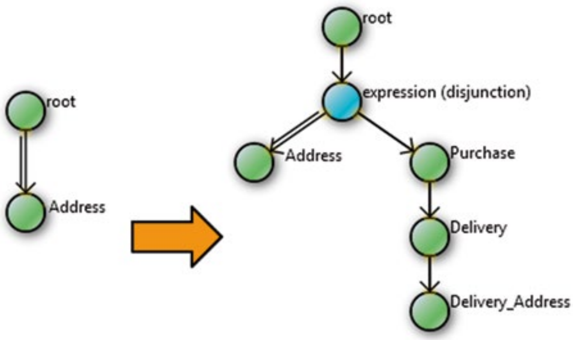


Fig. 11 XPath example for renaming

Example 3

Consider S in Fig. 10 on the left. In the red rectangles there are classes returned by $Q = //Address$ shown in Fig. 11 on the left. If the name of class $/Purchase/Delivery/Address$ is changed to *Delivery_Address*, then $Q^{'} = //Address \mid /Purchase/Delivery/Delivery\_Address$ depicted in Fig. 11 on the right.

4.4 Reordering

Now we suppose that an element x is in a *sequence* where the order of elements is significant and that element x has at least one sibling. It can be moved only one position left or right in one step (see Table 1). Again, its iteration can provide various reordering of the whole sequence. Let $p\ o\ s(x)$ be the position of x in S within its siblings and $p\ o\ s'(x)$ be the position of x in S' within its siblings. Consider elements x, y, z, p such that $parent(x) = parent(y) = parent(z) = p$, $pos(y) = pos(x) + 1$ and $pos(z) = pos(x) - 1$. (Due to space limitations we omit the table with all cases.)

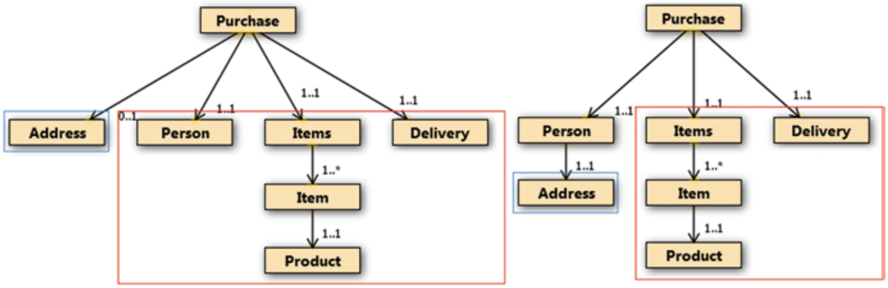


Fig. 12 Example of reconnection problem—following axis

4.5 Reconnection

For simplicity, we consider that element x can be reconnected only as a child of one of its siblings or as a sibling of its parent. Multiple iterations of the operation enable the element move to any place in the schema. Let $l(x)$ denote the level of x and $l(root) = 0$. Then x can move to $l(x) + 1$ or $l(x) - 1$. Next, we again assume that it is not possible to move an element as a child of an element where its child has the same name.

Note that an update of the query after a reconnection in the schema can be done with all axes, only if the location step is the last one in the query. If not, it is not possible to ensure query consistency in a simple way.

In general, reconnection of an element implies that an axis applied on this element can return different result than if they were applied on the element in the original position (see Fig. 12 for an example with following axis when element *Address* is moved). The particular location step has to be updated with disjunction of $R \setminus R'$ and exclusion of $R' \setminus R$.

Suppose that element x is moved to another position in the schema tree, i.e. we change parent element p of x to element p'. Let q be a parent of element p and r be a sibling of x. The particular situations are discussed in Table 5.

5 Proof of the Concept

The full implementation of the proposed approach was incorporated into the *DaemonX* framework [4] (see Sect. 2). (Note that except for Fig. 1, all figures are screen shots of the system.) Since there are no existing real-world project that provide similar abilities, it is not possible to compare our solution and results of others. Therefore, queries from *XPathMark XPath-TF* [7] are used to provide a proof of the concept. In addition, since this test set is quite simple, we also created our own more complex queries using various axes to test the abilities of the solution.

Firstly, from the test set we selected tests corresponding with our XPath syntax, i.e. A1–A11, P1–P11 (rewritten to queries without predicates) and O1, O3, O4. Examples of the queries are as follows:

Table 5 Reconnection—moving an element x

Axis	Description
Ancestor	If x is moved up, s.t. $parent(x)=q$, then $Q = x / ancestor :: *$ is updated to $Q' = x / ancestor :: * \mid absolute\_path\_to(q)$ If x is moved down, s.t. $parent(x)=r$, then $Q = x / ancestor :: *$ is updated to $Q' = x / ancestor :: *\ except\ absolute\_path\_to(r)$
Ancestor-or-self	If x is moved up, s.t. $parent(x)=q$, then $Q = x / ancestor-or-self :: *$ is updated to $Q' = x / ancestor-or-self :: * \mid absolute\_path\_to(q)$ If x is moved down, s.t. $parent(x)=r$, then $Q = x / ancestor-or-self :: *$ is updated to $Q' = x / ancestor-or-self :: *\ except\ absolute\_path\_to(r)$
Child	If x is moved up, s.t. $parent(x)=q$, then $Q = p / child :: *$ is updated to $Q' = p / child :: * \mid absolute\_path\_to\_element(x)$ If x is moved down, s.t. $parent(x)=r$, then $Q = p / child :: *$ is updated to $Q' = p / child :: * \mid absolute\_path\_to\_element(x)$ If x is moved up, s.t. $parent(x)=q$, then $Q = q / child :: *$ is updated to $Q' = q / child :: * \mid absolute\_path\_to\_element(x)$ If x is moved down, s.t. $parent(x)=r$, then $Q = r / child :: *$ is updated to $Q' = r / child :: *\ except\ absolute\_path\_to\_element(x)$
Descendant (-or-self)	If x is moved up, s.t. $parent(x)=q$, then $Q = p / descendant :: *$ is updated to $Q' = p / descendant :: * \mid absolute\_path\_to\_element(x)/ descendant-or-self :: *$ If x is moved down, s.t. $parent(x)=r$, then $Q = p / descendant :: *$ does not need any update If x is moved down, s.t. $parent(x)=r$, then $Q = r / descendant :: *$ is updated to $Q' = r / descendant :: *\ except\ absolute\_path\_to\_element(x) / descendant-or-self :: *$
Following	If $Q = x / following :: *$, the reconnected element is not x and the reconnection is done in the part of the tree hit by the following axis, no update is needed If $Q = x / following :: *$ and the reconnection of an element y causes that it is added into a part of tree hit by the axis ($R \subset R'$), then $Q = x / following :: *$ is updated to $Q' = x / following :: *$ except $absolute\_path\_to\_element(y) / descendant-or-self :: *$ If an element y is moved out from the hit part of the tree ($R' \subset R$), then $Q = x / following :: *$ is updated to $Q' = x / following :: *$ $absolute\_path\_to\_element(y) / descendant-or-self :: *$ If the reconnected element is x, the revalidation depends on its position of x among siblings. Location paths to the missing elements and locations paths to exclude redundant elements must be added
Following-sibling	If y is a sibling of x, where $pos(x)<pos(y)$, and if y is moved up as a sibling of p or down as a child of one of its siblings, then $Q = x / following-sibling :: *$ is updated to $Q' = x / following-sibling :: * \mid absolute\_path\_to\_element(y)$

(continued)

Table 5 (continued)

Axis	Description
	If y is a sibling of p or a child of one of siblings of x and if it is reconnected as sibling of x, where $pos(x) < pos(y)$, then $Q = x/following-sibling::*$ is updated to $Q' = x/following-sibling::*$ $except\ absolute\_path\_to\_element(y)$
	When we reconnected element x, the same situation as in the case of following axis occurs
Preceding	Symmetric situation to the following axis
Preceding-sibling	Symmetric situation to the following-sibling axis
Parent	Reconnection of element x in both cases (up or down) causes that $Q = x/parent::*$ is updated to $Q' = absolute\_path\_to(p)/self::*except\ absolute\_path\_to\_element(x)/parent::* \mid x/parent::*$
Self	A change of the position of x does not change the result of the query

```
 //l/ancestor::* (A5)
//l/following::* (A9)
//l/descendant::* (P5)
//l/preceding-sibling::* (P7)
      //q/following::*/parent::*    except    //g/
ancestor::* (O1)
```

All these queries have been applied on the respective schema and then all possible edit operations (see Table 1) have been tested. All updates are executed correctly in all cases.

Secondly, to test the approach on more complex queries, we take XML schema of an order from *Amazon AWS* [2] used for communication with customers by Web Services. XSEM PSM model is created from this schema and a set of XPath queries utilizing all available axes in various combinations is defined. These queries are automatically mapped to the schema by the *DaemonX* framework. Examples of the queries are as follows:

```
//RegionDefinition/parent::ExcludedRegions/parent::*
/Order/ParameterizedUrls/*/*
//AmazonUpsellPreferences/child::*
//ShippingRate/following-sibling::*/descendant::*
//MerchantUpsellItem/Images/preceding-sibling::*
//ShippingMethods/following::*
//RegionDefinition/ancestor::*
//Taxamount/following::Shipping/child::*
//Images/parent::*/ItemCustomDate/ancestor::Cart
```

Next, we make various changes in the schema to simulate a designer. After propagation the results of both original and new queries are checked. No limitations are found in the evolution process.

All queries and applied changes of the schema can be found at http://www.ksi.mff.cuni.cz/~mlynkova/daemonx/queryexamples.zip.

6 Conclusion

The main contribution of our approach is the ability to recognize and analyze changes in XML schema and to update related queries respectively. If the revalidation of the query is not possible, this situation is reported to the designer. Even though the approach is complex and robust, there exists problems that are not covered. First, there are cases when the propagation cannot be proceeded and a designer must interfere. A natural extension would be to suggest possible "clues" to simplify the process. Also, after the changes are made to the queries, it can get into a non-optimized state. Hence, an optimization would be a useful extension. And, considering more complex issues, other XPath construct can be incorporated to the adaptation process.

Acknowledgements This study was supported by the Czech Science Foundation, grant P202/10/0573.

References

1. Al-Jadir L, El-Moukaddem F (2003) Once upon a time a DTD evolved into another DTD. In: Object-oriented information systems. Springer, Berlin, pp 3–17
2. Amazon (2012) Amazon Web Services. http://amazonpayments.s3.amazonaws.com/documents/order.xsd. Accessed Feb 27 2012
3. Cate B, Marx M (2009) Axiomatizing the logical core of XPath 2.0. Theory Comput Syst 44:561–589
4. DaemonX Team (2011) Daemonx. http://daemonx.codeplex.com/
5. Domínguez E, Lloret J, Luis Rubio A, Zapata MA (2005) Evolving XML schemas and documents using UML class diagrams. In: DEXA'05, pp 343–352
6. Bouchou B et al (2004) Schema evolution for XML: a consistency-preserving approach. In: Mathematical foundations of computer science, Prague, Czech Republic. Springer, Berlin, pp 876–888
7. Franceschet M (2012) http://sole.dimi.uniud.it/~massimo.franceschet/xpathmark/FT.html. Assessed Feb 10 2012
8. Hartel PH (2005) A trace semantics for positive core XPath. In: TIME '05, Washington, DC, USA. IEEE, New York, pp 103–112
9. Klettke M (2007) Conceptual xml schema evolution - the codex approach for design and redesign. In: BTW workshops, pp 53–63
10. Klímek J, Nečaský M, Mlýnková I (2011) Evolution and change management of XML applications. http://exolutio.com/TR/evolution.pdf. Accessed 5 Feb 2012
11. Mesiti M, Celle R, Sorrenti MA, Guerrini G (2006) X-evolution: a system for XML schema evolution and document adaptation. In: EDBT'06. Springer, Berlin, pp 1143–1146
12. Miller J, Mukerji J (2003) MDA Guide Version 1.0.1. Object MNgement Group. www.omg.org
13. Moro MM, Malaika S, Lim L (2007) Preserving XML Queries During Schema Evolution. In: WWW'07, New York, NY, USA. ACM, New York, pp 1341–1342
14. Nečaský M (2009) Conceptual modeling for XML. In: Dissertations in database and information systems, vol 99. IOS Press, Amsterdam
15. Nečaský M, Mlýnková I (2009) On different perspectives of XML schema evolution. In: FlexDBIST '09, Linz, Austria. IEEE, New York
16. Polák M (2011) XML query adaptation. Master Thesis, Charles University in Prague, Czech Republic. http://www.ksi.mff.cuni.cz/~mlynkova/dp/Polak.pdf
17. W3C (2010) Extensible Markup Language (XML). World Wide Web Consortium, W3.org

Handling the Complexity of ISD Projects with Agile Methods: A Conceptual Foundation

Corina Raduescu and Asif Qumer Gill

Abstract Traditional approaches to software and information systems development (ISD) cannot fulfill the challenges presented by the complexity inherent in today's dynamic and changing environments. In this study we argue that ISD projects are socially complex endeavors and suggest that agile development methods display characteristics that justify them as being appropriate for such project environments. We suggest that one theory that justifies the appropriateness of agile methods in such contexts is the complex adaptive systems (CAS) theory. We first argue that ISD projects can be treated as CAS, and second, we assess the alignment between CAS characteristics and agile methods principles. We therefore propose and discuss a preliminary conceptual foundation for handling the complexity of ISD projects with agile methods. Our future research directions seek to investigate the applicability of specific agile methods and develop a comprehensive framework that will offer a validated theoretical justification of better approaches to manage complex ISD projects in practice.

1 Introduction

In recent times ISD projects take place in increasingly turbulent business environments characterized by unpredictable markets, changing customer requirements, pressures of shorter time to deliver, and rapidly advancing technologies [5, 16, 18]. Many projects are challenged because of lack of user input and incomplete or changing requirements. According to Geneca's [10] survey, 80 % of the professionals working on software or IS projects admit they have to do a considerable amount

C. Raduescu (✉) • A.Q. Gill
University of Sydney, Darlington, Australia
e-mail: corina.raduescu@sydney.edu.au; asif.qumer@sydney.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_32, © Springer Science+Business Media New York 2013

of rework, mainly because the project is out of sync with the business requirements in 78 % of the situations, or requirements are incomplete and fuzzy in 55 % of the situations.

Recent studies suggest that many ISD projects exhibit a high level of complexity and that the dominant management perspectives on such projects are characterized by management control, deliberate project design, and planning [11, 19]. In ISD environments, complexity can be high and it seems impossible to know the exact final state of the system (e.g., requirements, design methods, project cost) whether one follows a pure top-down or a bottom-up approach. A top-down ISD methodology is therefore unlikely to be successful for a complex ISD environment [9, 13]. This situation has encouraged practitioners to develop new adaptive or agile methods that can handle ISD projects [7, 17]. Practitioners claim that agile approaches have the ability to manage changing requirements and priorities when it is impossible to know their exact final state.

Despite these claims, the applicability of agile methods to complex ISD projects is often considered challenging, because the problem is amplified by the complexity of the application domain that is often beyond the experience or expertise of developers and stakeholders involved [16]. It is therefore important to further explore whether the agile methods are appropriate for complex ISD projects.

In this paper, we suggest that one theory that offers an understanding and supports the appropriateness of agile methods in such contexts is the complex adaptive systems (CAS) theory. The main objective of this paper is to shed some preliminary light by (1) offering a theoretical foundation that employs CAS to justify the applicability of agile methods in complex ISD projects and (2) opening future research directions for developing a framework of handling and managing the ISD complexity with agile methods. We argue that despite CAS-agile alignment, it is not always appropriate to use an off-the-shelf agile method for a complex ISD environment. Therefore, we believe it is appropriate to tailor the agile methods to the specific characteristics of the complex ISD initiatives, hence our future research in this direction.

This paper is organized as follows: first, we present a brief overview of CAS characteristics. Second we conceptualize ISD projects as CAS. Third, we propose a conceptual foundation to justify agile methods in complex ISD projects by mapping the CAS characteristics to the agile methods principles. Fourth, we discuss the appropriateness of using ASD methods in complex ISD projects. Last, we present our conclusions and future research directions.

2 Characteristics of Complex Adaptive Systems

Today *complexity* is a major feature that characterizes the world [3]. According to complexity theory, organizations are dynamic, evolving entities that can be best understood as *complex adaptive systems* (CAS). CAS are complex systems that consist of many autonomous agents interacting with each other.

> A CAS is a dynamic network of many agents (…) acting in parallel, constantly acting and reacting to what other agents are doing. The control of CAS tends to be highly dispersed

Table 1 Key characteristics of CAS

Characteristics	Description
1. Autonomous agents	Agents are autonomous, subject to simple and localized rules or norms. There is no central control; however their behavior is not random
2. Rich interactions	Components/agents engage in interaction. A CAS needs to be studied as a complete and interacting whole rather than as an assembly of distinct and separate agents
3. Feedback processes	Each agent in CAS reacts to information flows. The information received is influenced by agent's actions. Such feedback can also strengthen or weaken agent's actions
4. Nonlinear dynamics	Small causes can be amplified into extreme events, generate structural changes and organizational evolution (e.g., the butterfly effect). It is impossible to predict the CAS behavior due to the nonlinear rich interactions
5. Emergent behavior	Holistic patterns emerge (out of a multiplicity of simple interactions) that overlay the individual behavior of agents; bottom-up manifestation of order. Small actions of agents can lead to unexpected emergent system behavior
6. High degree of adaptation	CAS often react to changes in environment by taking advantages of such changes and adapting themselves
7. Energy exchange	Information is energy that serves as an agent of change and adaptation. Agents exchange various forms of energy with their environment and among themselves
8. Self-organization	CAS capability to generate its own new forms from inner interactions, rather than imposing it from the outside. New forms may result from interactions within the system in response to disruptive events
9. Coevolution	CAS coevolve with their environment; changes in environment cause changes in their behavior, which in turn causes changes in the environment
10. Edge of chaos	A state of a balance between control and freedom. The point where continuous learning and adaptation are in balance with continuous change

> and decentralized. If there is to be any coherent behavior in the system, it has to arise from competition and cooperation among the agents themselves. The overall behavior of the system is the result of a huge number of decisions made every moment by many individual agents [12].

CAS is also seen as a *formal theory* that offers suitable lens in assisting researchers in their efforts to explain emergent organizational behavior and has gained popularity in recent years. For example, Allen and Varga [2] explain the coevolution of IS with their organizational environments. Englehardt and Simons [8] suggest that CAS offer insights that explain innovation and organizational learning as a transformation process from individual to organization in which the primary forces are self-organization and emergence. Benbya and McKelvey [4] offer insights for dealing with the emergent nature of IS alignment. Vidgen and Wang [19] propose a framework of enablers and inhibitors of agility and emergent capabilities of agile teams rooted in CAS.

We present in Table 1 the key characteristics of CAS derived from prior literature, without their full description and explanation due to space limitation. Drawing upon prior literature, we recognize that CAS can support project managers in facilitating conditions and embracing processes that promote a more flexible approach to managing projects, based on principles of emergence and self-organization [4]. We therefore argue that CAS characteristics should be recognized in order for project managers to identify constraints and opportunities that can influence the evolution of their ISD developments.

3 Handling ISD Complexity with ASD Methods

Our main argument is that developing software and IS projects can be viewed as examples of CAS. Specifically, ISD projects are heterogeneous socio-technical systems that involve a huge range of interactions among large groups of users with their own practices trying to address dynamic changes in requirements and response time to market pressures. Often in such projects, releases developed by multiple teams concurrently developing different parts of the software can be important to achieve an edge in the market. It therefore naturally follows to turn our attention to CAS when exploring processes in ISD projects. Table 2 captures the nature of such projects and the fact they exhibit the main characteristics of CAS.

Table 2 ISD projects displaying CAS characteristics

CAS characteristics	ISD projects characteristics
1. Autonomous agents	Team members have their own autonomy but they follow the rules and norms of the development team/project
2. Rich interactions	People in development teams work and interact with each other
3. Feedback processes	Development teams rely upon peoples' previous experience when making decisions. These decisions in turn influence future actions
4. Nonlinear dynamics	Predictability in ISD is very difficult. Although traditional methodologies have a big plan developed upfront, in practice, the plan changes as the development progresses. Small errors may cause major problems or even stop the project
5. Emergent behavior	Each person in the development team knows some part of the project. The team communicating and collaborating can lead to unexpected results sometime
6. High degree of adaptation	Development team needs to react to environment changes, e.g., new requirements, technology, or team change
7. Energy exchange	People communicate and exchange information inside and outside the development team
8. Self-organization	Often new teams or groups organize ad hoc to address an emergent situation
9. Coevolution	Business evolves and the development team is forced to capture the evolutionary changes
10. Edge of chaos	It is hard to develop a system in full order and strictly adhering to the initial plan. This might limit the team creativity to address unexpected situations. But it is almost impossible to create systems in full chaos. Therefore there is a need for the right balance between planning/control and freedom/flexibility in such projects

3.1 Shortcomings of Traditional ISD Methods

Traditional software lifecycle development methodologies (i.e., top-down, waterfall) grew out of a need to manage and control large-scale projects to deliver results. A common criticism of traditional methodologies is that by the time the software is delivered, the users no longer need the functionality being developed [15]. Traditional methodologies emphasize planning, predictability (i.e., one has to plan every last detail), and top-down linear development cycles (e.g., business requirements led to analysis which led to design which in turn led to development). They inherit a planed approach that relies on task breakdown and stability, i.e., signed off requirements, analysis, and lock-in design. This rigidity was also emphasized by a tendency towards compliance as a means of project compliance and control.

While traditional methodologies have worked and are still working for some organizations, for others these methodologies only added cost and complexity while providing a false sense of security by exhaustively planning, measuring, and controlling the projects. Such approaches are unable to deal with the complexity and change that occur in today's project environments, and nowadays more and more organizations are increasingly deploying Agile Software Development (ASD) methods for projects of various sizes [16].

3.2 The Promise of Agile (ASD) Methods

Industry and research community have witnessed a period of change in the way software and IS development are approached in terms of globalization, technological innovation, meeting changing demands, and the evolution of new concepts displayed by the agile methods [7]. The technical community developed the Agile Software Development (ASD) methods, such as Extreme Programming, Adaptive Software Development, Scrum, and Crystal, in response to their frustrations with traditional rigid top-down management and the negative impact on their productivity and lack of successful deliverables.

The Agile Manifesto [1] provides twelve *principles* that qualitatively characterize agile development environment. Due to space limitation we only list these principles: (1) early and continuous delivery, (2) welcome change, (3) frequently delivery, (4) collaborative work, (5) motivate individuals, (6) face-to-face conversation, (7) working software, (8) sustainable development, (9) enhance agility, (10) simplicity, (11) self-organizing teams, and (12) reflection and tuning. Further explanations and details of these principles are found on Agile Manifesto Website and will be addressed in Sect. 4 of this chapter.

While agile methods are effective in small and medium contexts, complex ISD projects often require discipline and planning to ensure their overall success [16]. Notwithstanding the positive implications, ASD methods created in the context of small and medium non-life-critical project environments are often considered

Table 3 Mapping CAS characteristics to ASD methods principles

CAS characteristics	Corresponding agile methods principles
1. Autonomous agents	4. Collaborative work
	5. Motivate individuals
	8. Sustainable development
2. Rich interactions	4. Collaborative work
	6. Face-to-face conversation
	8. Sustainable development
3. Feedback processes	12. Reflection and tuning
4. Nonlinear dynamics	11. Self-organizing team
5. Emergent behavior	1. Early and continuous delivery
	2. Welcome change
	7. Working software
	9. Enhance agility
6. High degree of adaptation	1. Early and continuous delivery
	2. Welcome change
	3. Frequently delivery
7. Energy exchange	6. Face-to-face conversation
	11. Self-organizing team
8. Self-organization	11. Self-organizing team
9. Coevolution	2. Welcome change
	4. Collaborative work
	8. Sustainable development
10. Edge of chaos	4. Collaborative work
	5. Motivate individuals
	11. Self-organizing team

challenging to be adopted in larger and more complex adaptive situations [14, 16]. We challenge this view and therefore suggest that CAS theory is an appropriate theoretical foundation that can help us explain why ASD methods are suitable for complex ISD environments.

3.3 Mapping CAS Characteristics to Agile Methods Principles

To demonstrate the appropriateness of agile methods to complex projects, we now map the ASD methods principles to CAS characteristics. Some of the CAS characteristics are captured explicitly, while others implicitly. For example, feedback, energy information exchange, and adaptability are at the roots of agile methods.

Derived from our analysis of both CAS and agile methods, we now present a typology for mapping and capturing their alignment and synergies. Note that the comparison and analysis of CAS characteristics and specific agile methods, such as XP, SCRUM, etc., are beyond the scope of this paper and will be part of our future empirical investigation. The mapping of CAS characteristics and agile methods principles (see Table 3) gives us a strong reason to argue that we can use the CAS theory to justify and understand the need for ASD methods in complex ISD environments.

As a result of this mapping, we propose a conceptual foundation to justify the use of agility for handling a CAS-type project environment. Specifically, we argue that complex ISD projects could be managed piece by piece with various agile methods, that is, decomposing them into small projects or iterations with known requirements. Because agile methods are successful in small-scale projects and there is growing recognition that the lack of agility makes it more difficult to make changes that accommodate the evolving nature of requirements [16], we suggest that ISD projects should start small and incrementally grow and integrate the small pieces together.

4 Discussion

To support our conceptual foundation, below we discuss the appropriateness of using ASD methods in complex ISD initiatives. Specifically, our discussion refers to CAS characteristics inherent in complex ISD projects.

4.1 Autonomous Agents

Cockburn [6] defines the core of ASD methods as “the use of light-but-sufficient rules of project behavior and the use of human-and communication-oriented rules.” By contrast, in CAS agents follow simple rules, but their rich interactions lead to complex emerging behavior. In ASD methods critical to success is collaboration and giving teams a level of autonomy to quickly adapt solutions to changing situations. Because people are one of the most important success factor, recognizing individuals’ skills, motivating them, and placing value on their autonomy are fundamental in agile practices. An agile team structure is not hierarchal, and its focus is to facilitate as opposed to command. Thus, ASD methods offer enough flexibility and autonomy that can enhance the creativity, responsibility, and success rate for developing complex systems, without compromises.

4.2 Rich Interactions

In ASD methods business people and developers must work together on a regular basis throughout the project. Frequent face-to-face interactions and collaborative work are extremely popular and productive in agile environments. Teamwork and collaboration form the basis for rich interactions and cooperation among developers. For example, requirements can be clarified on a daily basis with the entire project team, rather than working on once upon agreed lengthy document often misunderstood or not really used in traditional methods in practice. In complex projects, the team faces emerging requirements that can be factored into the development schedule as appropriate to ensure the right product is delivered, in other words a sustainable development. If projects of any size are to succeed, there is a need to support ongoing collaboration and face-to-face interactions, that is, creating an agile environment.

4.3 Feedback Processes

ASD methods enable developers to be more flexible and responsive to changing environments [16]. A complex large project can therefore be developed, managed, and delivered in time-boxed (approx. 2–4 weeks) short cycles or functionality-bound small increments (e.g., small number of features per iteration). Every iteration can be viewed as an opportunity for the team to get feedback and guidance from customers and how to ensure the system is delivered on time. Because feedback loops are the norm in ASD methods, we believe such methods can reduce the risk of failure for more complex ISD projects.

4.4 Nonlinear Dynamics and Emergent Behavior

Despite being criticized by lacking control, ASD methods offer some control over unpredictability by using the benefits of adaptability. In order to control the unpredictability encountered in complex projects, focus should be on iterative, incremental, and adaptive development. Since ASD methods are about that, complex project development, management, and delivery should be addressed in short iterations spread out over a period of time as opposed to a big bang. In CAS, emergent behavior is expressed as the result of rich interactions of agents being connected and working in alignment with each others. This connectivity is manifested in agile methods through teamwork and collaboration, incremental deliveries, and managing the changing nature of requirements.

4.5 High Degree of Adaptation

Agile teams react continuously to environment changes by early and continuous deliveries. An agile team works hard to keep the structure of software flexible, so that when requirements change, the impact to the system is minimal. The ability to respond to change often determines the success or failure of an IS project. When plans are built, it is important to make sure that they are flexible and ready to adapt to changes in both business and technology. As the team gains knowledge about the system and as the customers gain knowledge about their needs, certain tasks on the plan will become unnecessary, other will be discovered and need to be added. ASD methods are therefore suitable for large-scale projects that cannot be planned very far into the future because they exhibit a high degree of change and need adaptation.

4.6 Energy Exchange

In ASD methods, the most efficient and effective method of communicating information to and within a development team is face-to-face conversation. For an agile team, information must be open and flowing. Likewise, in CAS, information flow is the lifeblood of change and adaptation. But traditional ISD methods have prevented openness because of fear it may lead to chaos. By contrast, in agile world, information is free, e.g., visible documentation, status track, and team wikis, placing teams within close proximity of each other to enhance the exchange between subject matter experts and development team. Consequently, ASD methods promote the open access to information.

4.7 Self-Organization

In ASD environments, the best architectures, requirements, and designs emerge from self-organizing teams. An agile team is by definition a self-organizing team. Responsibilities are not handed to individual team members from the outside but rather are communicated to the team as a whole. For example, each team member can select their focus according to the characteristics of a module they develop, without losing the tune with the whole project. The team then determines the best way to manage and fulfill the responsibilities. This relies upon development team autonomy, derived from self-organization.

4.8 Coevolution

The development team and system coevolve as the project progresses. An agile approach focuses on iterative, incremental, and adaptive development. ASD methods welcome changing requirements even at later stages of development. Agile processes harness change to deliver customer's competitive advantage. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly. An agile team knows that its environment is continuously changing and knows that it must change with that environment to remain agile. Consequently, a better way to manage the change is to use agile teams that continually adjust their organization, rules, conventions, and relationships.

4.9 Edge of Chaos

The edge of chaos is a state where the development team works with maximum efficiency. Explicit rules in traditional ISD methods seem to inhibit creativity. By contrast, motivation, collaboration, self-organization, and visibility in ASD methods can provide a rich and rewarding experience for teams to develop better quality software. In ASD methods, the best design emerges from interactions and communication as opposed to a prescriptive solution. Governing and managing agile environments require a balance of freedom and control. In a review of agile case studies, the authors suggest that a mix of agile and traditional methods can be used to create a hybrid method that tackles the complexity of the domain and adapts agile methods to more complex situations. Therefore, it is possible to adopt a partially agile and partially traditional approach (e.g., somewhere in the middle of top-bottom and bottom-up approaches) in order to keep things under control and yet be agile at the same time, that is, to be at the edge of chaos.

5 Conclusion and Future Research Directions

The ISD projects occurring in today's ever changing environments continue to be challenged or even fail. Driven by our quest to theoretically investigate and offer practical guidance on managing ISD projects, in this paper, we use CAS as our analytical lens to theoretically justify the need for employing ASD methods during the development of complex software or IS.

Our main objective was to propose a conceptual foundation towards a framework for handling complexity of ISD projects with agile methods. We based our arguments on the alignment between theoretically derived CAS characteristics and agile methods principles derived from practice. Guided by the low applicability of ASD methods to complex and adaptive environments, we proposed and discussed a CAS-based foundation that offers the theoretical justification of employing agile methods in complex ISD projects. This is a preliminary step towards a more comprehensive framework that will contribute to both theory and practice. Despite its limitations as a conceptual work, rather than an empirical validated study, this paper presents a number of future research opportunities related to CAS, agile methods, and ISD alignment.

First, we intend to validate and use our conceptual foundation as the lens to analyze a number of well-known agile methods, such as XP and SCRUM, and test their applicability within a CAS environment. Second, we plan to explore the key characteristics of complexity in ISD projects and identify the suitability of specific agile methods in specific ISD projects and situations. Third, given the emphasis of people factor, we seek to investigate how agile methods and processes can manage the rich interactions and resource allocations when a large number of developers are involved. Fourth, after completion of these analyses, we aim to develop an overall theoretical and practical framework that will provide specific guidance to current practitioners on what agile methods could work in what type of complex ISD situations.

References

1. Agile Manifesto (2001) Manifesto for Agile software development. http://agilemanifesto.org/. Accessed 8 Jul 2012
2. Allen PM, Varga L (2006) A co-evolutionary complex systems perspective on information systems. J Inf Technol 21:229–238
3. Axelrod R (1997) The complexity of cooperation: agent-based models of competition and collaboration. Princeton University Press, Princeton, NJ
4. Benbya H, McKelvey B (2006) Using coevolutionary and complexity theories to improve IS alignment: a multi-level approach. J Inf Technol 21:284–298
5. Besson P, Rowe F (2001) ERP project dynamics and enacted dialogue: perceived understanding, perceived leeway, and the nature of task-related conflicts. Data Base Adv Inf Sy 32(4):47–66
6. Cockburn A (2002) Agile software development. Addison-Wesley, Boston
7. Cockburn A, Highsmith J (2001) Agile software development: the people factor. IEEE Computer 34(11):131–133
8. Englehardt CS, Simmons PR (2002) Creating an organizational space for learning. Learn Organization 9(1):39–47
9. Fitzgerald B, Hartnett G, Conboy K (2006) Customising agile methods to software practices at Intel Shannon. Eur J Inf Syst 15(2):200–213
10. Geneca (2011) Why a majority of business and IT teams anticipate their software development projects will fail. http://calleam.com/WTPF/?page_id=1445. Accessed 20 Jun 2012
11. Hanseth O, Ciborra C, Braa K (2001) The control devolution: ERP and the side effects of globalization. Data Base Adv Inf Sy 32(4):21–33
12. Holland JH (2006) Studying complex adaptive systems. J Syst Sci Complex 19(1):1–8
13. Keenan F (2004) Agile Process Tailoring and probLem analYsis (APTLY). In: Proceedings of the 26th international conference on software engineering. Edinburgh, Scotland
14. Koch AS (2005) Agile software development: evaluating the methods for your organization. Artech Housem Inc., London
15. Mike C (2010) Succeeding with agile: software development using scrum. Pearson Education, Boston
16. Mishra D, Mishra A (2011) Complex software project development: agile methods adoption. J Software Mainten Evol Res Pract 23:549–564
17. Paetsch F, Eberlein A, Maurer F (2003) Requirements engineering and agile software development. In: Proceedings of the IEEE international workshops on enabling technologies: infrastructure for collaborative enterprises. Linz, Austria, pp. 308–313
18. Robbins G (2001) The perception by enterprises of their implementation of enterprise resource planning package. http://www.robbinsgioia.com. Accessed 15 Jun 2012
19. Vidgen R, Wang X (2009) Coevolving systems and the organization of agile software development. Inf Syst Res 20(3):355–376

OptBPPlanner: Automatic Generation of Optimized Business Process Enactment Plans

Andrés Jiménez, Irene Barba, Carmelo del Valle, and Barbara Weber

Abstract Unlike imperative models, the specification of business process (BP) properties in a declarative way allows the user to specify *what* has to be done instead of having to specify *how* it has to be done, thereby facilitating the human work involved, avoiding failures, and obtaining a better optimization. Frequently, there are several enactment plans related to a specific declarative model, each one presenting specific values for different objective functions, e.g., overall completion time. As a major contribution of this work, we propose a method for the automatic generation of optimized BP enactment plans from declarative specifications. The proposed method is based on a constraint-based approach for planning and scheduling the BP activities. These optimized plans can then be used for different purposes like simulation, time prediction, recommendations, and generation of optimized BP models. Moreover, a tool-supported method, called OptBPPlanner, has been implemented to demonstrate the feasibility of our approach. Furthermore, the proposed method is validated through a range of test models of varying complexity.

1 Introduction

Nowadays, there exists an increasing interest in aligning information systems in a process-oriented way as well as in the effective management of business processes (BPs, i.e., sets of activities which are performed in coordination in an organization to achieve a business goal) [1]. BP management (BPM) supports BPs using

A. Jiménez (✉) • I. Barba • C. del Valle
Dpto. Lenguajes y Sistemas Informáticos, University of Seville, Seville, Spain
e-mail: ajramirez@us.es; irenebr@us.es; carmelo@us.es

B. Weber
Department of Computer Science, University of Innsbruck, Innsbruck, Austria
e-mail: barbara.weber@uibk.ac.at

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_33,

methods, techniques, and software to design, enact, control, and analyze operational processes involving humans, organizations, applications, and other sources of information [2]. Typically, the traditional BPM life cycle [1] includes several phases, i.e., process design and analysis, system configuration, process enactment, and evaluation [1].

The quality of a BP design is essential for BP improvement, which has been ranked as the number one priority for top management by the 2010 Gartner survey [3]. When using an imperative approach, however, there are lots of manual work involved since the modelling expert has to describe exactly *how* it should be done. The usage of declarative processes, in turn, allows the user to specify *what* has to be done instead of *how*, thereby facilitating the human work involved and avoiding failures. There are frequently different ways to execute a declarative model in such a way that all constraints are fulfilled. The different execution alternatives, however, can vary significantly in how well different performance objective functions such as the overall completion time can be optimized. Thus, from the declarative process specification, optimized BP enactment plans can be automatically generated, which can greatly improve the overall BPM life cycle [1]. Specifically, these plans can be used, among others, for (1) simulation [4]; (2) time prediction [5], both improving the BP design and analysis phase; (3) recommendations [6], assisting users during process enactment; and (4) the generation of optimized BP models [7, 8], facilitating the human work which is involved in the BP design and analysis phase (cf. Sect. 5.1), which are innovative and interesting topics to be addressed nowadays. The main novelties of our approach regarding existing BP proposals are that our proposal (1) considers optimization and (2) deals with resource allocation.

In this work, we propose a tool-supported method, named OptBPPlanner,[1] for the automatic generation of optimized BP enactment plans from declarative process specifications to optimize the overall completion time. In this way, our proposal facilitates the work of the modelling expert since she only needs to specify the *what* and let our approach to decide the *how*. For this, activities to be executed need to be selected and ordered (planning problem [9]) considering both control flow and resource constraints (scheduling problem [10]) imposed by the declarative specification. For planning and scheduling (P&S) the activities such that the process objective function is optimized, a constraint-based approach is proposed. Moreover, the proposed approach has been validated through an empirical evaluation considering different test models of varying complexity (cf. Sect. 4).

This paper is organized as follows: Sect. 2 introduces background; Sect. 3 includes an overview of the proposed method including the associated tool support; Sect. 4 shows some experimental results; Sect. 5 presents a critical discussion of the advantages, drawbacks, and some applications of our proposal; Sect. 6 summarizes related work; and finally, Sect. 7 includes some conclusions and future work.

[1] A web-based application for the generation of optimized BP enactment plans from ConDec-R specifications can be accessed at http://regula.lsi.us.es/OptBPPlanner.

2 Background

We use the declarative language ConDec [11] as basis to specify constraint-based BP models (cf. Def. 1), since it allows the specification of BP activities together with the constraints which must be satisfied for correct BP enactment and for the goal to be achieved.

Constraints can be added to a ConDec model to restrict the desired behavior (cf. [11]). ConDec templates, i.e., parameterized graphical representation of constraints, are grouped into:

1. Existence templates: unary relations concerning the number of times one activity is executed, e.g., *exactly*(N,A) specifies that A must be executed exactly N times.
2. Relation templates: positive binary relations used to establish what should be executed, e.g., *precedence*(A,B) specifies that before any execution of activity B at least one execution of activity A must have been done.
3. Negation templates: negative relations used to forbid the execution of activities in specific situations, e.g., *notCoexistence*(A,B) specifies that if B is executed, then A cannot be executed, and vice versa.

Definition 1

A constraint-based process model $S=(\mathrm{A}, C_{\mathrm{BP}})$ consists of a set of activities A and a set of constraints C_{BP} limiting execution behaviors. For each activity $a \in \mathrm{A}$, resource constraints can be specified by associating a role with that activity. The activities of a constraint-based process model can be executed arbitrarily often if not restricted by any constraints.

On the other hand, the area of scheduling [10] includes problems in which it is necessary to determine an enactment plan for a set of activities related by temporal constraints. Moreover, the execution of all activities requires the use of limited capacity resources. In general, the goal consists of finding a feasible plan which satisfies both temporal and resource constraints, optimizing certain objective functions (e.g., minimization of the overall completion time). In a wider perspective, in planning [9], the activities to be executed are not established a priori, hence it is necessary to select them from a set of alternatives and to establish an ordering.

In a related way, constraint programming (CP, i.e., a software technology for modelling and solving problems by using constraints to relate variables) supplies a suitable framework for dealing with P&S problems [12]. To solve a problem through CP, it needs to be modelled as a constraint satisfaction problem (CSP, cf. Def. 2).

Definition 2

A **CSP** $P=(V, D, C_{\mathrm{CSP}})$ is composed of a set of variables V, a domain of values D for each variable in V, and a set of constraints C_{CSP} between variables, so that each constraint represents a relation between a subset of variables and specifies the allowed combinations of values for these variables.

A *solution* to a CSP consists of assigning values to CSP variables, being feasible when the assignments satisfy all the constraints. In CP, global constraints can be defined to improve the modelling of the problems. Similar to CSPs, constraint optimization problems (COPs, cf. Def. 3) require solutions that optimize certain objective functions.

Definition 3

A **COP** Po = (V, D, C_{CSP}, OF) is a CSP which also includes an objective function, OF, to be optimized.

Several mechanisms are available for solving CSPs and COPs, e.g., complete search algorithm, i.e., performing a complete exploration which is based on all possible combinations of assignments of values to the CSP variables. Regardless of the used search method, the global constraints can be implemented through filtering rules (i.e., rules responsible for removing values which do not belong to any solution) to efficiently handle the constraints in the search for solutions.

3 Method for Generating Optimized Enactment Plans

In our approach, two steps can be differentiated: (1) creating a declarative process specification (cf. Fig. 1a) and (2) generating optimized BP enactment plans (cf. Fig. 1b, c).

Creating declarative process specifications. In a first step, a declarative specification covering the control flow, the resource perspective, and the estimates of the BP to be supported is created. As stated, we use the constraint-based language ConDec (cf. Sect. 2) as basis. To plan and schedule the process activities, our proposal extends the ConDec specification by considering (1) estimations for the duration and the role of the required resource of the BP activities and (2) resource availabilities, resulting in a ConDec-R process model (cf. Def. 4). These estimates can be obtained by interviewing business experts or by analyzing past process executions.

Definition 4

A **ConDec-R process model** CR = (Acts, C_{BP}, Res) related to a constraint-based process model S = (A, C_{BP}) (cf. Def. 1) is composed of a set of extended BP activities Acts, which contains tuples (a, role, dur) which includes for each BP activity $a \in$ A the role of the required resource (i.e., role) and the estimated duration (i.e., dur); a set of ConDec constraints C_{BP}; and a set of available resources Res which is composed of tuples (role, #role) which includes for each role (i.e., role) the number #role of available resources.

To develop the OptBPPlanner tool, we have extended Declare [13], which is a workflow management system that can be used to specify ConDec models.

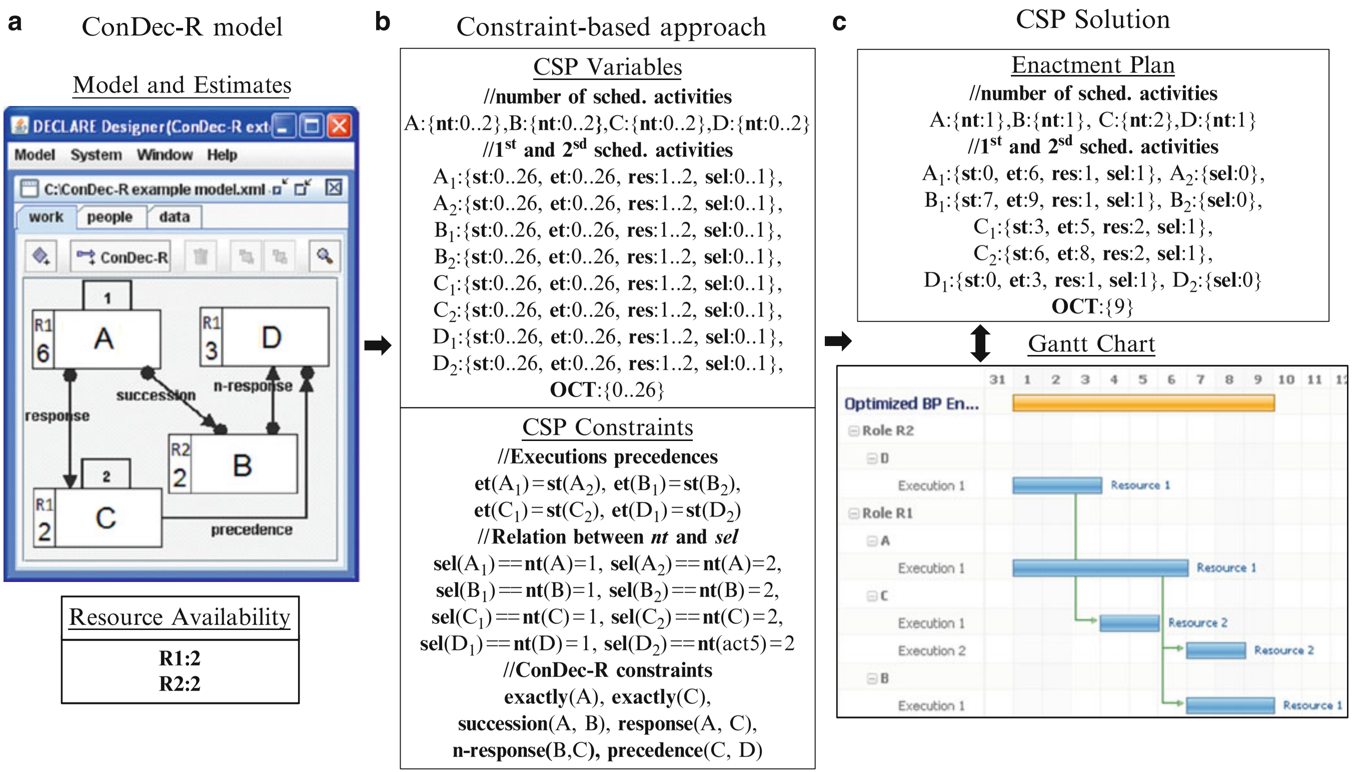


Fig. 1 Overview of our approach

The proposed extension allows the specification of ConDec-R models by including (1) duration and required resource for each BP activity and (2) number of available resources. Figure 1a shows a simple ConDec-R model (created using the aforementioned extension) (cf. Def. 4) where: Acts = {(A,R1,6), (B,R1,2), (C,R2,3), (D,R1,2)}; C_{BP} = {*exactly*(1,A), *exactly*(2,B), *succession*(A,B), *response*(A,B), *negate-response*(B,C), *precedence*(C,D)}; and *Res* = {(R1,2), (R2,2)}.

Generating of optimized BP enactment plans. In our proposal, optimized BP enactment plans are generated by applying a constraint-based approach for P&S the BP activities, taking the ConDec-R specification into account. In the proposed constraint-based approach, BP activities are modelled as repeated activities (cf. Def. 5), which are sequences of optional scheduling activities (cf. Def. 6). This is required since each execution of a BP activity is considered as one single activity which needs to be allocated to a specific resource and temporarily placed in the enactment plan, i.e., stating values for its start and end times.

Definition 5

A repeated activity ra = (*r*, dur, nt) is a BP activity which can be executed several times. It is composed by *r*, which represents the role of the required resource, the estimated duration dur, and a CSP variable nt which specifies the number of times the BP activity is executed.

For each repeated activity, $\mathrm{nt}_{\mathrm{MAX}}$[2] scheduling activities exist, which are added to the CSP problem specification, apart from including a variable nt.

Definition 6

A scheduling activity sa = (st, et, res, sel) represents a specific execution of a repeated activity, where st and et are CSP variables indicating the start and the end times of the activity execution, respectively, res is a CSP variable representing the resource used for the execution, and sel is a CSP variable indicating whether or not the scheduling activity is selected to be executed.

Moreover, to improve the modelling of the problems and to efficiently handle the constraints in the search for solutions, our constraint-based proposal includes a global constraint implemented through a filtering rule (cf. Sect. 2) for each ConDec-R template (cf. Fig. 1b). For a detailed description of these filtering rules, see [14]. In this way, the ConDec-R process model CR = (Acts, C_{BP}, Res) (cf. Def. 4, Fig. 1a) is translated into a COP Po = (V, D, C_{CSP}, OF) (cf. Def. 2, Fig. 1b) where:

1. $V = \{\mathrm{nt}(a) | a \in \mathrm{Acts}\} \cup \{\mathrm{st}(a_i), \mathrm{et}(a_i), \mathrm{sel}(a_i), \mathrm{res}(a_i) | i \in [1 \ldots \mathrm{nt}(a)], a \in \mathrm{Acts}\} \cup \mathrm{OC}$ T. OCT is a CSP variable which represents the overall completion time,[3] i.e., $\mathrm{OCT} = \max_{a \in \mathrm{Acts}}(\mathrm{et}(a_{\mathrm{nt}(a)}))$

---

[2] $\mathrm{nt}_{\mathrm{MAX}}$ represents the maximum value of the initial domain of nt (cf. Fig. 1b).

[3] The overall completion time is the time needed to complete all process instances which were planned for a certain period.

2. D is composed of the domains of each CSP variable. The domain [0…2] is used for nt since 2 is the maximum cardinality for the BP activities (established by existence relations in the constraint-based model). The domain [0…26] is used for et and st since 26 would be the completion time if all the scheduling activities were serially executed taking the maximum cardinality for the BP activities into account.
3. C_{CSP} is composed of the global constraints related to C_{BP} together with the constraints which are inherent to the proposed model:
   - $\forall a \in \mathrm{Acts}, \forall i : 1 \leq i \leq \mathrm{nt}(a) : \mathrm{et}(a_i) \leq \mathrm{st}(a_{i+1})$ (i.e., a specific execution of a repeated activity precedes the next execution of the same activity).
   - $\forall a \in \mathrm{Acts}, \forall i : 1 \leq i \leq \mathrm{nt}(a) : \mathrm{sel}(a_i) = \mathrm{nt}(a_{i+1}) \geq i$ (i.e., the nt variable of the repeated activity is directly related to the sel variable of this associated scheduling activity).
4. OF = OCT.

For the current approach, to solve the constraint-based problems, the COMET system [15] is used, since it is able to generate high-quality solutions for highly constrained problems in an efficient way. This system provides a scheduling module that offers high-level constraint modelling and search abstraction, both specific to scheduling. The COP related to the ConDec-R specification is considered as a scheduling problem (cf. Fig. 1b) to take advantage of the efficient COMET mechanisms and high-level modelling. The optimized BP enactment plan is then created from the CSP solution (cf. Fig. 1c) and is composed of (1) the number of times each BP activity is executed, (2) the start and the completion times for each activity execution, and (3) the resource which is used for each activity execution. The generated enactment plans can be graphically represented by a Gantt chart [16] (cf. Fig. 1c)[4]. This chart illustrates the activity schedules and allows users to understand the solution at a glance. Moreover, the relations between executions of activities are depicted in the Gantt chart due to the ConDec-R constraints of the model (e.g., the relation between the first execution of D and the first execution of the C is due to the constraint *precedence*(C,D)).

Since the generation of optimal plans for these types of problems presents NP complexity [17], it is not possible to ensure the optimality of the generated plans for all cases. The developed constraint-based approach, however, allows solving the considered problems in an efficient way, as demonstrated in Sect. 4.

4 Empirical Evaluation

To evaluate the suitability of our proposal, a controlled experiment is conducted.

Purpose: The purpose of the empirical evaluation is to analyze our proposal in the generation of optimized BP enactment plans from ConDec-R models.

[4] The generated Gantt chart of Fig. 1c groups activities by roles, e.g., the *Execution1* of D is performed by the *Resource 1* of the *Role R2*. The rest of activities are performed by *Role R1*.

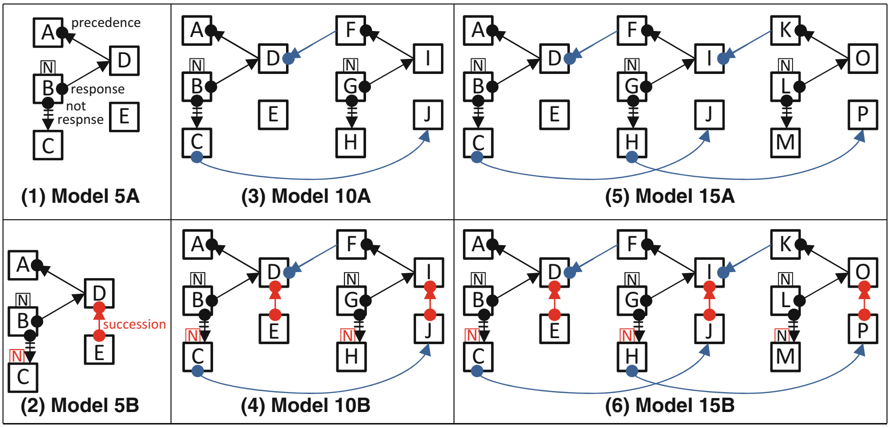


Fig. 2 Generic ConDec models

Objects: Different ConDec models are generated by considering correctness and representativeness. Consequently, we require the test models to be of medium size (i.e., including 5–15 BP activities which can be executed {15, 30, 60} times[5]) and comprise all three types of ConDec templates (cf. Sect. 2, cf. Fig. 2): (1) Model 5A includes 5 BP activities and few constraints, (2) Model 5B extends the Model 5A by including more constraints, (3) Model 10A includes 10 BP activities and few constraints, (4) Model 10B extends the Model 10A by including more constraints, (5) Model 15A includes 15 BP activities and few constraints, and (6) Model 15B extends the Model 15A by including more constraints. Moreover, in the case of *existence* constraints, a value for label *N* is established. In addition, different durations and required resources for each BP activity are considered, since these aspects have a great influence in the complexity of the search of optimal solutions. Specifically, 30 instances for each problem are randomly generated by varying activity durations between 1 and 40 and the role required between R1 and R2. In addition, two available resources for R1 and R2 are considered.

Furthermore, different time limits for the search algorithm are considered to show the applicability in scenarios with different response time requirements.[6]

Independent Variables: For the empirical evaluation, (1) M, i.e., the generic ConDec model, with the values {M5A, M5B, M10A, M10B, M15A, M15B}; (2) TL, i.e., the time limit (seconds) to find an optimal solution, with the values {5, 50, 300}; and (3) *N*, i.e., the value for the label *N* of the *existence* constraints in the models, with the values {15, 30, 60} are considered.

[5] These values are considered to analyze the behavior of our proposal when dealing with problems of different size, i.e., with different number of repetitions of certain activities.

[6] The set of problems which are used are available at http://regula.lsi.us.es/ISD12/EV.zip.

Table 1 Average values related to the experimental executions

Indep. var.			Resp. var. (TL = 5)			Resp. var. (TL = 50)			Resp. var. (TL = 300)		
M	*N*	Nact	OCT	%Opt	Topt	OCT	%Opt	Topt	OCT	%Opt	Topt
M5A	15	17	334.8	100.0	0.28	334.8	100.0	0.28	334.8	100.0	0.28
M5A	30	32	728.1	83.3	1.70	728.1	100.0	2.77	728.1	100.0	2.77
M5A	60	62	1,478.7	93.3	0.69	1,478.7	93.3	20.00	1,478.7	93.3	20.00
M5B	15	33	600.6	100.0	0.08	600.6	100.0	0.08	600.6	100.0	0.08
M5B	30	63	1,477.0	100.0	0.29	1,477	100.0	0.29	1,477.0	100.0	0.29
M5B	60	123	2,626.0	100.0	2.30	2,626	100.0	2.30	2,626.0	100.0	2.30
M10A	15	34	471.4	40.0	0.41	471.4	43.3	2.38	471.4	50.0	14.71
M10A	30	64	894.4	50.0	1.28	894.4	50.0	1.28	894.4	50.0	1.28
M10A	60	124	1,698.7	6.7	3.47	1,698.7	36.7	16.53	1,698.7	43.3	35.34
M10B	15	66	789.3	33.3	0.15	788.8	33.3	0.15	788.8	36.7	8.48
M10B	30	126	1,495.0	40.0	0.70	1,494.2	43.3	2.52	1,494.2	43.3	2.52
M10B	60	246	2,896.5	16.7	3.83	2,895.1	16.7	3.83	2,894.2	16.7	3.83
M15A	15	52	563.6	13.3	0.28	563.6	13.3	0.28	563.2	13.3	0.28
M15A	30	97	1,027.2	16.7	2.08	1,027.2	16.7	2.08	1,027.2	20.0	32.29
M15A	60	187	2,016.0	6.7	4.43	2,016.0	23.3	16.51	2,016.0	23.3	16.51
M15B	15	99	1,071.4	13.3	0.95	1,070.9	13.3	0.95	1,070.6	20.0	45.55
M15B	30	189	1,879.7	40.0	1.70	1,875.8	43.3	4.25	1,875.8	43.3	4.25
M15B	60	369	3,972.0	0	–	3,934.5	20.0	11.19	3,923.7	23.3	36.94

Response Variables: The suitability of our approach is tested regarding (1) the average value of the objective function which is obtained (i.e., OCT), (2) the average percentage of optimal solutions which are found (i.e., %Opt), and (3) the average time for getting the optimal solution, considering the cases in which an optimal solution is found (i.e., Topt).

Experimental Design: 540 instances are generated by considering different values for *M* (6 values), *N* (3 values), and the random generation of durations and required resources (30 problem instances). For each instance a complete search (cf. Sect. 2) is executed to optimize the OCT considering the three different values of TL. The response variables are then calculated by considering the average values of the 30 problem instances.

Experimental Execution: The constraint-based search algorithm is run on an Intel(R) Xeon(R) CPU E5530, 2.4 GHz, 8 GB memory, running Debian 6.0.3. The system COMET [15] is used to solve the developed constraint-based problems.

Experimental Result and Data Analysis: Table 1 shows for each problem (specified by *M*, *N*, and TL), the average values of the response variables (i.e., OCT, %Opt, and Topt) for the 30 problem instances. In addition, the number of scheduling activities (cf. Def. 6) which are executed (i.e., Nact) are shown for each pair $\langle M,N \rangle$.

As expected, %Opt decreases as the number of BP activities and/or scheduling activities increases (i.e., complexity of the problem). As TL increases %Opt increases and OCT decreases but not as significantly since the best values of OCT are achieved in a short time but it was not possible to ensure their optimality. Moreover, Topt increases as the complexity of the problems increases. In general, experimental results show that despite NP complexity of the considered problems, the values for the percentage of optimal solutions found and for the average time for getting optimums are quite good for small- and medium-sized problems (between 17 and 189 scheduling activities[7]). Hence, the approach becomes suitable for run-time applications (e.g., recommendations) and for scenarios where high quality is required.

5 Discussion

One advantage of our proposal is that the optimized BP enactment plans are generated by P&S all BP activities, allowing for a global optimization of the objective function. Moreover, the generation of the optimized plans is carried out through a constraint-based approach, which is suitable for modelling and solving P&S problems [12]. In addition, this approach allows modelling the considered problems in an easy way, since the considered specifications are based on high-level constraints. Furthermore, BPs are specified in a declarative way, which facilitates the human work involved and avoids failures [18]. Moreover, our approach, as extension of other similar works, considers the resource perspective besides the control-flow perspective; hence greater optimization can be obtained.

On the other hand, the proposed approach presents some drawbacks. First, the most important limitations are the assumptions that are made, i.e., the optimized plans are generated by considering estimated values for activity durations and resource availabilities; hence our proposal is only appropriate for processes for which the duration of the activities and resource availabilities can be estimated. However, to consider deviations in the estimates, the optimized plans can be updated—if necessary—through replanning, allowing to react to changes in a quick and flexible way. Secondly, the business analysts must deal with a not standard language for the specification of BPs, therefore a period of training is required. Moreover, the considered declarative specifications deal with both control-flow and resource perspectives, but do not consider the data perspective. It is intended to consider this aspect in future works. In our proposal we focus on the minimization of overall completion time. However, it can be easily extended to consider further objectives, such as cost.

5.1 Applications of Optimized BP Enactment Plans

Simulation: Simulation of BPs can be effectively used for analyzing processes and for improving BP models. BP simulation presents a “fast-forward” view on a

[7] Note that getting the optimum for scheduling problems of 189 activities can entail a great complexity. In fact, there are many scheduling benchmarks of smaller size for which their optimal values are not even known.

current BP, so that the generated simulation models can accurately reflect the real-world process of interest. One interesting application of BP simulation is to identify unbalances between the resources required for executing a particular process and the available resources [19]. Moreover, the effects of alternative resource schedules can be investigated. Our proposal can be used in *what-if* scenarios to evaluate the impact of changing something in the declarative BP. For example, in Fig. 1a, if we reduce the resource availability of B to 1, the generated plan is still valid (i.e., we can do the same considering fewer resources); however, if we reduce R1 to 1, A, B, and C cannot be executed in parallel, then the enactment plan changes. Thus, the results can be studied to analyze and to enhance the current BP model (process design and analysis phase of BPM life cycle [1]).

Time Prediction: There are many scenarios where it is useful to have reliable time predictions [5]. In the current approach, the generated optimized BP enactment plans can be used for this purpose since time information is available based on the estimated durations of the activities. For a given process instance state, the expected completion time for the instance and activities can be calculated by taking the end time of the remaining activities of the optimized plan into account. In this way, the BP enactment plans of the BP model can be used for predicting the completion time of running instances and activities, and hence improving the process design and analysis phase of BPM life cycle [1].

Recommendations: The application of BP enactment plan for generating recommendations is detailed in a previous work [6]. In the current dynamic business world, the economic success of an enterprise increasingly depends on its ability to react to changes in its environment in a quick and flexible way [20]. Therefore, flexible BPM systems are required to allow companies to rapidly adjust their BPs to changes in the environment [21]. In general, increasing flexibility in BPM systems tends to result in decreased user support [22] requiring more experienced users. Typically, given a certain partial trace users can choose from several enabled activities (i.e., activities whose execution does not violate any constraint or only lead to temporary violations [23]) which activity to execute next. This selection, however, can be quite challenging since objective functions of the process should be considered, and users often do not have an understanding of the overall process. Moreover, optimization of objective functions requires that resource capacities are considered. Therefore, recommendation support is needed during BP execution, especially for inexperienced users. As an application of the generated optimized BP enactment plans, recommendations which assists users during process enactment to optimize objective functions of the processes can be generated, hence enhancing the process enactment phase of BPM life cycle [1].

The Generation of Optimized BP Models: BP models are usually defined manually by business analysts through imperative languages. To this end, the analysts must deal with several aspects, such as resource allocation, activity properties, the relations between them, and, in most cases, even the optimization of several objectives. Therefore, the manual specification of BP models can form a very complex problem, can consume a great quantity of time and human resources, may cause some failures, and may lead to non-optimized models. To overcome these problems, taking the information of the optimized BP enactment plans and the constraints of the declarative BP model into account, BP models can be generated, therefore improving process design and analysis phase of BPM life cycle [1].

6 Related Work

There exist some proposals which could be used to generate optimized enactment plans for BPs from constraint-based specifications. Specifically [24] proposes the generation of an automaton from constraint-based specifications based on linear temporal logic (LTL) which represents exactly all traces that satisfy the LTL formulas. When extending this approach by including estimates, the overall completion time of all the traces could then be calculated (e.g., [5]). However, the big disadvantage following such an approach would be that it comes to a state explosion since all the LTL formulas have to be concatenated to build a big automaton [24], and, unlike the proposed approach, no heuristic has been used. In a similar way, CLIMB [25] could be used to generate quality traces from declarative specifications and calculate its completion time. Then, the best traces could be selected. Unlike the proposed approach [25], does neither consider optimality nor resource availabilities. Therefore, this would only cover the planning part of the current proposal, but not the scheduling aspects addressed by our approach.

There are additionally some proposals related to generating imperative BP models [7, 8, 26], giving recommendations [27], simulation, [28] and time prediction [5]. However, unlike in our proposal, the process optimization and the resource allocation are not considered.

7 Conclusions and Future Work

As a major contribution of this work, we propose a method for the automatic generation of optimized BP enactment plans from declarative specifications which assists users during different stages of the BPM life cycle, i.e., BP design and analysis and enactment stages, to optimize objective functions of the processes (i.e., minimization of overall completion time). The proposed method is based on a constraint-based approach for planning and scheduling the BP activities and considers both the control flow and the resource perspective. To demonstrate the feasibility of our approach, a tool, called OptBPPlanner, has been implemented. As for future work, it is intended to extend the proposed approach by considering further objective functions. Moreover, we will explore various constraint-based solving techniques and analyze their suitability for the generation of optimized plans.

Acknowledgments This work has been partially funded by the Spanish Ministerio de Ciencia e Innovación (TIN2009-13714) and the European Regional Development Fund (ERDF/FEDER).

References

1. Weske M (2007) Business process management: concepts, methods, technology. Springer, Berlin
2. van der Aalst W, ter Hofstede A, Weske M (2003) Business process management: a survey. Proc. business process management: International Conference, BPM 2003, pp 1–12
3. Group G (2010) Leading in times of transition: the 2010 CIO agenda (EXP Premier Report No. January 2010), in Report, Gartner, Inc.
4. Rozinat A, Wynn M, van der Aalst W, ter Hofstede A, Fidge C (2009) Workflow simulation for operational decision support. Data Knowl Eng 68(9):834–850
5. van der Aalst W, Schonenberg M, Song M (2011) Time prediction based on process mining. Inf Syst 36(2):450–475
6. Barba I, Weber B, Del Valle C (2011) Supporting the optimized execution of business processes through recommendations. In Proc. BPM Workshops, pp 135–140
7. González-Ferrer A, Fernández-Olivares J, Castillo L (2009) JABBAH: a java application framework for the translation between business process models and HTN. In: Proc. ICKEPS
8. R-Moreno M, Borrajo D, Cesta A, Oddi A (2007) Integrating planning and scheduling in workflow domains. Expert Syst Appl 33(2):389–406
9. Ghallab M, Nau D, Traverso P (2004) Automated planning: theory and practice. Morgan Kaufmann, Amsterdam
10. Brucker P, Knust S (2006) Complex scheduling (GOR-Publications). Springer-Verlag New York, Inc., Secaucus, NJ
11. Pesic M, van der Aalst W (2006) A declarative approach for flexible business processes management. In: Proc. BPM Workshops, pp 169–180
12. Salido M (2010) Introduction to planning, scheduling and constraint satisfaction. J Intell Manuf 21(1):1–4
13. Declare (2011) Declarative approach to workflow management systems. http://www.win.tue.nl/declare/. Online; Accessed 4 May 2012
14. Barba I, Del Valle C (2011) Filtering rules for ConDec templates—pseudocode and complexity. http://regula.lsi.us.es/ISD12/FilteringRulesforConDecTemplates.pdf
15. Dynadec. Comet. http://dynadec.com/support/downloads/. Online; Accessed 4 May 2012
16. Gantt H (1913) Work, wages, and profits. Engineering Magazine Co., New York
17. Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman & Co., New York, NY
18. Rychkova I, Regev G, Wegmann A (2008) High-level design and analysis of business processes: the advantages of declarative specifications. In: Proc. RCIS, pp 99–110
19. Reijers H, van der Aalst W (1999) Short-term simulation: bridging the gap between operational control and strategic decision making. In: Proc. IASTED conference on modeling and simulation, pp 417–421
20. Poppendieck M, Poppendieck T (2006) Implementing lean software development: from concept to cash. Addison-Wesley, Boston, MA
21. van der Aalst W, Jablonski S (2000) Dealing with workflow change: identification of issues and solutions. Int J Comput Syst Sci Eng 15(5):267–276
22. van Dongen B, van der Aalst W (2005) A meta model for process mining data. In: Proc. Caise, pp 309–320
23. Pesic M, Schonenberg M, Sidorova N, van der Aalst W (2007) Constraint-based workflow models: change made easy. In: OTM Conferences (1), pp 77–94

24. Pesic M (2008) Constraint-based workflow management systems: shifting control to users. Ph.D. thesis, Eindhoven University of Technology, Eindhoven
25. Montali M (2009) Specification and verification of declarative open interaction models: a logic-based approach. Ph.D. thesis, Department of Electronics, Computer Science and Telecommunications Engineering, University of Bologna
26. Hoffmann J, Weber I, Kraft F (2010) SAP speaks PDDL. In: Proc. AAAI, pp. 1096–1101
27. Schonenberg H, Weber B, van Dongen B, van der Aalst W (2008) Supporting flexible processes through recommendations based on history. In: Proc. BPM, pp 51–66
28. Reijers H (2003) Design and control of workflow processes: business process management for the service industry, vol 2617, LNCS. Springer, Berlin

Modelling Life Cycles of Generic Object Classes

Vaclav Repa

Abstract The paper analyses the problem of class generalisation from the non-traditional point of view—class life cycles. Model of class life cycle represents a process-oriented view on the class which is not usual in the field of conceptual modelling. This paper focuses on the problem of modelling life cycles of generic classes when there is a need to model several life cycles, valid at the same time. The paper shows that this problem is rooted in the natural contradiction of object- and process-oriented approach to modelling following from the fact that these two basic approaches are based on mutually contradicting primary types of hierarchical abstraction. The paper also shows that this problem is closely connected with the "problem of conflicting identities" in generalisation trees discussed on the border of the conceptual modelling and ontology engineering fields.

1 Introduction: Historical Background

The latest development of the *enterprise architecture* methodologies is strongly influenced by the need for modelling the real-world aspects of an enterprise as a universal base for the further design. The basic technique for modelling the real world, representing the basic approach to the analytical thinking in the field of informatics, is the *conceptual modelling*. The concept "conceptual" has been firstly used in the area of data modelling. It expresses the fact that the database should describe the essential characteristics of the real world: objects and their mutual relationships. This origin is still visible in common understanding of the adjective "conceptual" in the sense of modelling with the Unified Modelling Language [10]:

V. Repa (✉)
Department of Information Technology, University of Economics, Prague,
W. Churchill sqr. 4, Prague 3, 130 67, Czech Republic
e-mail: repa@vse.cz

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_34,

Craig Larman [8] describes conceptual modelling such as the following: classes represent concepts from the real-world domain, binary associations describe relationships between two concepts, and the concepts can have attributes but no operations. Cris Kobryn [6] speaks about "structural model" which shows the static structure of the system: the entities that exist, internal structure, and relationships to other entities. Roni Weisman [12] uses the term "conceptual system model" and distinguishes three types of objects: entity (objects which hold the system's data), boundary object (interface objects which directly interact with the external world—actors), and control object (objects which manage the system operations). Although there are various approaches to the conceptual modelling in object-oriented methods, each of them reduces the conceptual model (represented by the class diagram) to the model of objects and relationships between them, represented by their attributes but not by their methods. This reduction is present even in Weisman's approach (see above) where only "entities" represent the real-world objects while "control object" expresses rather the behaviour of the "system". The common understanding of the term "conceptual" thus tends to be the synonym for "static".

However, such an approach contrasts with the basic principle and the main contribution of the object-oriented paradigm—unity of data and operations. This principle evokes the idea that it is necessary to model not only static aspects of the real world but also its dynamics. Thus, not only attributes but also relevant object's operations together with their essential time consequences should be regarded as a property of the real world. Such description of the essential dynamics of the object is usually called the object's life cycle.

The interest in object life cycles originated in the historical context of business process modelling in the early 1990s. According to Kappel and Schrefl [5], *an object life cycle is a model that captures allowed states and state transitions for a particular object type*. A common means for modelling life cycles of objects is a non-deterministic finite state machine. The UML contains for this purpose the state chart (state machine) diagram. Other UML diagrams for description of dynamics (activity diagram, sequence diagram, objects interaction diagram) are also usable but not so suitable for this purpose because they are not so closely and explicitly connected with the class diagram as a crucial diagram for the conceptual modelling (see the further argumentation below).

The historical context of the business process modelling, mentioned above, also brought the general, still persisting, erroneous impression that business processes can be modelled via life cycles of participating objects. Küster et al. [7] clearly identified the substantial difference between an object life cycle and a business process formulating the set of rules for "generation of a compliant business process model from given object life cycles". Although this approach views the business process too mechanically, almost as a mechanical consequence of business rules expressed by object life cycles, the identified difference is a very significant shift in the perception of essence of object life cycles.

The motivation for modelling life cycles of classes presented in this paper comes from the methodology for information modelling of organisations [9]. This methodology is based on systematic work with two parallel model dimensions, conceptual (object oriented) and behavioural (process oriented), and distinguishes between the two main types of processes which are to be described in order to fully define the nature of the real world:

- Business (intentional) processes
- Life cycle (non-intentional, substantial) processes

The *business process* always represents some intention, expresses the way of achieving some goal, and has some products, and it is typical expression of the human will.

On the other hand, the *life cycle of an object* has no goal, nor product; it is rather the expression of the objective necessity, usually called business rules. Objects are typically taking different roles in different processes giving them the context (real-world rules), while business process typically combines different objects giving them the specific meaning (roles of actors, products, etc.).

As it follows from the previous paragraph, the objects life cycles cannot be regarded as business processes but rather as a *description of business rules in a process manner*. The presented methodology uses for the modelling objects life cycles the state chart from the UML which is, according to the UML metamodel [11], principally connected with the class diagram via the concepts of class, method, and some others. This connection is also well visible in the Business Substance Metamodel from the Business System Metamodel [1] as a part of the OpenSoul Methodology where the UML metamodel is extended with the concepts of class state, class life cycle step, and class life cycle which address exactly this methodical consequence.

The example in Fig. 1 illustrates the state chart describing the life cycle of the object *order* as a complement to the class diagram which describes the context of

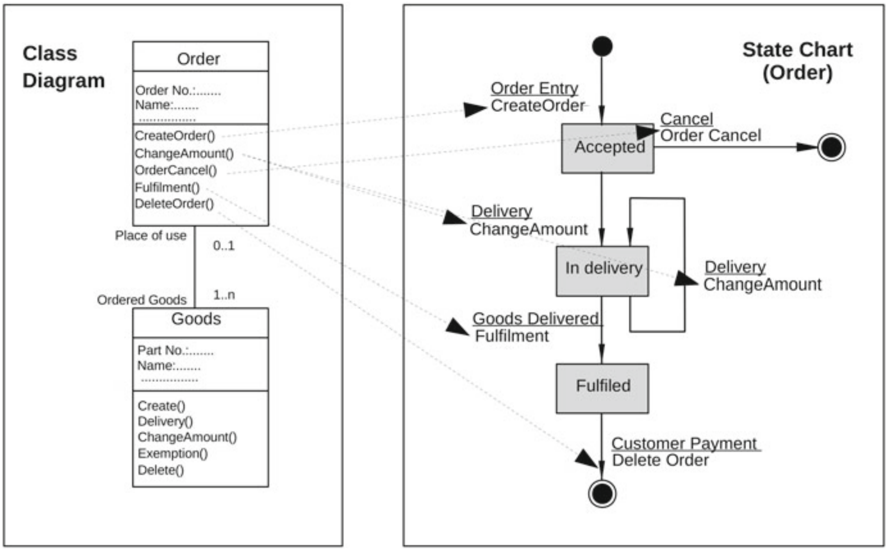


Fig. 1 Life cycle of the class

this object. All methods of the conceptual object should be ordered into one algorithm which defines the “conceptual meaning” of each method as its place in the process of the object’s life.

Generalisation—a natural way of building the hierarchy of concepts—is one of the most significant concepts in the field of conceptual modelling. This way of organising concepts is also used as a basic tool for the top-down process of the analysis of concepts. On the higher level of abstraction, we work with global—generic concepts and their general associations, then on more detailed level we can analyse detailed—more specific types of previously analysed concepts and their specific relationships. Generalisation as an essential way of building hierarchies of concepts forms the roots of conceptual (alias “object oriented”) modelling languages. In the UML the generalisation occurs as one of basic principles hidden in the so-called principle of inheritance.

Combining the above discussed need for modelling the dynamics of objects together with the importance of generalisation as a natural way of building the hierarchy of concepts, we can formulate the crucial question: *how to model life cycles of generic classes*?

In the following text, we will discuss the problem of generalisation of classes from a non-traditional (class life cycles) point of view. We will focus on the problem of modelling life cycles of generic classes when there is a need to model the life of the generic class together with the life of its specific subtype at the same time. This problem is rooted in the natural contradiction of object- and process-oriented approach to modelling following from the fact that these two basic approaches are based on mutually contradicting primary types of hierarchical abstraction. We also show that this problem is closely connected with the “problem of conflicting identities” in generalisation trees discussed in several works in the fields of conceptual modelling as well as ontology engineering.

In the first section, we will pay attention to the problem of using generalisation in conceptual models. We will formulate the problem of modelling life cycles of different concepts representing the same object in the generalisation tree, and we show that this problem has the common root with the language insufficiencies discussed before. In the last section, we will summarise previous sections and formulate basic conclusions.

2 Generalisation from the System Versus Process Point of View

Generalisation as a principal way of hierarchical classification of concepts is widely used in the conceptual modelling based on modal logics. Giancarlo Guizzardi [3] convincingly shows the necessity of the classification of different types of specialisation of concepts which occurs especially in the case of subtypes of the so-called <<phase>> type (see the example in Fig. 2). The problem is that the particular instance of the class *customer* can be over time of different subtypes *potential*,

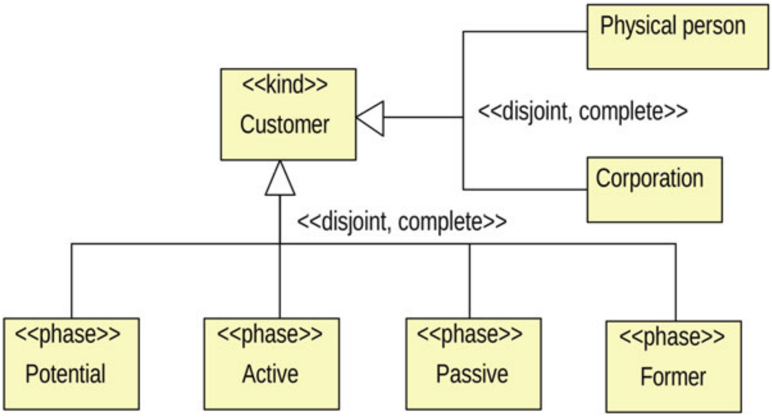


Fig. 2 Ontologically correct multiple specialisation of the class customer

active, *passive*, and *former* staying the same individual. Thus, these subtypes do not represent different instances of the class *customer* but rather different states (phases) of the same instance. Guizzardi also notes that the same problem is addressed by Heller and Herre in [4] who, in generalisations, distinguish between abstract substances called *persistents* and so-called presentials which represent momentary states.

In this example we pragmatically reduce our universe of discourse ignoring the general fact that the *customer* is rather the role of some general concept (person for instance) instead of the standalone kind as it is regarded here. In our real world, the *customer* is a substantial primary concept and we do not analyse its general super-classifications as it is not relevant there.

Figure 2 shows the nonconflicting multiple specialisation of the concept *customer*. From the point of view of the natural evolution of general *customer*, it is important to distinguish among its particular phases *potential*, *active*, *passive*, and *former* which the *customer* generally can pass over time. Nevertheless, it is also important to distinguish between two general types of *customer*: *physical person* and *corporation*, simultaneously. As the second classification (*corporation*/*physical person*) is not transitive, i.e. the particular instance of customer cannot transit from one type to another during its life, there is no danger of possible conflict of this classification with the first one (evolution phases of *customer*). Both classifications can exist simultaneously without mutual interference because they are independent in principle. The mechanism of evolution of a *customer* is exactly the same whether it is a *physical person* or *corporation*, and *customer* is either *physical person* or *corporation* no matter in which evolution stage it is.

If there is a need for time dependent—<<*phase*>> specialisation of a class—it can be supposed that there is also a need for detailed description of general conditions and circumstances for the transitions between its particular phases. Such description can be done as a so-called life cycle of a class where above mentioned conditions and other general circumstances are described as a process of possible transitions among identified states of a class.

While modelling life cycles of both generic classes and their specific subtypes, we always need to handle the following problem: we should express the common structure expressing both generalisation and aggregation. This problem is also addressed by Ebert and Engels in [2] where they pointed to the need to reconcile the conceptual modelling with modelling of life cycles in the case of generic objects. It is rooted in the natural contradiction of object- and process-oriented approach to modelling. This contradiction follows from the fact that each of these two basic approaches have a different primary type of hierarchical abstraction. In the object-oriented approach, hierarchy means primarily generalisation; while in the process-oriented approach to modelling, hierarchy means primarily aggregation. As these two basic types of hierarchical abstraction are mutually exclusive (the hierarchy can express either generalisation or aggregation), object- and process-oriented approach to modelling are in principal contradiction. Practically, this means that it is impossible to express all crucial process aspects of a reality by means of objects only (alias to model processes as objects and their relationships) as well as it is impossible to express all crucial object aspects of a reality just by means of processes (alias to model objects as processes and their successions).

In the object-oriented approach, the generalisation exists as a principle (so-called inheritance principle), and the second possible type of hierarchical abstraction—the aggregation plays the role of just a specific type of relationships of objects. In the process-oriented approach, the aggregation exists as a principle (the rule that "any activity as a part of a process can be taken as a standalone process on a deeper level of detail"), and generalisation plays a secondary role of just a specific kind of relationships of different, mutually exclusive, activities. Above discussed facts mean that to express the life cycles of both generic and specific classes of the same generalisation tree we have to describe the generalisation tree as a set (i.e. aggregation tree) of processes.

Figure 3 describes the life cycle of the generic class *customer* from Fig. 2. According to the methodology, each phase—specific subtype of the generic class—is represented by the specific state of its life cycle. The life cycle thus describes the general process of the class metamorphosis from one subtype to the others.

The example in Fig. 3 specifies the general mechanism of possible changes of the states (alias subtypes) of the object *customer*. Each transition between states has its reason (real-world event) and is performed by the action which belongs to the class (class method).

If needed, it is possible to describe also specific structure of the life of the object in the specific state as it is illustrated in Fig. 4. This allows expressing the generalisation structure of concepts as an aggregation of life cycles. Nevertheless, the very important fact, and the fatal condition of this situation, is that the whole generalisation is of the <<*phase*>> type (i.e. all subtypes are of this type of specialisation).

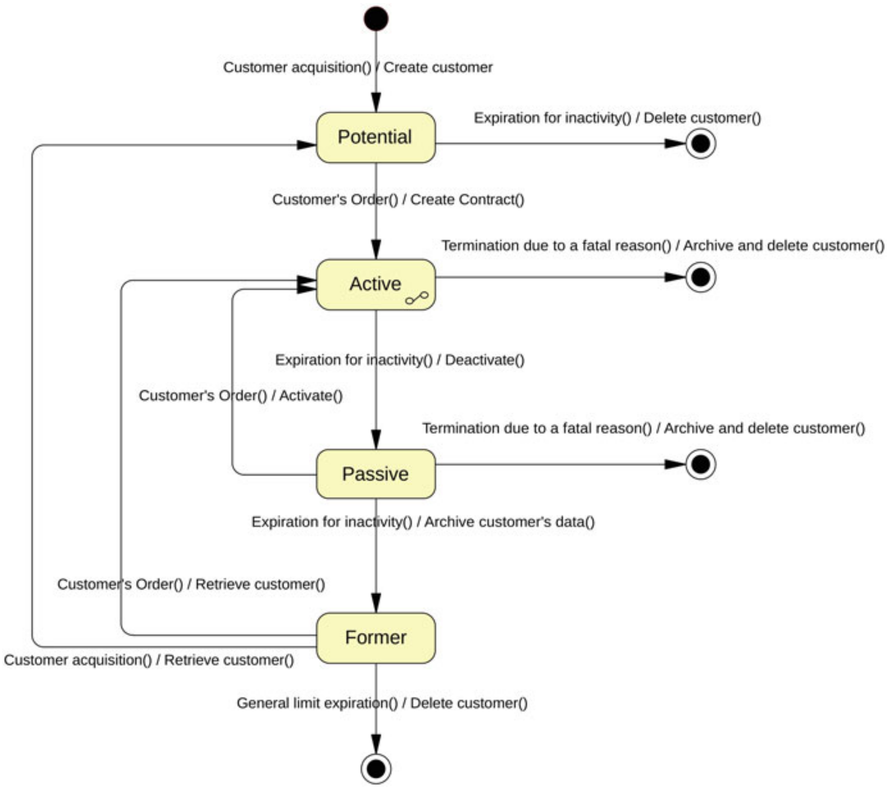


Fig. 3 Life cycle of the generic class customer

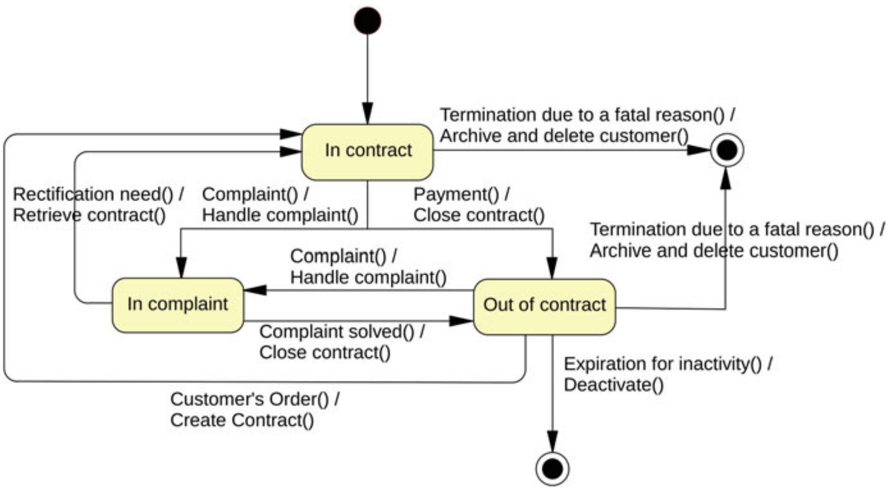


Fig. 4 Life cycle of the specific—active—type of the generic customer

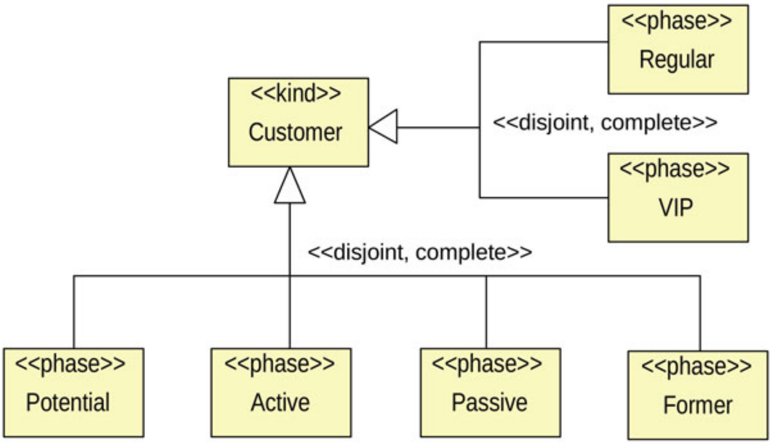


Fig. 5 Conflicting "phase" specialisations of the class customer

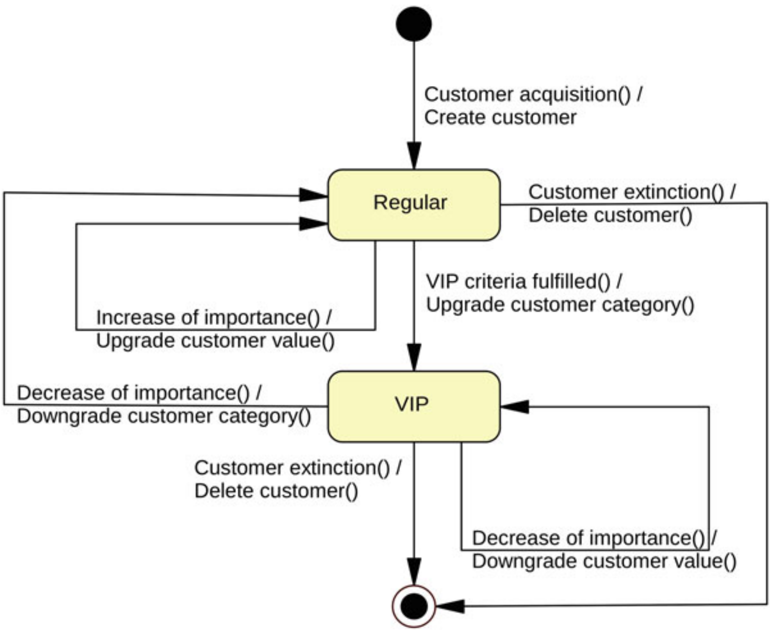


Fig. 6 Life cycle of the class customer from the importance point of view

The second crucial problem connected with describing life cycles of generic object classes occurs in the case of multiple <<*phase*>> generalisation trees for the same class. Such situation is illustrated by the example in Fig. 5.

In this situation there is the additional valid life cycle of the same class customer besides the life cycle described in Fig. 3. This additional life cycle is described in Fig. 6. The problem is that these two life cycles must be valid at the same time as it

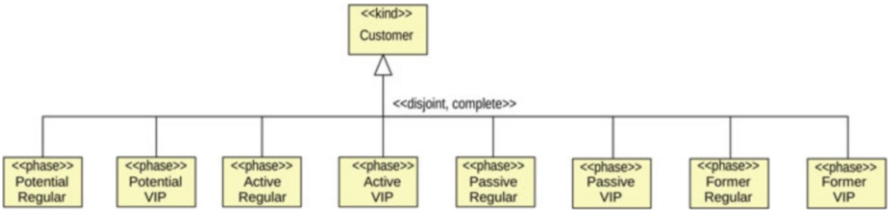


Fig. 7 Possible solution of the conflict of phases via the Cartesian product of possible phases

belongs to the real-life situation like the example illustrates. There are specific combinations of specific states of different types for the same object. At the particular moment, just one specific combination is valid.

The model in Fig. 7 expresses a virtually sufficient solution of the problem of conflicting "phase" specialisations of the class *customer* from Fig. 5. Both classification structures are combined in the Cartesian product and the result is one specific subtype for each combination of each two possibilities. From the global point of view (represented by the class diagram), it seems that the problem is no more existing; the only structure of the subtypes naturally prevents any possible conflict of identity, and the completeness is guaranteed by the combination of all possibilities.

The problem connected with the specialisation in Fig. 7 is that it is principally impossible to express exactly all combined phases as the states of one single life cycle (i.e. in one common state chart). Although from the system point of view (class diagram), each combination of states looks real, when we look at these combined states regarding the factor of time, we can realise that:

(a) Some combinations cannot exist in reality because they do not make sense (e.g. if there is a rule that the customer without an order cannot be *VIP*, the subtypes *Passive VIP* as well as *Potential VIP* are not real).
(b) Some combined states represent rather an aggregation of really elementary states of this type (from the time point of view) than a single state (i.e. in a single moment). This fact causes the possible situation where different states can exist at the same time. For example, the states *Former VIP* and *Former Regular* cannot be clearly distinguished as this single customer could be several times VIP and several times regular in history. If the states cannot be clearly distinguished it is impossible to express their ordering on the time line. Such a situation usually signals that there are more different objects which the states belong to; thus, it is not possible to describe these states as elements of the single algorithm (life cycle).

While the problem (a) is not critical because it can be easily solved by omitting non-relevant states, the problem (b) is fatal. It is a proof that the solution of the conflict of phases via the Cartesian product of the possible phase combinations is principally wrong.

The correct solution of the conflict of phases from Fig. 7 is described in Fig. 8. It is necessary to respect the fact that these two <<*phase*>> specialisations (*potential*, *active*, *passive*, *former* versus *regular*, *VIP*) are independent in principle and cannot

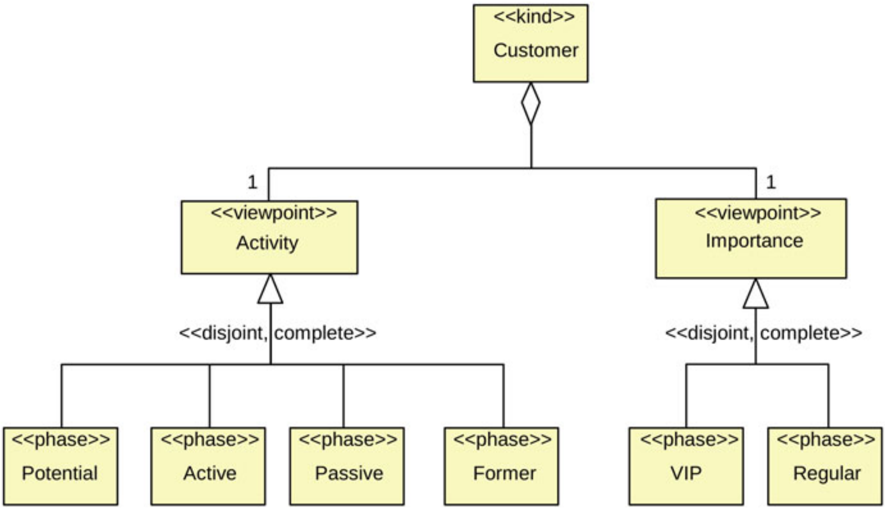


Fig. 8 Generally correct solution of the conflict of phases with <<*viewpoint*>> decomposition

thus be combined. Therefore, we introduce abstract concepts *activity* and *importance* which represent two different points of view which are, in fact, represented by these two independent classifications. Each new generic concept represents the part of the super-concept *customer* from its point of view.

For this general-methodical solution of the problem of multiple <<*phase*>> generalisation trees, we introduce a new type of abstract class <<*viewpoint*>> existing under following rules:

(a) <<*viewpoint*>> must be an abstract generic class representing specialisation tree of the <<*phase*>> type.
(b) <<*viewpoint*>> class never represents the real object.
(c) <<*viewpoint*>> must be a part of an aggregation structure. This aggregation represents an abstract class which is a generic concept representing all phases of all its components, and which:

   - Cannot be of the <<phase>> type
   - Always represents the real object

The purpose of this methodological construction is to allow expressing two crucial facts:

- The abstract class which is a head of the structure (the concept *customer* in the example) is specified with multiple life cycles which are mutually independent.
- All the specified life cycles of the class are valid at the same time.

This way the problem is definitely solved. Each of the both life cycles is guaranteed valid (independently of the second one), and both are valid at the same time as it is warranted by the aggregation of both viewpoints.

3 Summary and Conclusion

In this paper we have discussed the problem of modelling life cycles of generic classes. We have identified two basic types of problems which can occur:

- The problem of aggregated life cycles if there is a need to model the life of the generic class together with the life of its specific subtype at the same time
- The problem of multiple <<*phase*>> generalisation trees for the same class if there is a need to model the life of the generic class from different points of view which all are valid at the same time

We have shown that both crucial problems have a common root—the natural contradiction of two basic types of hierarchical abstraction. This root manifests itself also in the way of solution of both above stated problems: in both cases the solution lies in the use of the combination of both abstraction types together.

The general conclusion made from this paper is as follows: generalisation, unlike the aggregation, leads to the need to principally distinguish between the meta-concepts "object" and "concept". This difference is not always primarily visible at the global level (i.e. at the level of the system of objects). Nevertheless, it is well visible at the detail level (i.e. at the level of the object life cycle) especially in two basic situations:

- In the situation where it is a need for expressing the life cycle of the generic concept together with the life cycles of its specific subtypes
- In the situation where it is a need for expressing several different and mutually incompatible life cycles valid for the same class at the same time

This fact can be used as an additional and "physical" reason for distinguishing between the above mentioned types of generic objects. Consequently, it can be used as a proof of the general validity of the need for distinguishing between substantial and temporally dependent types of generalisation.

Acknowledgements The work presented in this paper has been supported by the Czech Science Foundation in the grant project No. P403/10/0303 Enterprise Architecture as Management Principle for SMEs.

References

1. Business Substance Metamodel. http://opensoul.panrepa.org/bsm.html
2. Ebert J, Engels G (1997). Specialization of object life cycle definitions. Fachberichte Informatik 19/95, University of Koblenz-Landau
3. Guizzardi G (2005) Ontological foundations for structural conceptual models. Telematica Instituut Fundamental Research Series No. 15, ISBN 90-75176-81-3, ISSN 1388-1795
4. Heller B, Herre H (2004) Ontological categories in GOL. Axiomathes 14: 71–90 Kluwer Academic
5. Kappel G, Schrefl M (1991) Object/behavior diagrams. In: Proceedings of the 7th international conference on data engineering. IEEE Computer Society, Los Alamitos, pp 530–539

6. Kobryn C (2000) Introduction to UML: structural modelling and use cases. Object modelling with OMG – UML tutorial series. http://www.omg.org
7. Küster JM, Ryndina K, Gall H (2007) Generation of business process models for object life cycle compliance. In: Proceedings of BPM 2007 international conference, LNCS 4714. Springer, Berlin, pp. 165–181
8. Larman C (2002) Applying UML and patterns: an introduction to object-oriented analysis and design and the unified process, 2nd edn. Prentice Hall, Upper Saddle River, NJ. ISBN 0-13-092569-1
9. Repa V (2003) Business system modeling specification. In: Proceedings of the CCCT2003 international conference, IIIS, Orlando, FL
10. Object Management Group (2003) UML OMG unified modeling language specification, v. 1.5. Document ad/03-03-01
11. Object Management Group (2004) UML Superstructure specification, v2.0 document 05-07-04
12. Weisman R (1999) Introduction to UML based SW development process. http://www.softera.com

It Is About Time: Investigating the Temporal Parameters of Decision-Making in Agile Teams

Niamh O. Riordan, Tom Acton, Kieran Conboy, and Willie Golden

Abstract The emergence and widespread adoption of agile methodologies is often explained by the need to improve time management in Information Systems Development (ISD). Indeed, a growing body of evidence supports the view that agile methodologies are an effective means of delivering productivity gains through time savings. That is to say, agile methodologies can be used to increase speed and efficiency in ISD projects. In addition, lightweight agile methodologies are designed, by definition, to minimise wastes in the design and delivery of Information Systems and can therefore be used to support sustainability in IS projects (cf. Schmidt et al. Towards a procedural model for sustainable information systems management, IEEE, pp 1–10, 2009). However, the impact of agile methodologies on ISD project outcomes is less clear. In addressing this question, this research-in-progress paper uses a combination of existing literature and empirical data to construct a conceptual framework to explain how three different temporal aspects of agile methodologies (time pressure, polychronicity and periodicity) impact upon decision quality, thereby affecting ISD project outcomes. It is envisaged that this framework will be used to shed light on how agile methodologies impact upon project effectiveness or velocity, which is defined in this context as movement in the "right" direction.

1 Introduction

The only reason for time is so that everything doesn't happen at once—Albert Einstein

Time is an inherent quality of human life [19]. All human knowledge and understanding is fundamentally shaped by the temporal nature of our being in this world. For thousands of years, philosophers and scientists have engaged in active debate on

N.O. Riordan (✉) • T. Acton • K. Conboy • W. Golden
National University of Ireland, Galway, Ireland
e-mail: n.oriordan@yahoo.com

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_35,

the nature of time, our experience of time and its association with causality ([19], p. 327). Heidegger ([20], p. 437) asks "does *time* itself reveal itself as the horizon of *being*?" The notion of time pervades everyday language: "time is of the essence", "timing is everything", something can be "just in time" and "a stitch in time saves nine". In work situations, we try to "save time" or not "waste time": "time is money" and we are "on the clock". Yet time remains an abstract notion [25]: it is a "hidden dimension" [10, 18] and remains one of the most elusive concepts related to work [8, 32, 40, 43, 45].

Contemporary Western culture is characterised by a pervasive desire to maximise the temporal ordering and synchronisation of activities that dates back to Taylor's famous Time and Motion studies [40]. In this context, time is primarily viewed as an objective, chronological [43] and material commodity that is scarce, valuable, homogenous, linear and divisible [42]. In today's increasingly high-velocity business environment, effective time management is incredibly important. As a result, there is renewed interest in the subjects of time and timing in organisational studies (cf. [2, 40, 42]).

Nevertheless, an overwhelming proportion of projects are delayed beyond estimated completion time in most industries [49]. This is problematic given that project outcomes are typically judged on whether time deadlines are met [43]. In Information Systems Development (ISD), projects have consistently been plagued by delays and late deliveries [1, 4, 6, 49, 50]. In response to this problem, contemporary systems development methodologies (e.g. Agile and/or Lean methods) are designed to avoid cumbersome and time-consuming processes that elongate the development process [16, 17] and have been shown to deliver tangible cost/time savings to organisations [16]. In particular, lightweight agile methodologies are argued to represent an opportunity to support sustainability in IS projects as they are designed, by definition, to minimise wastes in the design and delivery of Information Systems (cf. [47]). Yet whilst time savings are a key feature of agile methods, very little research has explored the temporal dynamics of software development in general [36] or agile software development in particular.

This research-in-progress paper uses a combination of existing literature and empirical data to construct a conceptual framework to explain how three different temporal aspects of agile methodologies (time pressure, polychronicity and periodicity) impact upon decision quality, thereby affecting ISD project outcomes. It is envisaged that this framework will be used to shed light on how agile methodologies impact upon project effectiveness or velocity, which is defined in this context as movement in the "right" direction.

2 Managing Time in ISD

In the main, IS development is thought to be "essentially programmable" and is commonly viewed from a "software factory" perspective [37]. Thus, time management is typically based on mechanistic project management techniques (such as PERT), where time is primarily perceived as measurable clock time (Nandhakumar 1999), and productivity is measured in "lines of code per person month" (e.g. [6]).

Yet IS development projects appear to have been poorly served by traditional time management techniques. This is evidenced by frequent late delivery problems in the software industry [44] and the fact that an overwhelming proportion of ISD projects are delayed beyond estimated completion time [1, 49–51].

This phenomenon is at least partly explained by considering the complexity of temporal dynamics in ISD projects. It is, for example, well established that time units are not homogenous: segments of time are not necessarily equivalent [28]. In iterative projects, for example, progress tends to accelerate across iterations as teams gain experience of projects and of each other (stabilise). Similarly, different individuals tend to expend time at different rates in accordance with skill and experience levels. Thus, Nandhakumar ([38], p. 257) reports that ISD team members' work is marked by a flux of interwoven activities and multiple temporalities:

> Team members' work was marked by significant changes of pace in which periods of relative inactivity, such as waiting for a colleague to become available to complete some joint task, were matched by intensive efforts as deadlines for projects approached… many different activities were competing for team members' limited time resources and had to be fitted into, and interwoven with, the stream of their other activities… there were many interruptions to team members' work… [and] team members were often simultaneously engaged in several different 'projects… switch[ing] between different activities during the day…

In light of these observations, McLeod and Doolin [34] suggest that it is appropriate to adopt Markus and Robey's [31] "emergent process" perspective on IS development, viewing it as a "dynamic, multi-dimensional process, in which a development outcome emerges unpredictably from complex and reciprocal interactions between people and technology within an organisational context".

In today's software industry, excessive budget and schedule compression have become the norm [36]. This serves to exacerbate already significant time management problems in ISD. Further, the need for short cycle development has become more pressing with the advent of Internet Time [4].

The emergence and widespread adoption of agile methodologies [9, 48] is at least partly explained by the need to improve time management in ISD: one of the main aims and purported benefits of Agile Systems Development methodologies is that they deliver productivity gains through time savings [5]. In fact, the concept of agile software development has become more or less synonymous with short cycle time development [4, 53]. As a result, ISD project outcomes are most often judged on whether deadlines are met (Sarkar and Sarkar 2004).

Despite these assertions, there remains a paucity of research on the temporal dynamics ISD in general [36] and ASD in particular. Indeed, Sanders [45] indicates that "in more cases than not, time is a silent visitor" in IS research (p. iii). Thus, there is very little to challenge the dominant perspective of time that permeates existing literature on ISD. This research-in-progress is designed to address this gap.

3 The Temporal Dynamics of Agile Systems Development

This section identifies three arguments in favour of investigating the temporal dynamics of ASD. First, the concept of agility is inextricably bound up with a number of related temporal concepts such as speed, velocity and flexibility. It is well understood that speed refers to rapidity of movement and that velocity refers, more specifically, to speed in a given direction. However, agility also relates to the concept of flexibility or the capacity to "adapt within a given time frame" [9].[1]

Thus, agile methods are inherently temporal in nature: agile methods call for the creation of organic, flexible and empowered teams, who work in close collaboration with customers over a series of rapid development iterations. ASD teams work under extreme time pressure to deliver working software in short iterations [16, 17] with frequent, short-term decisions [13]. In this context, high-speed release cycles significantly compress development time frames [4].

From this perspective, the introduction of agile methods in ISD is seen to typify a phenomenon known as "temporal structuring", whereby people (re)produce (and occasionally change) temporal structures to orient their ongoing activities (cf. [40]). In particular, the project iteration can be viewed as a deliberate attempt to displace well-established Western views of linear time with more circadian rhythms. In Scrum, for example, software is developed incrementally in a series of short development phases, called "sprints" where teams initially prioritise and then freeze tasks at the start of each sprint [4]. Similarly, projects based on extreme programming are divided into sequences of self-contained, 1–3-week iterations [4].

Finally, Kumar and Motwani [27] indicates that a variety of techniques have been used to deliver significant time compression in the context of agile manufacturing (e.g. concurrent engineering, group technology principles (e.g. variant process planning), design for manufacturability and assembly, design and process optimization through Taguchi methods). Taken together, this arsenal of techniques well illustrates Hassard's ([19], p. 342) argument that synchronisation, sequence and rate are critical factors when we want to distribute time so that activities consume it in the most efficient manner ([19], p. 342). An investigation of the temporal dynamics of ASD may reveal new insights in relation to the application of these and other techniques in software supply chains.

In order to facilitate the development of a nuanced understanding of the role of temporal dynamics of ASD, this study adopts a decision-making lens in order to investigate the impact of the temporal dynamics of ASD on project outcomes. This is also an opportunistic approach as it represents an opportunity to address repeated calls for research on decision-making in agile setting [13, 33, 56].

[1] Volberda (cited in [9]) observes that flexibility measurable in terms of absolute speed of change; one must instead take into account both time taken to adapt to change and the variety of that change.

4 Building a Conceptual Framework

The purpose of this section is to develop a conceptual framework of the temporal parameters that have a bearing on decision-making in the context of ASD. This framework is based on a multidisciplinary review of literature and is informed by a semi-structured focus group that was carried out in December, 2011, to probe scrum master issues. This focus group was attended by seven scrum masters (representing three companies) and six researchers (including the focus group chair). The session focused on three focal areas: Issues, Recommendations and Resolutions, and flip charts and audio recordings were made to capture proceedings.

4.1 Time Pressure in Agile Decision-Making: Need for Speed

It is well established that schedule constraints have a critical impact on software development outcomes [36]. Problems arise, for example, when developers who feel that they are under pressure to meet task deadlines decide to take shortcuts in dealing with unanticipated complications [3]. Thus, time pressure is an often-cited source of quality problems in technological systems [7, 11].

Time compression has become the norm on today's software industry [36]. Thus, agile methods have emerged in response to the inefficiency of existing software development methods [21, 52]. Agile methods are designed to achieve software quality under time pressure and in unstable requirements environments [23]. In particular, agile methods represent an opportunity to support sustainability in IS projects as they are designed, by definition, to minimise wastes in the design and delivery of Information Systems (cf. [47]). Agile teams work under extreme time pressure to deliver working software in short iterations [16, 17]. They are required to follow fine-grained time planning and time reporting procedures, and their progress is explicitly, continually and publically measured.

As a result, agile decision-making processes are significantly more challenging than when traditional development approaches are used [9]. Agile decision-making is frequent and short term [17] and time and resource constrained [13]. In agile projects, decision-makers are frequently called upon to manage ongoing uncertainty in the face of severe time pressure. In these situations, decision-makers' information load—the "the amount of data to be processed per unit of time" [55]—is increased. As a result, decision outcomes can be compromised in agile teams as a result of time pressure [13] as decision-makers must maximise their ability to rapidly acquire and process information [15].

In these contexts, high-velocity [14] or hyper-vigilant [24] decision-making patterns are commonly used. The difficulty is that these patterns are characterised by (a) a nonsystematic or selective information search, (b) consideration of limited alternatives, (c) rapid evaluation of data and (d) selection of a solution without extensive review or reappraisal [26]. In this equation, the calibre of the information available to decision-makers is crucially important. One participant explained that *when you're in sprints, you've got a bit of pressure to deliver… you're not analysing what you're doing… you're skipping over it because you know you've deadlines to achieve*.

At the same time, it is acknowledged that decision-makers build-up a body of knowledge and familiarity about particular decision scenarios over time and can leverage that knowledge and experience to make more effective decisions at speed.

Proposition 1

Time pressure is negatively associated with decision quality in agile teams; this association is most pronounced in the earliest stages of projects and in the earliest stages of project iterations but has the potential to be moderated if information quality can be improved.

4.2 Synchronisation, Polychronicity and Timeliness

Under industrial capitalism, efficient organisation became synonymous with detailed temporal assessment of productivity ([19], p. 329), and the time period replaced the task as the focal unit of production (cf. [35], p. 14). In this context, the clock is the epitome of chronological time (*chronos*) and is the basis upon which functionally segmented parts and activities are temporally coordinated or synchronised.

Similarly, ASD methodologies are based on a linear (or iteratively linear) view of time and assume that events and tasks can be monochronically ordered. That is to say, it is assumed that events and tasks will manifest in an organised temporal way and follow a predetermined or at least predictable sequence (cf. [46], p. 114). However, the lived reality of agile practitioners is often polychronic: regardless of previous planning decisions, events and tasks frequently occur in an unexpected temporal way; they are irregular, sporadic, uneven and do not follow a fixed schedule (cf. [46], p. 114). Given that agile methods are designed to ensure flexibility, it is important that agile teams maintain what Hassard ([19], p. 333) refers to as "flexible, event-based trajectories". Nevertheless, excessive polychronicity in agile teams results in what one respondent referred to as *drag*. In this case, *everybody just has to stop what they're doing and go fix things*. As a result, momentum or velocity is either lost or it is not achieved in the first place. In this context, more sophisticated temporal structures are needed to manage and coordinate organisational processes ([19], p. 329). The challenge is one of *kairos*, or appropriate timing ([54], p. 57), rather than *chronos*.

From a decision-making perspective, a rich variety of dynamic decision-making studies has revealed much about the flaws in people's abilities to manage dynamic complexity [12]. In agile settings specifically, excessive polychronicity impairs planning decisions as agile teams are forced to engage in *poker planning*: *we have to cost things [in terms of time, but] the plan changes two hours later. We build insurance into the sprint, knowing that we'll lose people or knowledge from the team.*

Thus, we propose that:

Proposition 2

Excessive polychronicity is an indicator of environmental uncertainty in agile teams; it is negatively associated with decision quality, but this association can be moderated if decision-making timeliness can be improved.

4.3 The Agile Iteration: Periodicity and Decision-Making

The project iteration is one of the hallmarks of ASD. It helps to ensure that potential problems are detected as soon as possible. By breaking time into digestible fragments, it reduces temporal uncertainty ([19], p. 338), improves synchronisation within and across teams and increases productivity by minimising the effects of Parkinson's Law (1962).[2] Whilst it has been established that decision-making in agile teams is dynamically complex [33], the impact of the project iteration on decision quality in agile teams is not well understood.

The project iteration may increase susceptibility to immediacy bias [51] whilst reducing risk aversion (cf. [30, 41]) as it reduces the apparent implications of a particular decision to the time frame of a given iteration: it is always possible, in theory at least, to undo a particular decision in a subsequent iteration. At the same time, empirical studies indicate that agile decision speed increases over time as decision-making is informed by experiences from previous iterations. By the same token, the nature of the iteration is such that decisions made in the early stages of a project may influence subsequent decision-making behaviour. The difficulty is that "decision making behaviour at time t is largely predictable from decision making behaviour at time $t-1$—irrespective of whether this is appropriate" [22]. Finally, decision-making in early iterations may impose unforeseen constraints on choices available in subsequent iterations as iterative decisions are more likely to be intertemporal in nature (cf. [29]).

Proposition 3

Iterative decision-making practices in agile teams are negatively associated with decision quality.

4.4 Agile Decision-Making and IT

Finally, information technology (IT) has played a key role in transforming the temporalities of work and in creating temporal symmetry between work groups [28]. The proper use of IT has been identified as a key facilitator of agility in

[2] Parkinson's Law states that work will expand to fill the available time. This law has been formalised by a number of authors (e.g. [1]).

organisations [4, 39] as it enables faster information flows and speedier decision-making, resulting in time compression in the supply chain [27]. In the context of ISD, automated software tools affect the temporal and spatial organisation of work [40, 43] by enabling the surveillance and control of team members' work practices over time–space [38, 57]. Given the decision-making challenges that are inherent in ASD, there is a need to explore the tension between the need for information to aid decision-making in ASD and the desire to create minimal documentation in ASD.

5 Conclusions

Figure 1 summarises the conceptual framework of the impact of three temporal parameters on decision-making in agile teams developed in this paper.

Time pressure is negatively associated with decision quality in agile teams; this association is most pronounced in the earliest stages of projects and in the earliest stages of project iterations but has the potential to be moderated if information quality can be improved.

Excessive polychronicity is an indicator of environmental uncertainty in agile teams; it is negatively associated with decision quality, but this association can be moderated if decision-making timeliness can be improved.

Iterative decision-making practices in agile teams are negatively associated with decision quality.

This framework explores the impact of three temporal parameters on decision-making in agile teams and is informed by both existing literature and preliminary data. It serves as a starting point in deconstructing the intertwined concepts of agility, flexibility, uncertainty and time in the context of decision-making and may be used guiding future research in this area. In particular, it is envisaged that understanding the temporal aspects of decision-making in agile teams will help to lay a foundation for further studies of the interactions between dynamic complexity and human performance (cf. [12]).

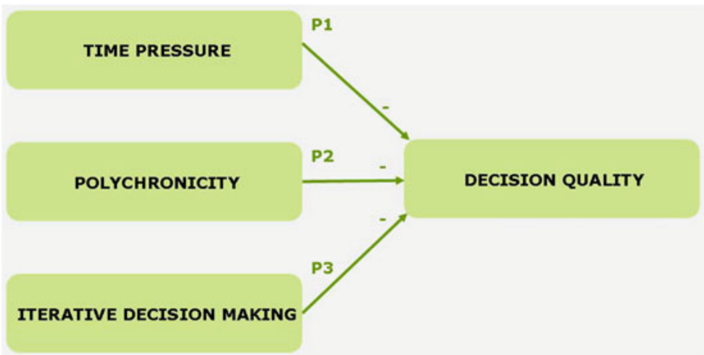


Fig. 1 Conceptualising the temporal aspects of decision making in agile teams

As such, this research-in-progress paper reports on an ongoing study investigating the temporal parameters of decision-making in agile teams. It is part of a growing stream of research that explicitly recognises and calls attention to the concept of temporality in the fields of organisation science, decision-making and IS. It also serves to call attention to the capacity of agile methodologies to support sustainability as well as flexibility in ISD.

Acknowledgment This research is supported by the Irish Social Sciences Platform (ISSP), funded under the Programme for Research in Third Level Institutions, administered by the HEA and co-funded under the European Regional Development Fund (ERDF), and also supported in part by Science Foundation Ireland grant 10/CE/I1855 to Lero.

References

1. Abdel-Hamid TK, Madnick SE (1989) Lessons learned from modeling the dynamics of software development. Commun ACM 32(12):1426–1438
2. Ancona D, Chong CL (1996) Entrainment: pace, cycle, and rhythm in organizational behavior. In: Cummings LL (ed) Research in organizational behavior: an annual series of analytical essays and critical reviews. Elsevier Science, New York, pp 251–284
3. Austin RD (2001) The effects of time pressure on quality in software development: an agency model. Inf Syst Res 12(2):195–207
4. Baskerville R, Pries-Heje J (2004) Short cycle time systems development. Inf Syst J 14(3):237–264
5. Begel A, Nagappan N (2007) Usage and perceptions of agile software development in an industrial context: an exploratory study. First international symposium on empirical software engineering and measurement, IEEE, pp 255–264
6. Blackburn JD, Scudder GD, Van Wassenhove LN (1996) Improving speed and productivity of software development: a global survey of software developers. IEEE Trans Softw Eng 22(12):875–885
7. Brooks FP Jr (1975) The mythical man-month. Addison-Wesley Longman, Boston, MA
8. Cooper LC, Rouseau DM (2000) Trends in organizational behavior: time in organizational behavior. Wiley, Chichester
9. Conboy K (2009) Agility from first principles: reconstructing the concept of agility in information systems development. Inf Syst Res 23(3):329–354
10. Das TK, Teng BS (2001) Strategic risk behaviour and its temporalities: between risk propensity and decision context. J Manage Stud 38(4):515–534
11. DeMarco T (1993) Why does software cost so much? Dorset House, New York
12. Diehl E, Sterman JD (1995) Effects of feedback complexity on dynamic decision making. Organ Behav Hum Decis Process 62(2):198–215
13. Drury M, Conboy K, Power K (2012) Obstacles to decision making in Agile software development teams. J Syst Softw 85(6):1239–1254
14. Eisenhardt KM (1989) Making fast strategic decisions in high-velocity environments. Acad Manage J 32(3):543–576
15. Fiedler FE (1986) The contribution of cognitive resources and leader behavior to organizational performance1. J Appl Soc Psychol 16(6):532–548
16. Fitzgerald B, Hartnett G, Conboy K (2006) Customising agile methods to software practices at Intel Shannon. Eur J Inf Syst 15(2):200–213
17. Fowler M, Highsmith J (2001) The Agile manifesto. Softw Dev 9:28–32
18. Hall ET (1966) The hidden dimension. Anchor, New York
19. Hassard J (1999) Images of time in work and organization. In: Clegg S, Hardy C (eds) Studying organization: theory and method. Sage, London, pp 327–344

20. Heidegger M (1962) Being and time. Blackwell, Oxford
21. Highsmith J (2002) Agile software development ecosystems. Addison-Wesley Longman, Boston, MA
22. Huber GP (1981) The nature of organizational decision making and the design of decision support systems. Manage Inf Syst Q 5(2):1–10
23. Huo M, Verner J, Zhu L, Babar MA (2004) Software quality and agile methods. Proceedings of the 28th Annual International Computer Software and Applications Conference, IEEE, vol 521, pp 520–525
24. Janis IL, Mann L (1977) Decision making: a psychological analysis of conflict, choice, and commitment. Free Press, New York
25. Jaques E (1982) The form of time. Crane Russak, New York
26. Johnston JH, Driskell JE, Salas E (1997) Vigilant and hypervigilant decision making. J Appl Psychol 82(4):614
27. Kumar A, Motwani J (1995) A methodology for assessing time-based competitive advantage of manufacturing firms. Int J Oper Prod Manage 15(2):36–53
28. Lee H (1999) Time and information technology: monochronicity, polychronicity and temporal symmetry. Eur J Inf Syst 8(1):16–26
29. Loewenstein G, Thaler RH (1989) Anomalies: intertemporal choice. J Econ Perspect 3(4):181–193
30. Lopes LL (1996) When time is of the essence: averaging, aspiration, and the short run. Organ Behav Hum Decis Process 65(3):179–189
31. Markus ML, Robey D (1988) Information technology and organizational change: causal structure in theory and research. Manage Sci 34:583–598
32. Massey AP, Montoya-Weiss MM, Hung YT (2003) Because time matters: temporal coordination in global virtual project teams. J Manage Inf Syst 19(4):129–156
33. McAvoy J, Butler T (2009) The role of project management in ineffective decision making within Agile software development projects. Eur J Inf Syst 18(4):372–383
34. McLeod L, Doolin B (2012) Information systems development as situated socio-technical change: a process approach. Eur J Inf Syst 21:176–191
35. Mumford L (1934) Technics and civilisation. Harcourt, Brace and World, New York
36. Nan N, Harter DE (2009) Impact of budget and schedule pressure on software development cycle time and effort. IEEE Trans Softw Eng 35(5):624–637
37. Nandhakumar J (1999) Managing Time in a Software Factory: Temporal and Spatial Organization of Is Development Activities. Inf Soc 18(4):251–262
38. Nandhakumar J (2002) Managing time in a software factory: temporal and spatial organization of IS development activities. Inf Soc 18(4):251–262
39. Newman WS, Podgurski A, Quinn RD, Merat FL, Branicky MS, Barendt NA, Causey GC, Haaser EL, Kim Y, Swaminathan J (2000) Design lessons for building agile manufacturing systems. IEEE Trans Rob Autom 16(3):228–238
40. Orlikowski WJ, Yates JA (2002) It's about time: temporal structuring in organizations. Organ Sci 13:684–700
41. Read D, Loewenstein G, Rabin M (1999) Choice bracketing. J Risk Uncertain 19(1):171–197
42. Sahay S (1997) Implementation of information technology: a time-space perspective. Organ Stud 18(2):229–260
43. Sarker S, Sahay S (2004) Implications of space and time for distributed work: an interpretive study of US Norwegian systems development teams. Eur J Inf Syst 13(1):3–20
44. Sauer C (1993) Why information systems fail: a case study approach. Alfred Waller Ltd., Oxfordshire, UK
45. Saunders C, Kim J (2007) Editor's comments: perspectives on time. Manage Inf Syst Q 31(4):pp. iii–xi.
46. Schein EH (1992) Organizational culture and leadership. Jossey-Bass, San Francisco, CA
47. Schmidt NH, Erek K, Kolbe LM, Zarnekow R (2009) Towards a procedural model for sustainable information systems management. IEEE, pp 1–10

48. Tan CH, Teo HH (2007) Training future software developers to acquire agile development skills. Commun ACM 50(12):97–98
49. Toxvaerd F (2006) Time of the essence. J Econ Theory 129(1):252–272
50. Van Genuchten M (1991) Why is software late? An empirical study of reasons for delay in software development. IEEE Trans Softw Eng 17(6):582–590
51. Van Boven L, White K, Huber M (2009) Immediacy Bias in Emotion Perception: Current Emotions Seem More Intense Than Previous Emotions, Journal of Experimental Psychology: General 138(3):368
52. Wang X, Conboy K, Cawley O (2012) 'Leagile' software development: an experience report analysis of the application of lean approaches in agile software development. J Syst Softw 85:1287–1299
53. Wetherbe JC, Frolick MN (2000) Cycle time reduction: concepts and case studies. Commun AIS 3(4es):1
54. Williams JD (2009) An introduction to classical rhetoric: essential readings. John Wiley & Son
55. Wright P (1974) The harassed decision maker: time pressures, distractions, and the use of evidence. J Appl Psychol 59(5):555
56. Zannier C, Maurer F (2007) Comparing decision making in agile and non-agile software organizations. In: Concas G, Damiani E, Scotto M, Succi G (eds) Agile processes in software engineering and extreme programming. Springer, Berlin, pp 1–8
57. Zuboff S (1988) In the age of the smart machine: the future of work and power. Basic Books, New York

Towards Monitoring Correspondence Between Education Demand and Offer

Peteris Rudzajs and Marite Kirikova

Abstract One of the goals of educational systems is to ensure correspondence between education provided by educational institutions and education needed in industrial environment. Therefore, the correspondence between education demand and offer should be a subject of continuous monitoring, especially in current highly dynamic work environment of age of knowledge economy. Monitoring in different systems is used with the purpose to achieve their effective functioning with respect to the goals of these systems. The monitoring in educational domain is challenged by qualitative nature and diversity of information that should be analyzed to obtain the meaningful monitoring results. The paper proposes architecture of education demand and offer monitoring information system, which encompasses services for handling diverse textual information from various sources at different levels of automation.

1 Introduction

Monitoring is "the act of observing something (and sometimes keeping a record of it)" [1]. The main purpose of monitoring systems is to provide the users with the ability to observe a situation for any changes, which may occur over time. Basic functionality of monitoring systems includes gathering of source data (mainly quantitative data), data processing, and analysis to provide decision support information to *scientific institutions,* the users of the system.

While there are different monitoring systems available in education domain, none of them can currently monitor the correspondence between education *demand*

P. Rudzajs (✉) • M. Kirikova
Department of Systems Theory and Design, Riga Technical University,
Kalku iela 1, 1658 Riga, Latvia
e-mail: Peteris.Rudzajs@rtu.lv; Marite.Kirikova@cs.rtu.lv

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_36,

and offer (further in text d/o). Furthermore, education institution is a member of an educational “ecosystem” [2] that consists of several interdependent subsystems, namely, *education institutions* that provide education and that influence the content of study courses by incorporating topics emerging from new scientific findings, *industrial organizations* that need professionally well-educated employees, and administrative bodies that support and assess education institutions. For the education institution to be a productive member of the educational ecosystem, it is necessary to satisfy the needs of scientific, industrial, and other organizations. Education d/o monitoring system/services can help to meet these needs by providing an insight in knowledge, skills, or competences currently demanded/offered in educational, industrial, and scientific environment.

In managing study process, it is essential to have an up-to-date knowledge about the correspondence between education d/o. Taking into consideration that analysis of this correspondence requires continuous knowledge updating with new information due to rapid changes in business environment, there is a need to design tools that enable amalgamation and processing of relevant data for acquisition of information which triggers knowledge updates. Education d/o monitoring system, described in this paper, is one of such tools.

For simplicity in this paper, the terms “education d/o information” and “skills” are used interchangeably to denote concerned competences, knowledge, and skills. The basic challenges that prevent educational institutions from monitoring education d/o are the following:

- Information should be gathered from diverse information sources, e.g., study and certification course descriptions, job advertisements, various competence frameworks [3, 4], and national occupation standards [5].
- Information is stored in diverse formats like unstructured and semi-structured textual documents (mostly Web pages), as well as in databases.
- Information can be processed by different degrees of automation: manually, automatically, and with the partial IT support.

This requires a number of functions for gathering and analyzing information that is needed for education d/o monitoring. The paper proposes education d/o monitoring system based on the service architecture that includes services for handling diverse textual information from various sources at different levels of automation and thus helps to meet abovementioned challenges.

The paper is structured as follows. In Sect. 2, related work is outlined. Further, in Sect. 3, the overview of the monitoring system’s architecture is presented. In Sect. 4, we show how the monitoring system’s services help to deal with the challenges of education d/o monitoring. Next, in Sect. 5, the usage scenarios of education d/o monitoring information are outlined. Conclusions and further research directions are presented in Sect. 6.

Currently education d/o information monitoring is considered in the domain of Information and Communication Technology. The system has not yet been applied in other domains.

2 Related Work

Monitoring systems are designed for (1) mechanical systems monitoring (e.g., in aerospace [6]), (2) Earth's condition monitoring [7], (3) surveillance [8], (4) IT infrastructure monitoring [9], (5) quality of service (QoS) monitoring [10], (6) social network monitoring [11], (7) key performance indicators (KPI) monitoring in business environment [12], (8) education system monitoring [13, 14], and other purposes.

Görgens and Kusek [14] provide the monitoring system framework based on 12 components organized in three subsets: (1) components related to people, partnerships, and planning that support data production and data use constitute the enabling environment for monitoring to function; (2) components related to data management processes that involve data sourcing, collection, collation, analysis, reporting, and use of all types of monitoring data; and (3) a component capturing the system's central purpose—to analyze the data in order to create information to inform and empower decision-making. Although provided 12 components were initially designed to measure the status of national health program, this approach has general applicability. It can be used for monitoring systems in education, agriculture, tourism, finance, civil society, private sector, and so forth.

In the most of cases, education system monitoring is considered in the context of education management information systems where the goals of monitoring are to gather information from education information sources and trace the execution of education plans [13]. Example indicators monitored by education institutions are updated files on students, personnel, buildings, and equipment; class results compared to previous years and other similar schools; demographic data by age group; previous rates of admission flow; projection of personnel and teacher supply; distribution by grade, gender, and age; and number of non-teaching personnel by category, age, and gender. Monitoring systems in education are also targeted to monitor and evaluate the indicators for ICT in education projects [14]. Indicators include ratio of students to computers, availability of computer networks in schools, integration of ICT into the curriculum, number of schools incorporating ICT, ICT-based learning outcome evaluation, and other indicators.

Usually the architectures of monitoring systems are represented in layers (e.g., data collector, data analyzer, data presentation facilities), where each layer is designed for a specific purpose and for providing of services to other layers. In the most of cases SOA is adopted for the design of monitoring systems [16]. SOA is a system design concept that builds an information system by connecting loosely coupled services. This architecture type is well suitable for designing the architecture of monitoring systems as it supports connections of distributed services available in monitoring system. The education d/o information monitoring system outlined in this paper differs from other abovementioned monitoring systems with its ability to obtain the monitoring source information from unstructured and semi-structured textual information sources. These sources cannot be handled in the same way as sources where information is quantitatively measurable.

3 Education D/O Information Monitoring Architecture

The architecture of the education d/o information monitoring system (EduMON) presented in Fig. 1, initially had been designed using bottom-up approach by analyzing the university-industry collaboration in the context of education d/o information exchange [17, 18] and then it had been revisited and updated using top-down approach by applying the architecture design principles (layers of services, service orientation, etc.) of monitoring systems in education domain [19].

The revisited architecture of the EduMON, presented in this paper, can be used as a dynamic and systemic framework to facilitate d/o monitoring. The architecture consists of several classes of services and supports the monitoring process from the retrieving and extracting information from relevant sources, in its initial steps, to the analysis of the correspondence between education information in various sources, in its final steps. Currently some parts of services of the architecture had been instantiated in methods and prototypes, e.g., Web crawler for job advertisement retrieval [20] (as a part of retrieval services) and tool for indirect comparison between job positions and university courses [21] (as a part of Analysis services). Instantiation of other services is still under the development.

The representation of revisited architecture of EduMON was designed with the purpose to focus on high-level design elements and relations between the elements rather than on every detail of the system. By design elements we mean modules, components, databases, and external systems. As service-oriented architecture (SOA) approach is used, modules and components are denoted as services.

Basic functionality of the monitoring systems includes *gathering of source information* and *information processing* and *analysis* to provide decision support

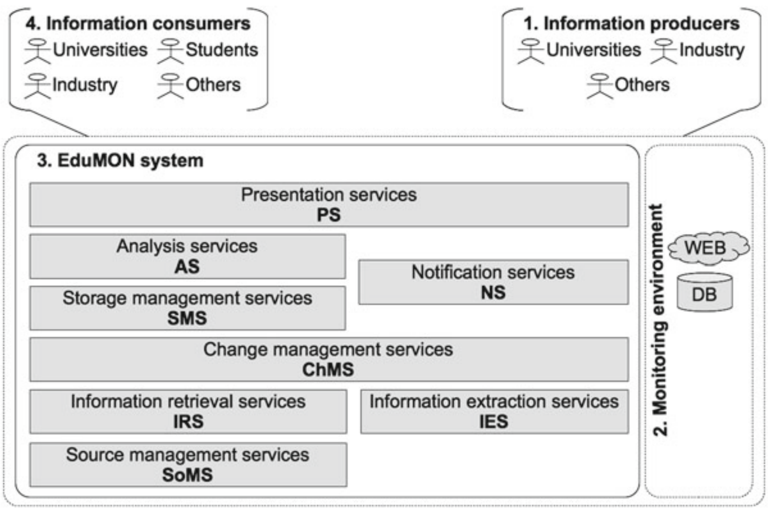


Fig. 1 EduMON system's overview

information to the users of the system. EduMON (see Fig. 1) gathers source information prepared by Information producers (see 1 in Fig. 1) from the Monitoring environment (see 2 in Fig. 1). Information producers are Universities (providing course descriptions), Industry (providing job advertisements), and other institutions (e.g., ones providing occupational standards, social network profiles). Monitoring environment is the medium where information is stored, such as the World Wide Web available publicly and various databases available locally in organizations. After gathering source information, EduMON processes and analyzes this information and, via internal services, provides it to Information consumers for decision support (see 4 in Fig. 1). Information consumers are University, Industry, or other users interested in information provided by the EduMON.

EduMON architecture is designed as a set of extendable services of the following classes: (1) *Source management services* for preparation and management of information sources; (2) *Information retrieval services* for continuous information gathering from various information sources; (3) *Information extraction services* for information extraction from information source documents based on information source models and for annotation (the assignment of metadata) of information source documents; (4) *Change management services* for management of information source changes; (5) *Storage services* for storing information source descriptions and extracted information; (6) *Notification services* for notification of interested parties in case of the changes in the information sources and other elements represented by the system; (7) *Analysis services* for any kind of analysis based on gathered information, e.g., most demanded skills in industry, differences between skills provided by university courses, and skills included in other information sources such as job advertisements, occupational standards, and certification standards that represent actual demand of competencies in industry; and (8) *Presentation services* are intended to be a Web-based graphical user interfaces that offers concise (in the form of tables, charts, figures, etc.) monitoring information, and it is designed according to a role-based strategy to offer different views depending on the type of Information consumer.

Services of each service class can be designed and implemented with different degrees of automation: so that they can be performed manually, with the partial IT support, or automatically.

Compared to initially developed architecture [18], the revisited architecture is more flexible and allows various services to meet the d/o monitoring challenges, e.g., services for retrieving, extracting, and managing information from diverse information sources in diverse formats with different degree of automation.

The education d/o is dynamic and changing over time. To be alert when education d/o changes occur, EduMON architecture provides dynamic services responsible for continuous information retrieval (e.g., using Web crawler), change detection, and services for notifications to interested parties (e.g., responsible teachers, HR managers). For more usage cases, see Sect. 5.

Basic functionality and implementation considerations of the services of each service class are outlined in [19]. In the next section, the monitoring challenges and corresponding support provided by EduMON services are discussed.

4 Meeting the Education D/O Monitoring Challenges

As emphasized in Sect. 1, basic challenges preventing educational institutions from monitoring education d/o are the diversity of information sources and their formats. In Sects. 4.1 and 4.2, we provide detailed description of these challenges and provide the description of the possibilities of EduMON to resolve them partly.

4.1 Handling Textual Representation and Diversity of Information

Information about education d/o can be reflected in such information sources as study and certification course descriptions, job advertisements, various competence frameworks (e.g., [3, 4]), national occupation standards [5], and other sources. Education d/o information can be encoded in different attributes in each source of information, e.g., see the examples of conceptual models for two main information sources—course descriptions and job advertisements in Fig. 2a, b (the main attributes are highlighted). The conceptual model of courses descriptions is constructed on the basis of course descriptions of seven curricula from five universities (Vienna University of Technology, University of Vienna, University of Rostock, Dublin City University, Riga Technical University). In these descriptions, education d/o information was reflected in such sections as the *course abstract*, *required and obtainable knowledge and skills*, *learning outcomes*, and *topics*. The conceptual model of job advertisements is constructed by analyzing job advertisements of two main job-seeking portals in Latvia—www.workingday.lv and www.cv.lv. Education d/o information was found in such sections as the *job summary*, *professional skills and qualification*, and *responsibilities*.

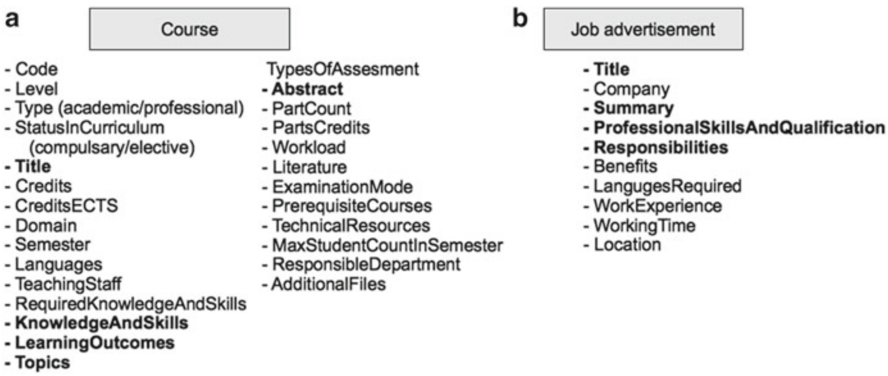


Fig. 2 Conceptual models of course descriptions (**a**) and job advertisements (**b**)

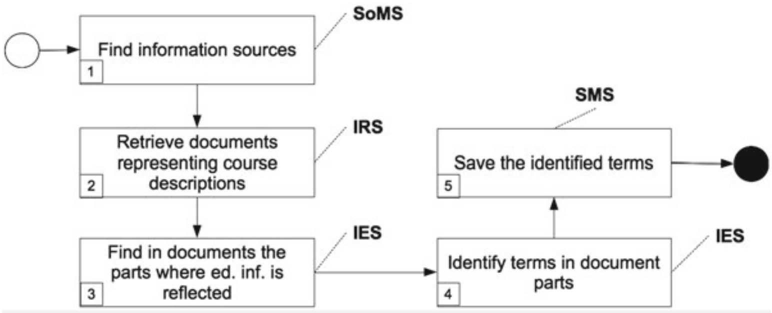


Fig. 3 Typical process to obtain and store education d/o information

In Fig. 3, an example of typical process to obtain and store d/o information reflected in university course descriptions is illustrated. For each process step the corresponding EduMON services are identified. Other education d/o sources, e.g., job advertisements, can be handled using similar process steps and EduMON services. The process starts with the need to collect education d/o information from university courses. First the corresponding information sources should be identified, e.g., university Web pages. Next, the retrieval of course descriptions from university Web pages is performed. Then, the particular parts reflecting education d/o information should be found in the document. As the education d/o information usually is represented as textual information in form of sentences, the most important terms can be identified and associated with particular textual information [22] for better understanding and for easing later processing. Currently we apply the approach that reduces the large textual descriptions to terms representing the text, since it eases their further processing and reference establishment [22]. For instance, in the textual description of learning outcome, "Students will know the main data analysis tools and techniques available" (see Fig. 5), we can identify such terms as "data analysis tools" and "data analysis techniques" (when the automated information extraction is used, we can use dictionaries where the possible terms are found). Thus we associate these terms with the textual description of the learning outcome—one of the attributes of the course. Next, the education d/o information and associated terms should be stored. Other approaches are possible [23]; however, their particular implementation in EduMON is still under research.

Manual only execution of the process reflected in Fig. 3 would be too time-consuming, thus IT support is vital to speed up the education d/o information retrieval, extraction, and storage and, afterwards, the analysis and reporting. To fill-up the EduMON with information that is needed for IT-supported education d/o monitoring, diversity of conceptual structures should be respected and appropriate methods and functions [19] combined to retrieve and extract the parts of documents that potentially reflect education d/o information. As it is presented in Fig. 3, the services from the EduMON can be enabled in execution of each process step, namely, these are services from the source management (SoMS), information retrieval (IRS), extraction (IES), and storage management (SMS) service classes.

Thus by enabling the IT support for retrieval, extraction, and storage process steps and applying "normalization" of large textual information (identifying or just extracting terms from particular attribute), the EduMON helps to handle the diversity of information sources.

In addition to diversity of information sources, the information can be presented in various formats, e.g., as structured information (e.g., XML, OWL), as unstructured information (simple text as in Web pages), or as data in databases. Therefore, additional functionality is needed to deal with the variety of formats (for more details see [17]).

4.2 Degrees of Service Automation

The architecture of EduMON is designed to allow the extensions of services in each service class. Thus the architecture enables to implement the services with different degrees of automation. Such approach helps to deal with the diversity of information sources and availability of information in different formats. Table 1 represents possible service combinations for education d/o information monitoring regarding the degree of automation of each service. In the table, the first column and row list all the services of EduMON. Filled cells denote the relation between the services and possible combinations of degrees of their automation (M, manually; SA, semiautomatically; A, automatically). Table 1 should be read by rows, e.g., row SoMS, column IRS: M-M denotes that both Source management services (SoMS) and Information retrieval services (IRS) are related and implemented as manual services. M-SA denotes that SoMS services are conducted manually, but IRS—with the partial IT support (semiautomatically). One possible implementation of process steps (given in Fig. 3) is highlighted in bold in Table 1 and for better perception illustrated in Fig. 4b.

Figure 4a represents the services with manual execution of process steps (see the M in brackets). In the case of manual execution of process steps (and thus manual execution of the EduMON services), obtained information is stored in the text or spreadsheet editors and processed manually.

The use of advanced tools and methods (e.g., [24, 25]) for processing, analysis, and presentation of education d/o information is hindered in this case.

Regarding Fig. 4b, the IT support (partial of full) for the process steps is enabled; thus services can be executed automatically or semiautomatically (with partial human involvement). The service of the SoMS class is executed manually in order to reduce the scope of information retrieval and use only trusted information sources. The services of IRS and IES classes are executed semiautomatically, thus enabling the use of automated information retrieval and extraction techniques (some of them are described in [22, 23]).

Table 1 Possible service combinations based on the degree of automation of services

	SoMS	IRS	IES	ChMS	SMS	NS	AS	PS
SoMS	–	M-M **M-SA**	–	–	–	–	–	–
IRS	–	–	M-M M-SA SA-M **SA-SA**	M-M SA-M **SA-SA** SA-A A-SA A-A	–	–	–	–
IES	–	–	–	M-M M-SA M-A SA-M **SA-SA** SA-A	–	–	–	–
ChMS	–	–	–	–	M-A **SA-A** A-A	M-M **SA-SA** SA-A A-SA A-A	–	–
SMS	–	–	–	–	–	–	M-M A-M **A-SA** A-A	M-M A-SA **A-A**
NS	–	–	–	–	–	–	–	M-M **SA-A** A-A
AS	–	–	–	–	–	–	–	M-SA Sa-SA SA-**A** A-A
PS	–	–	–	–	–	–	–	–

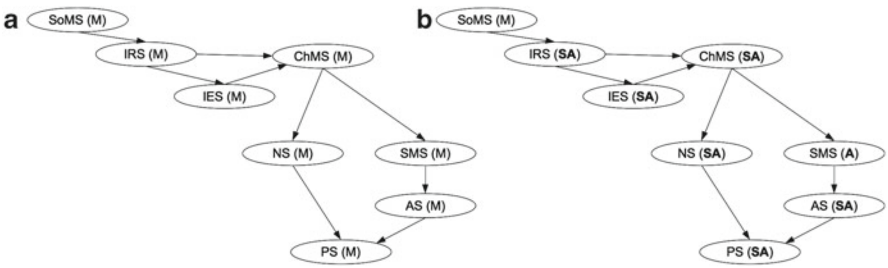


Fig. 4 Relations between services and the order of execution. The degree of automation of the services is given in brackets (**a**) only manually executed services, (**b**) both semiautomatically and automatically executed services. When branching, services can be executed in parallel

By supporting various degrees of service automation, the EduMON fosters the selection of the appropriate degree of automation in each particular case. Automatically and semiautomatically implemented EduMON services reduce the time and effort of manual work in preparation and analysis of the education d/o information.

5 Usage of the Monitoring System

When the essential information is collected by the EduMON services, analysis and reporting (supported by the classes of Analysis services and Presentation services of the EduMON) become available. Typical usage scenarios are determined by the referable elements represented by the system. For instance, the following elements can be considered: a curriculum, a course, course attributes (in this paper, learning outcomes and topics are mainly used as course attributes; however, it is possible to change these attributes or introduce the new ones), and a job advertisement. It means that from each representation element, it is possible to "see" related information at all other representation elements. For instance, from the *course* EduMON can provide related information about *curricula* where similar courses are available; *other courses*, *topics*, and *learning outcomes* (from the same curriculum or related curricula), and related *job advertisements*. An example of related representation of learning outcomes and job advertisements is given in Fig. 5.

EduMON provides *Change management* and *Notification services* (see ChMS and NS in Fig. 1) to enable continuous monitoring of information sources and education d/o correspondence. These services provide the users with ability to trace and to be notified about changes regarding related representation elements. This

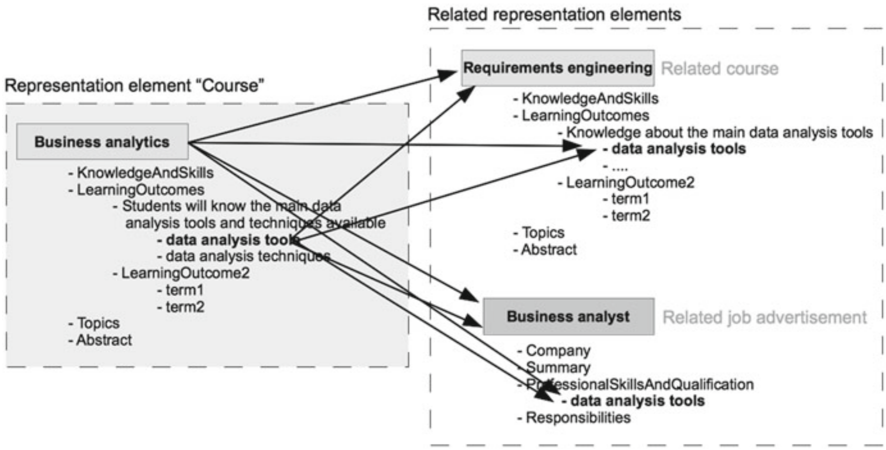


Fig. 5 An example of related representation of learning outcomes and job advertisements

feature is essential in maintaining the correspondence of education d/o. For instance, regarding the course, if the changes are detected in the related courses or in the job advertisements, responsible users (course designers, teachers) can be notified about changes, and they can trace the source of changes (if a related course has been changed, responsible user can see the trace and the reason why the related course has been changed). A responsible user can decide to ignore this change or to make changes in the course she/he is responsible for. We should be aware that each change in job advertisements cannot and is not useful to be reflected in course content. Perhaps, the better way is to aggregate education d/o reflected in job advertisements regarding to a job position or a profession—thus, if new knowledge or skills are being demanded in a particular job position, they should be reviewed and anticipated in course if necessary. Furthermore, seeing that new skills are being demanded in some of companies, other companies will perhaps follow the trend. Thus the anticipation of these skills and including corresponding knowledge in the university courses can be of high value for preparing skilled workforce that meets the industry demands.

As represented in Fig. 1, the users of the monitoring results can be not only university but also students and industry. For students, the EduMON can provide information for deciding which courses to attend in order to obtain knowledge that is required in job position of their interest. For industry, the results can be used to identify further education possibilities for current employees, to see the potential curricula where the graduates (future employees) could have skills required in newly created job positions, and the monitoring results can be as a basis for on-demand course development. For sure, the usage scenarios are not limited to those described in this paper.

6 Conclusions and Future Work

In this paper the approach of application of the monitoring system in education domain was briefly described. The approach is new and can provide benefits both for University and Industry, e.g., by helping to get an insight into current education d/o in industrial and education environment and providing information for decision support regarding curricula management and choosing courses for continuing education.

The approach enables us to handle three main challenges in education d/o monitoring, namely, (1) textual representation of monitoring information, (2) diversity of sources and formats of information, (3) and the need for various levels of automation of services involved in education d/o monitoring. The proposed architecture of EduMON, its flexibility and support for different levels of automation of services, is the key for handling these challenges with IT support to education d/o information retrieval, extraction (including the normalization of large textual information), storage, and analysis. The benefits of the usage of the results obtainable by EduMON include the enhanced maintenance of courses by overseeing and anticipating the information about related courses, curricula, and job positions; course selection by job position of interest; the selection of future employees based on skills and knowledge obtainable in university courses; and other possibilities.

Currently only part of services, such as Web crawler for job advertisement retrieval [19] (as a part of retrieval services) and tool for indirect comparison between job positions and university courses [20] (as a part of Analysis services) are implemented and evaluated. Further development of services of the EduMON includes new methods for Information extraction service class, as well as gradual development and evaluation of methods and software for Change management, Notification, and Presentation service classes.

Acknowledgments This work has been supported by the European Social Fund within the project "Support for the implementation of doctoral studies at Riga Technical University."

References

1. Princeton University. About WordNet, http://wordnet.princeton.edu
2. Kirikova M, Grundspenkis J (2008) Educational "ecosystem" for information systems engineering. Knowledge Creation Diffusion Utilization, Technical Report, Riga Technical University, pp 1–15
3. European Committee For Standardization. European e-Competence Framework 2.0. http://ecompetences.eu/
4. SFIA Foundation. Framework reference SFIA version 5. http://www.sfia.org.uk/v5/en/
5. VISC. Latvian Nacional Occupation Standard Registry. http://visc.gov.lv/saturs/profizgl/stand_registrs.shtml
6. Phillips P, Diston D (2011) A knowledge driven approach to aerospace condition monitoring. Knowl Based Syst 24:915–927
7. Wang X, Gao W, Slusser JR, Davis J, Olson B, Janssen S, Janson G, Durham B, Tree R, Deike R (2008) USDA UV-B monitoring system: an application of centralized architecture. Comput Electron Agric 64:326–332
8. Vallejo D, Albusac J, Castro-Schez JJ, Glez-Morcillo C, Jiménez L (2011) A multi-agent architecture for supporting distributed normality-based intelligent surveillance. Eng Appl Artif Intel 24:325–340
9. Mehaoua A, Ahmed T, Asgari H, Sidibe M, Nafaa A, Kormentzas G, Kourtis T, Skianis C (2006) Service-driven inter-domain QoS monitoring system for large-scale IP and DVB networks. Comput Commun 29:1687–1695
10. Semenov A, Veijalainen J, Boukhanovsky A (2011), A Generic Architecture for a Social Network Monitoring and Analysis System. In Network-Based Information Systems (NBiS), 14th International Conference on (pp. 178–185). IEEE
11. Wetzstein B, Danylevych O, Leymann F, Bitsaki M, Nikolaou C, Heuvel W-JVD, Papazoglou MP (2009) Towards monitoring of key performance indicators across partners in service networks. 1st Workshop on Monitoring Adaptation and Beyond MONA, pp 7–17
12. Carrizo L, Sauvageot C, Bella N (2003) Information tools for the preparation and monitoring of education plans. Paris: Unesco
13. Wagner D, Day B, James T, Kozma R, Miller J, Unwin T (2005) Monitoring and evaluation of ICT in education projects: a handbook for developing countries. infodev/World Bank, Washington, DC
14. Görgens M, Kusek JZ (2009) Making monitoring and evaluation systems work: a capacity development toolkit. The World Bank, Washington, DC

15. Erl T (2007) SOA principles of service design (The prentice hall service-oriented computing series from Thomas Erl). Prentice Hall PTR, Upper Saddle River, NJ
16. Rudzajs P, Kirikova M (2009) IT knowledge requirements identification in organizational networks: cooperation between industrial organizations and universities. 18th International conference on information systems development, pp 187–199
17. Strazdina R, Kirikova M, Penicina L, Rudzajs P (2010) Knowledge requirements monitoring system: advantages for industry and university. Second international conference on information, process, and knowledge management, pp 120–125
18. Rudzajs P (2011) Towards automated education demand-offer information monitoring: the system's architecture. Selected Papers from workshops and doctoral consortium of the international conference on perspectives in business informatics research, BIR2011, pp 252–265
19. Rudzajs P (2008) Development of knowledge renewal service for the maintenance of employers' database. Bachelor thesis (in Latvian), Riga Technical University, Riga, Latvia
20. Rudzajs P (2011) Mediated competency comparison between job descriptions and university courses. Appl Comput Syst 47:48–56
21. Rudzajs P (2012) Towards automated education demand-offer information monitoring: the information extraction. Sixth international conference on research challenges in information science, pp 453–454
22. Ngo D, Bellahsene Z, Coletta R (2011) A flexible system for ontology matching. CAiSE Forum 2011, pp 73–80
23. Kiyavitskaya N, Zeni N, Cordy JR, Mich L, Mylopoulos J (2009) Cerno: light-weight tool support for semantic annotation of textual documents. Data Knowl Eng 68:1470–1492
24. Butters J, Ciravegna F (2008) Using similarity metrics for terminology recognition. Proceedings of the sixth international conference on language resources and evaluation (LREC'08). European Language Resources Association (ELRA), Marrakech, Morocco, pp 2817–2822
25. Ryu P-M, Choi K-S (2006) Taxonomy learning using term specificity and similarity. Computational Linguistics, pp 41–48

IT Governance: A Baseline Implementation Roadmap of a Bank in Thailand

Sureerat Saetang, Abrar Haider, and Andy Koronios

Abstract IT governance is one of the crucial aspects in the agenda of the modern organizations. Many leading organizations face an uncertain environment involving complexity in information technology architectures and infrastructure, and more demands for top management to validate and secure day-to-day operations. This research has shown the significant factors of successful IT governance implementation, which are strategic alignment and value delivery. It proposes these findings from a case study by showing a snapshot of how IT governance contributes to an enriched operational planning/management and strategic performance protocol in organizations. It scrutinizes these data to identify the influences on the implementation of IT governance in achieving business objectives and meeting needs and shaping ongoing implementation within current organizational contexts. Moreover, this paper provides hands-on material for practitioners to address their IT governance challenge, which could be a sample case study illuminating lessons learned for other organizations.

1 Introduction

Governance is a vital business concept, which maintains certain benefits for stakeholders, as well as reviews performance management, operational activities, and the role of a board of directors and the executives to ensure an organization follows best practice [1]. Corporate governance is a cluster of processes, practices, policies, regulations, and associations, along with relationships among various stakeholders, as well as organizational objectives, which influence the strategic direction of an

S. Saetang (✉) • A. Haider • A. Koronios
School of Computer and Information Science, University of South Australia,
Mawson Lakes Boulevard, Mawson Lakes, Adelaide, SA, Australia
e-mail: saesy005@mymail.unisa.edu.au; abrar.haider@unisa.edu.au; andy.koronios@unisa.edu.au

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_37,

organization and its management [2]. With respect to the importance of IT, governance is one of the main factors in organizations that have growing demands in the use of IT, IT infrastructure implementation, and various IT activities, which operate as the foundation of internal processes to support each business growing productively and rapidly [3]. According to the definition of IT governance, it is a subclass discipline of corporate governance that concentrates on information technology systems (ITS), business operations, and risk management [2]. Based on modern enterprise environments, it is highly important to have effective IT governance along with significant administrative performance [4, 5]. Thus, it is imperative to execute IT governance to strengthen organizations and empower responsible people in managing their work performance effectively. With this in mind, IT and business realize that IT governance is an investment instrument, which helps organizations to align uncontrollable business activities and complex IT [6], an uncertain business environment [7], and conflict in internal relationships [8]. Therefore, it is essential to have IT governance to support any business in encountering, managing, and overcoming chaotic business processes.

This study presents the findings from a case study and emphasizes IT governance, strategic IT alignment, and value delivery as the main factors at AAA bank by presenting a snapshot of how IT governance contributes to an enriched operational planning/management and improved strategic performance of the organization. It scrutinizes the bank to identify the influences or challenging problems concerning the implementation of IT governance in achieving business objectives and meeting its needs and shaping ongoing implementation in current organizational contexts. Firstly, however, the literature in relation to IT governance and those attributes that constitute effective IT governance will be discussed. Following that, this paper will then explain the details about the research methodology and the case study with research findings. Lastly, the conclusion of this paper will be summarized.

2 Literature Review

2.1 IT Governance

IT governance is an organizational ability managed by the board of directors, senior management, and IT management to assist them to execute the organizational structure, organizational direction, and IT strategy in order to assure the integration of business and IT [9]. The structure of an IT governance model consists of top management, organizational structures and processes [10], the responsibilities of the board of directors and the overall administration of organizational information management [11, 12], decision-making arrangements, alignment processes, and collaboration instruments [3]. Moreover, IT governance allows top management of business and IT to make decisions in planning the policies and processes of significant projects and examining results [13]. Thus, these components of IT governance provide appropriate levels of support to organizations in budgeting and management control systems by ensuring that the organization gets full capacity in its use of IT, keeps developing organizational strategies, and keeps achieving business objectives. Due to the capability of IT governance, it works as a tool of strategic business-IT alignment [14, 15]. Therefore, these factors and procedures lead IT governance to align the business model and IT implementations successfully and effectively.

2.2 Strategic Alignment

Focusing on the foundation of strategic alignment, there are three key areas, which are (1) business-information technology (IT) strategic alignment [14, 16, 17], (2) business-information systems (IS) strategic alignment [18, 19], and (3) business-information management (IM) strategic alignment [20]. Various analysts [21, 22] stated that IM is the alignment of business strategy and organizational resources only, with no relationship to strategic alignment. In the information management area, strategic alignment is concerned with the alignment of strategy and internal attributes, which is a senior management issue [23, 24] in developing organizational performance [25]. Moreover, strategic fit and functional integration are also the main fundamental components of strategic alignment [14, 26]. In implementing these components, strategic fit is considered as a dynamic process in the organizational environment, which always keeps adapting and changing in forming their internal systems (IS, IT infrastructure, and business processes) within internal mechanisms appropriate to achieving business objectives, along with various external mechanisms [14, 26]. Therefore, strategic alignment is the synthesis between external mechanisms (organizational marketplaces and competitors) and internal mechanisms (organizational administrative structures' components) of organizations. It typically regards the outcomes of decisions, which affect functional aspects of an organization [14, 26]. Therefore, it is clearly understood that IT is important to an organization because it works as an essential organizational function in supporting business and integrating diverse aspects with the right decision-making processes, thus achieving business objectives. In brief, the stability between organizational functions in knowledge and IT, along with the organizational business strategies, is defined as strategic alignment [26, 27].

2.3 Value Delivery

Values are those factors comprising people, cultures, and societies [28–30]. Given that IT exists in these environments, its overall function may be described in terms of supporting, optimizing, and developing organizations to gain a better business model. By doing so, values make people consider their relationships between IT and values in organizational environments. According to precise measurements and business value delivery for IT investments, it is vital for organizations to focus on various critical concerns, especially effort, cost, benefits, risk in IT-enabled business change, and innovation [31]. Regarding the value of IT definition, it is concerned with the value proposition through cycle delivery, by ensuring that IT delivers the assured benefit against the strategy by focusing on minimizing expenses, delivering the best IT value [32], and governing projects, operational processes, and practices that increase the opportunity for success in meeting key objectives such as top quality, low risk, on time, on budget, and inter alia [33]. These concerns lead to the emergence of a Val IT framework, which supports IT investment decisions and also comprises three domains: (1) value governance, (2) portfolio management, and (3) investment management [34]. With this in mind, the Val IT framework emboldens its users to have value delivery practices, which identify various types of investments, indicate and monitor key metrics, and rapidly acknowledge any change or fluctuation of concern by involving all stakeholders, allocating responsibilities, monitoring, evaluating, and developing continuously [35].

3 Research Methodology

A single case study was selected for this research [36], which mainly focuses on how IT governance addresses the issues of strategic IT alignment and value delivery within one Thai financial institution, AAA (an alias). This bank has adopted and implemented current technologies, vital business processes, and new technologies, which influence the organizational objectives. Particularly, this case study research highlights the interest in using qualitative research method [37] by adopting the "how" questions in dynamic existing environments [36–39] through examining IT governance. Additionally, this research chose to employ an in-depth interview approach to examining the organization and IT governance implementation in its usage contexts, which contains the components of technologies and complex human/social mechanisms [40]. Moreover, the method of this research identifies numerous serious confusions and complex relationships that have happened within the organization, which lead to provide better understanding and more organizational awareness [41], and also achieve the research objective. Therefore, these well-supported factors from this case study could lead to form good IT governance practices. Meanwhile, this research employs an interpretive qualitative approach in collecting data from involving organizations, by conducting in-depth semi-structured interviews (face-to-face meetings, email, and telephone follow-ups), personal observation in workplaces, and qualitative questionnaire surveys. These processes and approaches support the bank's opportunities to examine IT governance in AAA, which express the purpose of this research objective by focusing on strategic IT alignment, and value delivery as the main factors in illuminating the main question, which is: "How does implementation of an IT governance framework contribute to the value profile of the IT infrastructure in achieving business objectives and meeting its needs?" This paper notes the findings from the case study, which emphasizes IT governance, strategic IT alignment, and value delivery as the main factors at AAA bank. This is achieved by presenting a snapshot of how IT governance contributes to an enriched operational planning/management and the strategic performance of the organization. It scrutinizes the research, relevant information, and staff responses to identify the influences or challenging problems facing implementation of IT governance procedures in achieving business objectives and meeting its needs and shaping ongoing implementation within current organizational contexts.

Research in AAA bank in Thailand was conducted over a period of 2 months with five face-to-face interviews. All interviews were digitally voice recorded and transcribed, with additional and updated information drawn from AAA bank's website. The informants were senior IT strategy and policy specialists, a senior software quality assurance specialist, the head of IT strategy and policy, the head of software quality assurance, a senior IT architect specialist, a senior regulatory and policy compliance specialist, a senior information systems auditor, and a senior IT resource and portfolio management specialist. Data were collected from all interviews in a repetitive process, which collated empirical data, theoretical views, and relevant documentation [38]. Furthermore, this research ensured the complete picture by directly quoting significant comments from the interviewees [42]. Thus, there is the potential to create a mind map to depict the empirical data [43]. Subsequently, the whole picture of each informant's statements and mind maps was analyzed along with theoretical perceptions. In addition, this data analysis process was carried through to the final assessment, which helped to explain the overall findings from the case studies [38].

4 Case Study

4.1 Background of the AAA Bank

In 1945, AAA was established. Today, there are approximately 16,000 employees, and the bank has a registered capital of Baht 30,000 million (USD 99 million) and a return on assets (ROA) in the first half of 2011 that has been assessed at 1.7 %. It has been indexed with the bank of Thailand, which operates as Thailand's central bank, e.g., designating monetary policy, examining financial systems, supporting banking assistance to the government and financial institutions, releasing securities and new prints of banknotes, and controlling the foreign exchange rate in Thailand. AAA is named as one of the main local commercial banks in Thailand and has been listed on SET (The Stock Exchange of Thailand) since 1976. AAA is in the top five largest commercial banks as defined by total assets, loans, and deposits, and it has 16 % of market share. In addition, there are more than 800 branches across Thailand with eight overseas offices in the USA, Hong Kong, Cayman Islands, China, and Japan. Currently, there are over than 7,000 automated teller machines (ATM) across Thailand.

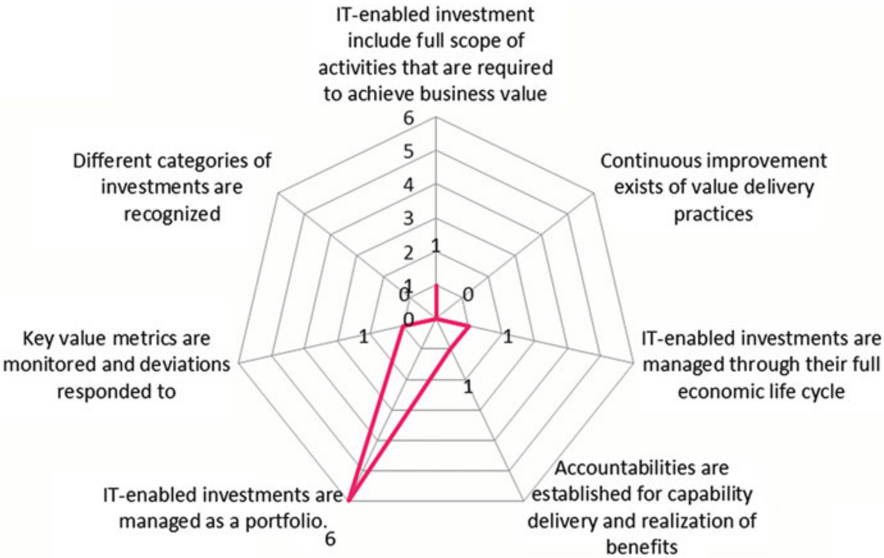


Fig. 1 Outlook for IT investment

4.2 IT Governance

AAA updates IT governance frameworks by regularly adopting new versions of software and other systems. However, AAA is always careful in selecting new functionalities in the systems prior to adopting them in the business operations, in order to avoid risks and related problems. A senior IT strategy and policy specialist commented:

> In order to maintain an up to date system, we always follow up new versions of the framework. We have updated our **Information Technology Infrastructure Library (ITIL)** from version 2 to version 3 for example. However, we are always concerned with new functions. If there are some new functions, which are not suitable to our business, we will not deploy them; but it rarely happens. We mainly adopt frameworks, which are available in the market such as **Control Objectives for Information and related Technology (Cobit)**, ITIL, and **Capability Maturity Model Integration (CMMI)**. We use all of these as we have different business areas. In my department, we mainly focus on Cobit as we have the responsibility of measuring the maturity level of other sections of the bank. Cobit does not mandate working methods. It only provides the steps to follow to determine what to do. In respect of the IT architecture department, they have to adapt CMMI for it to work in their areas. Consequently, these frameworks are infrequently changed. In reference to project management, there is a Project Management Provenance team, which looks after all those processes. If there are new technologies and approaches, they will study them carefully prior to any implementation.

Figure 1 explains the outlook for IT investment. In the globalized economy of the early twenty-first century, most people in AAA chose the only outstanding answer

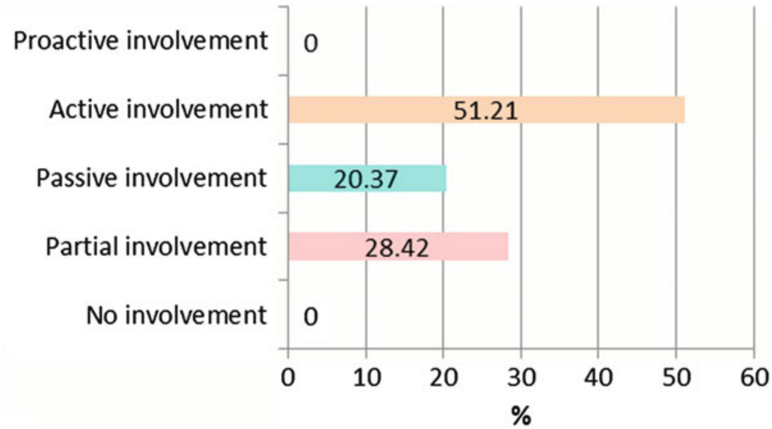


Fig. 2 Understanding of, and participation in, IT governance at all levels of the organization

from the cohort of options, which is IT-enabled investments are managed as a portfolio. However, other choices were recognized in different answers. So, it is easily understood that the portfolio is a vital consideration for AAA in gaining more generous returns from IT investments. It seems likely that top management people strongly encourage and support efforts to optimize IT. And now in 2012, there is no doubt that AAA is one of the most profitable and leading innovative Thai banks.

4.3 Strategic Alignment

In relation to strategic alignment mainly concerning the role of the CEO and CIO, a senior regulatory and policy compliance specialist commented on the policies by suggesting:

> As I have seen in my role, the CEO always gives directions. According to the policies, AAA always lays down the policies and sets conditions by following the Bank of Thailand and IT management. Policies were passed on to different hierarchical levels of management and were considered by the CIO, management committees, board of directors, and the CEO respectively. In terms of policies, there are substantial matters, which have been discussed, such as how to manage matters properly and correctly. The CEO and CIO always provide more ideas, feedback, directions, and advice in addition to existing issues. According to maintenance policies, there are some reviews that are achieved by maintaining the policies and getting approvals from the leadership.

Referring to Fig. 2, active involvement gains the highest understanding and participation from respondents in IT governance at all levels of the organization. AAA encourages staff to have a strong focus on IT governance. It is incumbent upon staff to deliver, respond to, and introduce new developments in IT governance in a comprehensive manner. It, therefore, leads them to understand and participate more

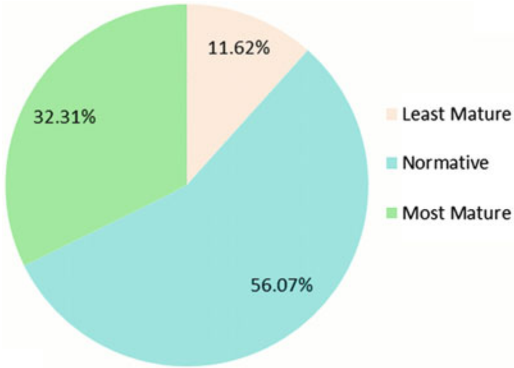


Fig. 3 Value of IT service delivery

fully. AAA expects that staff will improve their initiatives and demonstrate creativities in memorizing the IT government requirements at various levels of the company. Then, they can integrate IT governance seamlessly. Regrettably, passive and partial involvements remain a concern for a minority of respondents.

4.4 Value Delivery

According to the strategic level, top management has realized the importance of value delivery so that AAA has an effective IT governance adoption strategy because of its reliable processes. Due to the size of AAA, it needs skilled people to work there and to support the numerous difficult projects that they have to operate simultaneously. This in turn leads AAA to recruit additional people on a regular basis to serve operational needs and duties of increasing complexity. This is an important impact of IT governance adoption in AAA. Moreover, some full-time employees are not skilled enough, while others have poor technical skill and less experience in handling these tasks. Therefore, it is important to hire people from an outsourcing vendor to oversee this growth. The head of IT strategy and policy commented:

> I think all our processes and management are well organized; they are run in an orderly and systematical way. The processes are auditable, traceable, scalable, and able to be escalated. Therefore, the results are produced transparently and with high quality. I think that the impact of IT governance adoption provides for good operations that can be run systematically and automatically, within the timeline. The processes also involve a short waiting time, which is required to allow particular authorized people to decide on, approve and proceed with their decisions, prior to providing command and control, direction, commitment, authorization, and communication. It assists to gain an agreeable consensus from the top management. However, IT governance adoption also provides a poor effect in terms of recruiting experts to operate long term processes for big projects.

In regard to the value of IT service delivery, as depicted in Fig. 3, Normative is the answer selected by most respondents. This indicates that AAA adopts a moderately powerful IT service delivery, which could be at the same standard or level as the other Thai banks. In fact, IT service delivery in AAA is similar to other Thai banks. And AAA is not powerful enough to be critical of others. Moreover, there are more successful banks in the Thai market, and this brings intense pressure for change and also shapes AAA's IT service delivery. The other banks' excellent returns and products, compared to the value of IT service delivery in AAA, do not show such exceptional performance and outcomes. This has elicited useful information to renovate bank processes and make the forces driving the structural change to introduce a gain that provides a technological advance. Senior managers have to find a new plan for the firm's survival. They may be required to do a lot more regarding prospects of advancement and concerning continuous improvement that result in innovation and better ways of doing things. This result needs to be mainly driven by information and communications technology evolution, which has immensely improved the rapidity of improvement and shortened the cost at, which information, money, and people progress. It shows that AAA has an optimistic attitude towards productive progress and developing its practices in respect of IT service delivery.

5 Conclusion

This paper encapsulates the integration of business and IT by implementing IT governance in order to facilitate and execute the business processes and thus grow the organization effectively. In general, organizations must focus on business objectives beforehand and analyze their effect prior to deploying technologies in their efforts to gain successful outcomes. This paper proposes the findings from a case study that emphasizes IT governance, strategic IT alignment, and value delivery as the main factors at AAA bank. It did this by presenting a snapshot of how IT governance contributes to enriched operational planning, management, and strategic performance of the organization. That is, the staff in positions of influence in AAA bank appear to have made positive evaluations of IT governance implementation effectiveness, and these results, and the lessons learned, could strengthen ongoing IT governance implementation, strategic alignment, and value delivery of other organizations and result in divergent management, control, and performance management as well. Furthermore, the present research may provide valuable advice to organizations that are planning to implement IT governance. Finally, this research suggests scrutinizing and identifying the influences and problems facing implementation of IT governance in achieving business objectives, such as by meeting needs, and shaping ongoing implementation in current organizational contexts as well.

In keeping with the data of this research, it emerges that IT governance implementation is an approach to strategic alignment and value delivery in validating processes such as through sharper decision-making, stronger business-IT alignment, and bolder internal collaborations processes. Consequently, it develops processes of IT governance implementation for the banking industry, which could promote more sophisticated business performance in achieving business objectives. Due to the immaturity of the present research, it is difficult to make strong assertions regarding the outcomes, which are only based on a single case study. For that reason the author intends to conduct further productive research in this area.

References

1. Fama EF, Jensen MC (1983) Separation of ownership and control. J Law Econ 26:301–325
2. Kooper MN, Maes R, Lindgreen EEOR (2011) On the governance of information: introducing a new concept of governance to support the management of information. Int J Inf Manage 31(3):195–200
3. Weil P, Ross JW (2004) IT governance—how top performers manage IT decision rights for superior results. Harvard Business School Press, Boston, MA
4. Berghel H (2005) The two sides of ROI: return on investment vs. risk of incarceration. Commun ACM 48(4):15–20
5. Pennington R, Tuttle B (2007) The effects of information overload on software project risk assessment*. Decis Sci 38(3):489–526
6. Chan YE (2002) Why haven't we mastered alignment? The importance of the informal organization structure. Manage Inf Syst Q Exec 1(2):97–112
7. Ciborra C (1997) De Profundis? Deconstructing the concept of strategic alignment. Scand J Inf Syst 9(1):67–82
8. Maes R (2007) Information management: an integrative perspective. In: de Vries EJ, Huizing A (eds) Information management: setting the scene, Elsevier Ltd, Amserdam, pp 11–29
9. Van Grembergen W (2004) Strategies for information technology governance. IDEA Group Publishing, Hershey, PA
10. IT Governance Institute (2003) Board briefing on IT governance 2003, 2nd edn. http://www.isaca.org/Content/ContentGroups/ITGI3/Resources1/Board_Briefing_on_IT_Governance/26904_Board_Briefing_final.pdf. Accessed 6 Apr 2012
11. O'Donnell E (2004) Discussion of director responsibility for IT governance: a perspective on strategy. Int J Account Inf Syst 5(2):101–104
12. Bernroider EWN, Ivanov M (2011) IT project management control and the control objectives for IT and related technology (CobiT) framework. Int J Proj Manage 29(3):325–336
13. Weil P, Broadbent M (1998) Leveraging the new infrastructure: how market leaders capitalize on information technology. Harvard Business School Press, Boston, MA
14. Henderson JC, Venkatraman N (1999) Strategic alignment: leveraging information technology for transforming organizations. IBM Syst J 38(2&3):472–484
15. Hirschheim R, Sabherwal R (2001) Detours in the path toward strategic information systems alignment. Calif Manage Rev 44(1):87–108
16. Alavi M, Leidner DE (2001) Review: Knowledge management and knowledge management systems: conceptual foundations and research issues. Manage Inf Syst Q 25(1):107–136
17. Pollalis YA (2003) Patterns of co-alignment in information-intensive organizations: business performance through integration strategies. Int J Inf Manage 23(6):469–492
18. Sabherwal R, Chan YE (2001) Alignment between business and IS strategies: a study of prospectors, analyzers, and defenders. Inf Syst Res 12(1):11–33
19. Segars AH, Grover V (1998) Strategic information systems planning success: an investigation of the construct and its measurement. Manage Inf Syst Q Exec 22(2):139–163
20. Rogers PR, Bamford CE (2002) Information planning process and strategic orientation: the importance of fit in high-performing organizations. J Bus Res 55(3):205–215
21. Burgelman RA, Doz YL (2001) The power of strategic integration. MIT Sloan Manage Rev 42(3):28–38
22. Hooff BVD, Vijvers J, Ridder J (2003) Foundations and applications of a knowledge management scan. Eur Manage J 21(2):237–246
23. Luftman J (2003) Assessing IT/business alignment. Inf Syst Manage 20(4):9–15
24. Moody KW (2003) New meaning to IT alignment. Inf Syst Manage 20(4):30–35
25. Miles RE, Snow CC (1978) Organizational strategy, structure and process. McGraw-Hill, New York
26. Wang WT, Belardo S (2009) The role of knowledge management in achieving effective crisis management: a case study. J Inf Sci 35(6):635–659

27. Porter ME (1996) What is strategy? Harv Bus Rev 74(6):61–78
28. Rokeach M (1973) The nature of human values. The Free Press, New York
29. Maslow A (1968) Toward a psychology of being. Van Nostrand Reinhold, New York
30. Schwartz SHS (1992) Universals in the content and structure of values: theoretical advances and empirical tests in 20 countries. Adv Exp Soc Psychol 25:1–65
31. Symons C (2007) From IT governance to value delivery. http://www.atkinson.yorku.ca/~bresani/From%20IT%20Governance%20to%20value%20delivery.pdf. Accessed 28 Apr 2012
32. ISACA (2011) Val IT 2.0: the challenge of IT value delivery. http://www.arquivos.comuv.com/grcmeeting/IT_GRC_Meeting_arquivos/apresentacao/ILC-IT-GRC-Meeting-2011.vENG_.FINAL_.pdf. Accessed 12 Apr 2012
33. Lew D, Stephen P (2006) Conducting an IT governance assessment. http://www.sfisaca.org/events/conference06/Presentations/G12.pdf. Accessed 20 Apr 2012
34. ISACA (2012) Val IT FAQs. http://www.isaca.org/Knowledge-Center/Val-IT-IT-Value-Delivery-/Pages/Val-IT-FAQ.aspx. Accessed 29 Apr 2012
35. ISACA (2012) Val IT: based on COBIT. http://www.isaca.org/Knowledge-Center/Val-IT-IT-Value-Delivery-/Documents/Val-IT-Brochure.pdf. Accessed 29 Apr 2012
36. Yin RK (2003) Case study research: design and methods. Sage Publications, Thousand Oaks, CA
37. Benbasat I, Goldstein DK, Mead M (1987) The case research strategy in studies of information systems. Manage Inf Syst Q 11:369–386
38. Eisenhardt KM (1989) Building theories from case study research. Acad Manage Rev 14(4):532–550
39. Walsham G (1995) Interpretive case studies in IS research: nature and method. Eur J Inf Syst 4(2):74–81
40. Tyrell S (2002) Using information and communication technology in healthcare. Radcliffe Medical Press, Oxford
41. Bhattacharjya J, Chang V (2006) Adoption and implementation of IT governance: cases from Australian Higher Education. The 17th Australian conference on information systems, Adelaide, Australia
42. Pratt MG (2009) From the editors. For the lack of a boilerplate: tips on writing up (and reviewing) qualitative research. Acad Manage J 52(3):856–862
43. Layngley A (1999) Strategies for theorizing from process data. Acad Manage Rev 24(4):691–710

DBMS Application Layer Intrusion Detection for Data Warehouses

Ricardo Jorge Santos, Jorge Bernardino, and Marco Vieira

Abstract Data Warehouses (DWs) are used for producing business knowledge and aiding decision support. Since they store the secrets of the business, securing their data is critical. To accomplish this, several Database Intrusion Detection Systems (DIDS) have been proposed. However, when using DIDS in DWs, most solutions produce either too many false-positives (i.e., false alarms) that must be verified or too many false-negatives (i.e., true intrusions that pass undetected). Moreover, many approaches detect intrusions a posteriori which, given the sensitivity of DW data, may result in irreparable cost. To the best of our knowledge, no DIDS specifically tailored for DWs has been proposed. This paper examines intrusion detection from a data warehousing perspective and the reasons why traditional database security methods are not sufficient to avoid intrusions. We define the specific requirements for a DW DIDS and propose a conceptual approach for a real-time DIDS for DWs at the SQL command level that works transparently as an extension of the Database Management System (DBMS) between the user applications and the database server itself. A preliminary experimental evaluation using the TPC-H decision support benchmark is included to demonstrate the DIDS' efficiency.

1 Introduction

Intrusion is a set of actions that attempt to violate the integrity, confidentiality, or availability of a system [4, 11, 16]. Intrusion Detection Systems (IDS) detect unauthorized access in an automated way based on two main approaches: *misuse*

R.J. Santos (✉) • M. Vieira
CISUC – DEI – FCTUC, University of Coimbra, Coimbra, Portugal
e-mail: lionsoftware.ricardo@gmail.com; mveira@dei.uc.pt

J. Bernardino
CISUC – DEIS – ISEC, Polytechnic Institute of Coimbra, Coimbra, Portugal
e-mail: jorge@isec.pt

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_38,

detection, matching user actions with well-known predefined attack patterns; and *anomaly detection*, analyzing user actions to find deviations from determined normal behavior. Many IDS have been proposed, but they mainly focus on the network or operating system level. These IDS are ineffective to detect application level attacks, as they are perceived as authorized commands executed by authorized users and because they do not have the knowledge of application level semantics, required separation of duties, and normal working scope of the users. Thus, the best way to avoid these attacks is to place an additional ID layer at the database level, i.e., a Database IDS (DIDS).

Data Warehouses (DWs) are used for producing business knowledge and decision support purposes, making them the core of enterprise sensitive data, which in some cases is worth millions of dollars. Given the ad hoc and unpredictable nature of decision support queries and the diversity of their data access patterns, the boundary between normal and anomalous behavior is frequently fuzzy, i.e., it is very difficult to determine *normal* user behavior and *probable attack* behavior, making it very hard to efficiently model attack patterns.

Although several DIDS have been proposed in the past, they are poor at detecting intrusions in DWs without typically resulting in too low true intrusion detection rates where most intrusions pass undetected or too many false alarms are raised (i.e., false-positive alerts) [12, 13, 16, 18]. In fact, in the last case, the number of generated alerts is typically so large that it often leads to a huge waste of time and limited resources. Although alert correlation techniques have been proposed for filtering the most relevant alerts [1, 3, 12, 13], in highly heterogeneous environments such as DWs, the number of alerts is simply just too high to be checked [12, 16]. This jeopardizes the credibility and feasibility of the whole IDS [1, 8, 12, 18].

Many business models using web-based infrastructures work on 24/7 schedules. Such enterprises require 24/7 management, which implies that decision makers require continuous access to decision support means such as Data Warehouses (DWs). To allow decision makers to have this kind of access, DWs have been shifting to the cloud computing paradigm, enabling them to be available at any time from any location. However, this creates a key security issue since it imposes providing a mean for accessing their databases that can be reached from outside the enterprise. Given the uniqueness of certain features intrinsic to data warehousing environments (explained in Sect. 3) which do not exist in other types of database systems, we argue that DWs require specifically tailored DIDS. To the best of our knowledge, no such DIDS has been proposed so far.

In this chapter, as main contributions, we refer the main types of DIDS and propose the conceptual design of a DIDS tailored for DWs. We present the most relevant features for analyzing user actions in data warehousing environments and propose misuse and anomaly detection techniques as a support for DIDS. Given that not all intrusions present the same risk of data damage or disclosure, we also propose a risk exposure method to rank the generated intrusion alerts. This enables taking care of intrusions with a potentially higher risk to the enterprise more rapidly than other DIDS using alert correlation techniques. Our DIDS works as an extension of any DBMS, adding intrusion detection capabilities to the native database server functionalities. In practice, it acts as an additional feature at the application layer that enables intrusion detection between DW user applications and the database transparently without jeopardizing their joint functionality. To the best of our knowledge, this is the first proposal presenting the requirements and a conceptual approach for a DIDS specifically tailored for DWs. It is also the first to use risk exposure on alert generation to deal with intrusions more efficiently.

The remainder of this chapter is organized as follows. Section 2 presents background and related work on DIDS. Section 3 characterizes a typical DW environment and the security issues that concern intrusions in DWs. Section 4 presents the features that we have determined as the requirements for building a DIDS for DWs and the description of the proposed DIDS itself. In Sect. 5 an experimental evaluation is presented using the TPC-H decision support benchmark. Finally, Sect. 6 presents our main conclusions and points out future work.

2 Background and Related Work

Main commercial DBMS such as Oracle and Microsoft SQL Server provide Fine-Grained Access Control (FGAC) for applying data access policies to the database itself, rather than within applications. Thus, security needs to be built only once in the database instead of being implemented across applications. However, while FGAC can restrict intrusive activities according to the defined data access policies, it cannot detect intrusions. In this sense, our approach complements FGAC at the DBMS level.

The work in [9] presented an anomaly detection approach for real-time database systems using time signatures. Other anomaly detection approach in [15] is based on statistical functions representing data relations. However, both approaches focus on anomalies in data changes rather than querying actions. Moreover, the authors of [15] suggest the intrusion detection procedure should run once a day or even once a week in an a posteriori mode instead of performing online intrusion detection. This is not adequate for DWs. In [2], database transactions are represented by directed graphs describing the SQL command types (select, insert, delete, etc.) used for malicious data access detection. This approach cannot handle ad hoc queries and works at the coarse-grained transaction level as opposed to the fine-grained query level. The solution proposed in [1] uses tree kernel-based learning and clustering for extracting context modeling of SQL instructions and anomaly detection. A Role-Based Access Control (RBAC) mechanism has been proposed in [5]. Data mining techniques are used, namely, classification and clustering, against SQL instructions stored in database audit files to deduce role profiles of normal user behavior. A limitation of this approach is that it cannot extract correlation among queries in transactions. Another RBAC DIDS solution using unsupervised machine learning was also proposed in [14], analyzing read–write operations. Since these solutions are role-based, they work at a higher coarse-grained level than the user-based profiling proposed in our approach.

Detecting attacks by comparison with a set of known legitimate database transactions, summarizing SQL statements into compact access patterns named as fingerprints, is the focus of [8]. Analyzing the sequence of user commands for detecting ID is used in [7]. Profiling the data accessed by users to try to determine their intent is an approach used in [10] using statistical learning algorithms. They argue that analyzing what the user is looking for (i.e., what data) instead of analyzing how she/he is looking for it (i.e., which SQL expressions) is more efficient for anomaly detection. In our chapter, we integrate both these views.

Several correlation techniques have been proposed to reduce the number of false-positives. Data correlation among transactions is used in [3], recurring to data mining using read–write dependency rules. However, this system can only match well-defined read–write patterns; it is unable to detect malicious behavior in individual commands. Similarly, in [16], the dependency between attributes is used to generate rules based on identified malicious transactions, considering the sensitivity of each data item to attribute different weights. In [12] an Adaptive Learner for Alert Classification is described, using machine learning to classify alerts and correlate them. Reference [19] presents an automatically tuning IDS, filtering alerts according to a detection model which is tuned on the fly according to the feedback of decisions made by a security operator. Although these proposals seem to improve false-positive rates, each attack in DWs has unique characteristics, which makes it difficult to define or find alert correlation; actions that are normal in certain environments may be malicious in others [12].

3 Data Warehouse Security vs. Intrusions

The DW database usually stores the complete history of a business, which is updated by ETL (Extraction, Transformation, and Loading) tools. The ETL processes extract relevant data from the transactional source systems, clean and transform the data, and load the final formatted data into its database. Most DW database schemas are star schemas [6], where business facts are stored in a central table called *Fact Table* (e.g., Sales fact table), and the tables containing the business descriptors are called *Dimension Tables* (e.g., Customer and Product tables). Dimension tables are usually small in size (less than 10 % of DW total storage space) when compared with fact tables, which are typically huge in size (at least 90 % of DW total storage space) and have an enormous amount of rows (millions or billions) [6]. From a usability perspective, DWs have the following well-known assumptions [6]:

- Business facts are mainly stored in numerical-typed attributes within fact tables; since fact tables typically take up at least 90 % of the DW total storage size, we can state that DW databases are mainly composed by numerical values.
- Typically, there are a relatively low number of DW users (a few tens).
- End users can only execute read-only instructions, i.e., they are not allowed to change data. Database Administrators (DBAs) and ETL users may execute any operation on data.

- More than 90 % of user actions are decision support queries (i.e., SELECT statements, even in real-time DWs), mainly against fact tables.
- Automated commercial ETL tools are usually the common way to update DW databases. These tools typically function in very well-defined schedules and with predefined frequencies. The amount of data which is loaded into the DW in each update is often also predefined.
- Although decision support queries may typically access huge amounts of data, the query response usually results in a small dataset with a few hundred bytes and a relatively low number of columns (no more than a few tens).
- Reporting (i.e., periodically running reports for answering fixed predefined decision support queries) is a typical action in DWs.
- Many decision support queries are issued in ad hoc manner, which makes them mostly unpredictable. Most queries are CPU intensive and can take up to hours.

Given typical data warehousing environments, there are mainly three types of attacks mobilized against DWs:

1. *Attacks aiming for corrupting data*. In these attacks, the intruder seeks actions that compromise the database's integrity, such as maliciously modifying or deleting the data in a given database object (e.g., such as table or view).
2. *Attacks aiming for stealing information.* In these types of attacks, the intruder is focused on breaking confidentiality issues, such as stealing business information, rather than damaging the contents of the database.
3. *Attacks aiming for making the DW unavailable.* These attacks focus on making database services unavailable, i.e., Denial of Service (DoS) attacks. Some examples are erasing a database object (e.g., database tablespace), flooding database services with a huge amount of requests (e.g., executing a large number of simultaneous instructions that process and/or return huge amounts of data, thus creating processing bottlenecks at the server and jeopardizing network bandwidth with their responses), halting or crashing database server instances, etc.

4 Data Warehouse Intrusion Detection System

The DIDS architecture is shown in Fig. 1, functioning as an extension of the DBMS. Intrusion detection is handled at the DBMS command level, in two moments, according to the following: whenever the DBMS is required for executing a command, the DIDS analyzes the command before it is executed, by the Command Analyzer (step 2), and analyzes the response after it is computed and before returning it to the user which required its execution, by the Response Analyzer (step 9).

The DW Security Administrator manages the Security Manager Interface, which allows visualizing and managing the DIDS-DW database (user profiles, intrusion detection actions and rules, and intrusion history), as well as monitor intrusion alerts and define the actions to respond to each detected intrusion. The DBA manages all DW database(s) data structures and objects such as datafiles, tables,

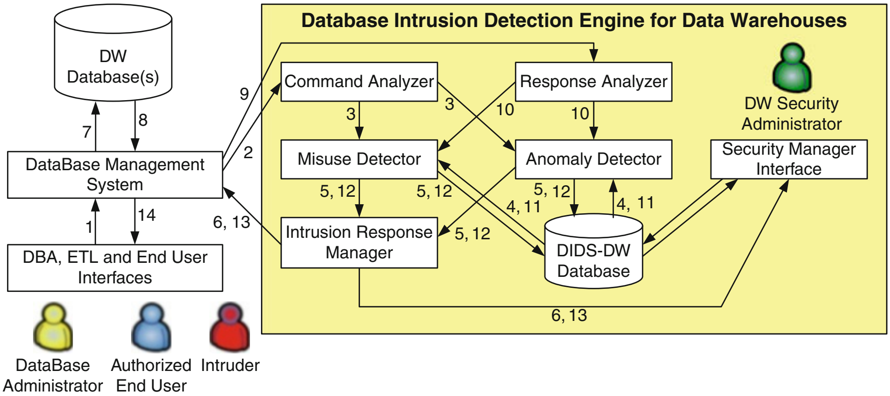


Fig. 1 The conceptual architecture of the DIDS for DWs

indexes, and views and performs their maintenance. The Authorized End User is a regular DW end user who is interested in querying data for decision support purposes. The Intruder represents both inside and outside attackers with intention to steal or damage data (e.g., as a masqueraded insider by using an authorized user's interface after taking possession of a valid end user username/password login; as an outsider if she/he is able to bypass a database connection with open data access policies—such as a typical web connection—by using SQL injection techniques, etc.).

The sequence of steps is labeled in Fig. 1 and described as the following: a user submits a command through any type of interface (internal or external to the DW) to the DBMS for execution (step 1). Before executing it, the DBMS will send the command text to the Command Analyzer (step 2), which parses the command and splits it into the interesting features for misuse and anomaly detection, and simultaneously passes this information to the Misuse Detector and Anomaly Detector (step 3). These detectors retrieve the existing user profiles and intrusion detection rules and actions from the DIDS-DW Database (step 4) and apply the misuse and anomaly detection algorithms. Afterwards, the detectors will decide if the command is probably an intrusion or not. In the affirmative case, they will pass on the intrusion alert and correspondent action to be taken to the Intrusion Response Manager (IRM), storing the intrusion features in the DIDS-DW Database for future reference (step 5). If the detectors conclude that the command is not an intrusion, they will simply update the user profiles and feature statistics in the DIDS-DW Database and discard notifying the IRM.

Whenever the IRM is notified of an intrusion alert, it assesses the alert's risk exposure, and its consequent alert is sent to the Security Manager Interface to notify the DW Security Administrator (step 6). The IRM takes actions through the DBMS by pausing or killing the user command execution, or killing the user session, either automatically or on request of the DW Security Administrator after she/he has seen the alert information and decided what action should be taken (step 6). For paused commands, if the DW Security Administrator decides it is a false alert, the intrusion record referring to that alert previously stored in the DIDS-DW Database by step 5 is deleted and command execution is resumed. If the IRM decides the user command is not an intrusion, it notifies the DBMS to execute it against the DW (step 7). After the user command response has been computed by the DBMS (step 8), the response itself is analyzed by the Response Analyzer, before returning it to the user interface which requested it (step 9), which extracts the interesting response features for intrusion detection and passes them on to the Misuse Detector and Anomaly Detector (step 10). These detectors will then repeat what occurred in steps 4, 5, and 6, as steps 11, 12, and 13. Finally, if the IRM considers the user command as a nonintrusion or as an intrusion (step 13), the computed results are, respectively, either sent back to the user interface which requested them or eliminated and given back a null response (step 14).

4.1 Alert Classification, Risk Exposure Assessment, and Response Handling

Many IDS evaluate the probability of a given user action being suspicious to classify that particular action (or set of actions to which it belongs) as an intrusion; when that probability exceeds a predefined threshold, an alert is generated. When those thresholds are too low, a huge amount of alerts is typically generated, which mostly turn out to be false alarms. Contrarily, when the thresholds are too high, many true intrusions pass undetected. Given the sensitivity of DW data, it is preferable to have low thresholds and consequently check the generated alerts because the potential cost of undetection is often considered too high or even unacceptable.

Alert correlation techniques such as [12, 13, 19] try to improve the efficiency of IDS when the number of generated alerts is large. These techniques filter the alerts to determine which are worthy of checking from those that are probably false alarms. We argue that alert correlation is not the best way to define which alerts should be checked and in which order of priority. Since the value of DWs resides on the fact that they store the secrets of the business, the resulting impact from an intrusion is closely linked with what data was exposed or corrupted. There can be intrusions that have been positively correlated for checking but have a small impact on the enterprise (e.g., the exposed or damaged data is not very sensitive), while alerts referring to true intrusions with high impact are ignored by correlation techniques.

To avoid this from happening, we propose an approach for ranking alerts according to a measure of risk exposure. Given a user action, *risk exposure* is a function of both the *probability* that action has of being an intrusion and the *impact* that action may have, i.e., the potential magnitude of the cost for the enterprise related to the damage or disclosure of the sensitive data which the action affects. Risk analysis consists on ranking the alerts given their computed risk exposure, according to a matrix similar to Fig. 2. This allows ensuring that the security staff will treat the most significant intrusions (given alerts with higher risk exposure) prior to intrusions that might produce less important damage, thus treating intrusion alerts more efficiently.

Probability \ Impact	Very Low	Low	High	Very High
Very High	High	High	Very High	Critical
High	Low	High	High	Very High
Low	Very Low	Low	High	High
Very Low	Very Low	Very Low	Low	High

Fig. 2 The risk exposure matrix

To define the probability of each type of alert given the feature that generated it, and filtering user, value-based and/or temporal-based conditions, the DW Security Administrator may define rules with the following syntax (clauses in brackets are optional, where those appearing with | are values to be chosen from):

```
1. DEFINE PROBABILITY AS VeryLow|Low|High|VeryHigh
2. ON FEATURES {FeatureName1, FeatureName2,…}, AllFeatures
3. [WHERE {List of filtering conditions}]
4. [WHEN {List of time-based conditions}]
5. [FOR USERS {UserID1, UserID2,…}, AllUsers,
     USERS WITH ROLES {UserRole1, UserRole2, …}]
```

The rule allows defining various probabilities to the same feature in case the DW Security Administrator considers it appropriate. For example, in systems such as banking and e-governance applications, the number of allowed consecutive unsuccessful login attempts for all users typically ranges from three to five. It is considered common to accept two consecutive unsuccessful attempts followed by a successful attempt as a nonintrusion, while more consecutive unsuccessful attempts indicate a possible intrusion tentative. Thus, the probability of an intrusion given the number of consecutive failed login attempts can be defined by the following rules:

```
DEFINE PROBABILITY AS Low
ON FEATURES
   #ConsecFailedLoginAttempts
WHERE
   #ConsecFailedLoginAttempts=3
FOR USERS AllUsers

DEFINE PROBABILITY AS High
ON FEATURES #ConsecFailedLoginAttempts
WHERE #ConsecFailedLoginAttempts=4
FOR USERS AllUsers

DEFINE PROBABILITY AS VeryHigh
ON FEATURES
   #ConsecFailedLoginAttempts
WHERE
   #ConsecFailedLoginAttempts>=5
FOR USERS AllUsers
```

Although we leave the assessment of the probability entirely to the DW Security Administrator (for she/he has the knowledge about the DW and its context and, thus, better judgment for defining these values), we recommend that for the first implementation of the DIDS for DWs a `High` probability should be given to all features. After the learning phase of the DIDS defined by the DW Security Administrator, the features' probability should be set accordingly to the alerts that they generated that turn out to have produced a higher or lower true-positive and false-positive rates.

The assessment of the impact caused by a user action is based on *which*, *how much*, and *when* sensitive data can be exposed or damaged by the user command. This assessment is also left entirely to the DW Security Administrator, as it depends on the nature and structure of each DW itself, which is mostly unique in each real-world context. It is managed by using the following rules, which allow defining the impact coefficient for the actions ranged by each command, valid for the list of nominal-based, value-based and/or temporal-based conditions:

```
1. DEFINE IMPACT AS VeryLow|Low|High|VeryHigh
2. ON FEATURE {FeatureName1, FeatureName2, …}, AllFeatures,
3. [ON COMMAND Insert, Update, Delete, Select,CreateAll,
   DropAll, AlterAll,
   CreateTable, DropTable, AlterTable,
   CreateIndex, DropIndex, AlterIndex,
   CreateProcedure, DropProcedure, AlterProcedure,
   CreateFunction, DropFunction, AlterFunction,
   CreateView, DropView, AlterView,
   CreateTrigger, DropTrigger, AlterTrigger]
4. [WITH COLUMNS {Column1, Column2, …}, AllColumns]
5. [WHERE {List of filtering conditions}]
6. [WHEN {List of time-based conditions}]
7. [JOINED WITH {Column1, Column2, …}, AllColumns]
8. [FOR USERS {UserID1, UserID2,…}, AllUsers,
     USERS WITH ROLES {UserRole1, UserRole2, …}]
```

To define which responses should be taken given the risk exposure matrix, the DW Security Administrator may define rules as the following:

```
1. GIVEN RISK EXPOSURE AS VeryLow|Low|High|VeryHigh|Crit
   ical
2. OF Misuse|Anomaly
3. ON FEATURE {FeatureName1, FeatureName2, …}, AllFeatures
4. [WHERE {List of filtering conditions}]
5. [WHEN {List of time-based conditions}]
6. TAKE ACTION {DoNothing, Alert, PauseUserCommand,
     TerminateUserCommand, KillUserSession}
```

All risk exposure, probability and impact rules will be stored in the DIDS-DW Database and used by the Intrusion Response Manager (IRM), as explained formerly.

Table 1 Misuse detection features in the DIDS for DWs

Feature name	Description
#ConsecFailedLoginAttempts	The number of consecutive failed database login attempts by a UserID or from an IPAddress (accumulated or in a given timespan)
#SimultSQLSessions	The number of active simultaneous database connections on behalf of a UserID or IPAddress
#SimultSQLCommands	The number of simultaneous SQL commands executing on behalf of a UserID or IPAddress, per session
#RowsInsert	The number of rows to insert in a given table by a UserID or IPAddress
#RowsUpdate	The number of rows to update in a given table by a UserID or IPAddress
#RowsDelete	The number of rows to be deleted in a given table by a command issued by a UserID or from a given IPAddress
#UnauthorAccessAttempts	The number of consecutive user requests to execute unauthorized actions (e.g., an end user that requests to modify data when the database is read-only or requests to query data to which he does not have access privileges) from a UserID or IPAddress

4.2 Misuse Detection

Considering the assumptions discussed in the prior sections, we will now identify the information required for misuse detection in DWs. Taking the first step of an intrusion as the attempt to login to its database, there are mainly two possibilities: (1) the attacker does not possess valid login credentials and tries multiple attempts to guess a valid login, and (2) the attacker has a valid login credential. In the first case, she/he may try simultaneously opening several database login sessions in order to rapidly try many possibilities, or she/he can open one database login session and sequentially attempt diverse login possibilities. These attempts may also be spaced in time, to try avoiding being detected. This means that we must be able to isolate these attacks, in order to be effective on this type of misuse detection. Thus, the following information must be captured and analyzed: in each database login attempt, the DIDS must store the entered login credentials (e.g., username and password), the IP address of the machine from which the user is trying to login and the session ID open by the communication established with the database server, and the date/time of the login attempt. With this information and based on general features in DW database management, we consider the features shown in Table 1 as interesting for misuse detection purposes.

A maximum value for each misuse feature should be defined by the DW Security Administrator for each UserID and IPAddress. To detect misuse, every time a user command is requested to be executed by the DBMS, the Misuse Detector will verify these values for each feature to check if the user action exceeds them. If it does, then an alert is generated that is analyzed and ranked by the IRM given the predefined probability and impact rules, as explained in the previous subsection.

4.3 Anomaly Detection

Our anomaly detection approach is based on building individual user profiles which are incrementally updated. To build these profiles, we consider features coming from two sources: the command itself and the result of its execution. If no alert is generated after the command has been analyzed a priori to its execution, then it is processed and its result will be analyzed before it is returned to the user. Before executing each user command, we extract the UserID and IPAddress that requested its execution, the SQL command text, tables and columns affected by it, restriction columns and literal values used for filtering data (from the SQL WHERE clause), the used functions (sum, avg, max, min, count), the command type (select, insert, update, delete, create, alter, etc.), the date and time when it was submitted to the DBMS, and the type of user (Database Administrator—DBA, ETL interface, End User). For the result of the command's execution, we consider the CPU time spent in computing the result and the result's size, the returned number of rows, and the returned number of columns. With this information and based on the specificity of the data warehousing assumptions presented in the previous sections, we consider the features shown in Table 2 as interesting for anomaly detection for each requested user action.

For each user, for each feature, a probabilistic distribution should be built based on the values for each feature obtained during a learning phase defined by the DW Security Administrator. After that, they are incrementally updated with the database log. For checking anomalies in each user action we propose the execution of two probabilistic hypothesis tests for each feature, both with a 95 % confidence level:

1. One to verify if the values for each anomaly feature for the individual user command are outliers of their respective predefined distribution. For each feature where there is a probable outlier, an alert is generated by that feature.
2. Another to verify if the sample of values for each anomaly feature for all the user commands issued in the current active session (to which the user command being analyzed belongs) also belongs to their respective predefined distribution. For each feature where the sample of session values probably does not belong to its distribution, an alert is generated by that feature.

Similarly to misuse detection, if an alert is generated by the Anomaly Detector, it is dealt with by the IRM, given the defined user/command's probability and impact.

Table 2 Anomaly detection features in the DIDS for DWs

Anomaly features per command	
Feature name	Description
CPUTime	CPU time spent by the DBMS to process the command
ResponseSize	Size (in bytes) of the result of the command's execution
#ResponseLines	Number of response lines in the result of the command's execution
#ResponseColumns	Number of columns in the result of the command's execution
#ProcessedRows	Number of accessed rows during the command's execution
#ProcessedColumns	Number of processed rows during the command's execution
CommandLength	Number of characters in the command
#GroupBy	Number of GROUPBY columns in the command
#Union	Number of UNION clauses in the command
#Sum, #Max, #Min, #Avg, #Count	Number of SUM, MAX, MIN, AVG, and COUNT functions in the command
#And, #Or	Number of AND and OR operators in the command's WHERE clause(s)
#LiteralValues	Number appearances of literal values in the command's WHERE clause(s)
Anomaly features per session	
Feature Name	Description
#Select, #Insert, #Delete, #Update, #Create, #Alter, #Drop	Number of executed SELECT, INSERT, DELETE, UPDATE, CREATE, ALTER, and DROP commands, per session
#Insert-Select, #Create-Select	Number of executed INSERT and CREATE commands that included SELECT commands, per session
#GroupBy	Number of GROUPBY columns in all SELECT statements, per session
#Union	Number of UNION clauses used in all SELECT statements, per session
#Sum, #Max, #Min, #Avg, #Count	Number of appearances of SUM, MAX, MIN, AVG, and COUNT functions in all commands, per session
TimeBetwCommands	Time period (seconds) between execution of commands, per session
#SimultCommands	Number of commands simultaneously executing, per session

5 Experimental Evaluation

We used the TPC-H benchmark [17] to build a 1GB DW using Oracle 11g DBMS on a Pentium 2.8 GHz machine with a 500 GB SATA hard disk and 2 GB SDRAM (with 512 MB dedicated to the database server), running Windows 2003 Server for executing a preliminary test in order to measure the efficiency of the proposed approach. We defined a scenario with ten web connections to the DW (simulating a cloud DW environment) in which there were seven "true" DW users (non-intruders) and three "intruders." For each "true" DW user workload, a random set of randomly chosen TPC-H queries was selected, i.e., each user had different queries to execute, as well as a distinct number of queries. In each workload, several queries were randomly picked for randomly changing their fixed parameters (namely, in their

Table 3 Composition of each "true" DW end user workload

True user	Query workload
1	Oq1, Mq3, Oq6, Mq8, Oq11, Mq12, Mq15, Oq16, Mq19, Oq21 + 2 random queries
2	Oq1, Mq2, Oq4, Mq6, Oq8, Oq10, Oq13, Mq15, Oq17, Mq18, Oq20, Mq22 + 3 random q
3	Oq2, Mq4, Mq7, Mq9, Oq12, Oq14, Mq16, Oq23 + 1 random query
4	Mq5, Oq7, Oq9, Mq14, Mq23 + 5 random queries
5	Mq1, Oq3, Oq5, Mq10, Mq11, Mq13, Mq17, Oq18, Oq19, Mq20, Mq21, Oq22 + 3 random q.
6	Oq2, Mq4, Oq7, Mq9, Oq12, Oq15, Oq18, Mq19, Mq21, Mq23 + 2 random queries
7	Oq3, Mq5, Mq8, Oq10, Mq12, Oq15, Oq17, Mq18, Oq20 + 5 random queries

WHERE clause) to obtain a larger scope of user actions from the benchmark queries (modified parameter queries). Each workload also included a random number of *random* queries (built by randomly picking a set of tables, columns, functions to execute, grouping and sorting, and literal restrictions for columns included in the WHERE clauses). The proportion of TPC-H and random queries used in each workload was, respectively, 80 and 20 % (on average). The first represent typical analytical reporting behavior in DWs, while ad hoc queries were simulated by the random queries, in smaller number. The workload for each "true" user is shown in Table 3, where *OqXX* represents the original TPC-H query number *XX* and *MqXX* stands for TPC-H query *XX* with modified parameters, as explained previously, with *XX* = 1..23.

To build the individual mean and standard deviation models for each feature of each "true" user, we executed each user's workload 25 consecutive times and extract its values. To build each "intruder" workload, we generated 300 random intrusion queries from several types: SQL injection tautologies; login/password guessing; inserting, changing, or deleting a random number of rows; selecting a random amount of columns from a random number of tables, with and without range value restrictions; selecting a random amount of columns with a random amount of functions (MAX, SUM, etc.) from a random number of tables, with and without a random number of grouping columns and with and without range value restrictions; selecting a random amount of columns from a random number of tables with a random amount of grouping columns, with and without range value restrictions; SQL union queries with a random amount of columns and a random amount of tables; query flooding; and unauthorized actions (create, drop, etc.). These intrusion queries were executed in random order in the experiments, representing a wide variety of attacks.

The TPC-H benchmark has approximately 7 years of business data. We consider the data from the most recent year to have high impact due to intrusion actions, the data from the two previous years as high impact, the data from two years before as low impact, and the remaining as having very low impact. All probability features were initially set to *High*. Table 4 shows the intrusion detection results from "true" user and intruder actions (*True Positive (TP)*: an alert referring to a true intrusion, *False-Positive (FP)*: an alert that results in a false alarm, *True Negative (TN)*: the absence of an alert given a true "normal" user action, and *False Negative (FN)*: the absence of an alert given an undetected true intrusion; *TP* and *FP rates*

Table 4 Experimental results for the generated alerts (absolute values)

# True user actions	# Intruder actions	#TP	TP rate	#FP	FP rate	#TN	#FN
997	300	236	78.6 %	95	9.5 %	902	64

Table 5 Number of generated alerts per risk exposure measure

Total number of alerts	Very low	Low	High	Very high	Critical
331	61	72	83	68	47

(TP/(TP + FN)) and FP/(FP + TN), respectively). Table 5 shows the number of generated alerts for each risk exposure measure. To compute the approach's efficiency, *accuracy* is defined as ((TP + TN)/(TP + FP + TN + FN)) and *precision* as (TP/(TP + FP)).

As seen in Table 4, the TP rate is considerably high (78.6 %), while the FP rate is relatively low (9.5 %), with an absolute number of 95 false alarms for a total number of 331 generated alerts. It can be seen that the absolute number of false-negatives is relatively low (64 in a total of 997 non-intrusions). The computed approach's precision is considerable (71.3 %), and its accuracy is considerably high (87.7 %). Observing Table 5, it can be seen that the most relevant alerts (very high and critical) represent approximately one third of all alerts, which should be the ones first deserving full attention on behalf of the security staff, instead of wasting potentially precious time checking the remaining alerts with the remaining two thirds. Finally, in what concerns the impact on database performance, we measured an average overhead of 24 % on each user's workload response time due to running the DIDS detection algorithms.

6 Conclusions and Future Work

We have referred and analyzed the existing DIDS and pointed out their issues from a data warehousing perspective. We have proposed an approach for building a DIDS that works transparently between user applications and the database server as an extension of the DBMS itself. The solution is specifically tailored for detecting and managing intrusions in DWs, given the features and assumptions of typical data warehousing environments. Risk exposure assessment is used for ranking and prioritizing the generated intrusion alerts, enabling to rapidly respond to alerts which may cause greater impact in the enterprise. The proposed features for anomaly and misuse detection are those which we have concluded to be the most relevant; however, new features can be added by security staff for increasing the DIDS overall flexibility and scope. The results from the experimental evaluation show that our proposal is efficient and enables dealing more rapidly with the most important alerts. As future work, we will test our approach in real-world DWs and focus on how to assess and automatically tune the importance of the ID features, given each of their individual contribution to the overall intrusion detection processes, as well as how to refine the efficiency, accuracy, and performance of the misuse and anomaly detection procedures.

References

1. Bockermann C, Apel M, Meier M (2009) Learning sql for database intrusion detection using context-sensitive modelling. In: Detection of Intrusions and Malware, and Vulnerability Assessment (pp. 196–205). Springer Berlin Heidelberg
2. Fonseca J, Vieira M, Madeira H (2008, March). Online detection of malicious data access using DBMS auditing. In: Proceedings of the 2008 ACM symposium on Applied computing (pp. 1013–1020). ACM
3. Hu Y, Panda B (2004, March). A data mining approach for database intrusion detection. In: Proceedings of the 2004 ACM symposium on Applied computing (pp. 711–716). ACM
4. Jin X, Osborn SL (2007) Architecture for data collection in database intrusion detection systems. In: Secure data management (pp. 96–107). Springer Berlin Heidelberg
5. Kamra A, Terzi E, Bertino E (2008) Detecting anomalous access patterns in relational databases. Springer VLDB J 17(5):1063–1077
6. Kimball R, Ross M (2002) The data warehouse toolkit, 2nd edn. Wiley, New York
7. Kundu A, Sural S, Majumdar AK (2010) Database intrusion detection using sequence alignment. Int J Inform Secur (9), 2010
8. Lee SY, Low WL, Wong PY (2002) Learning fingerprints for a database intrusion detection system. In: Computer Security—ESORICS 2002 (pp. 264–279). Springer Berlin Heidelberg
9. Lee VC, Stankovic JA, Son SH (2000) Intrusion detection in real-time database systems via time signatures. In: Real-Time Technology and Applications Symposium, 2000. RTAS 2000. Proceedings. Sixth IEEE (pp. 124–133). IEEE
10. Mathew S, Petropoulos M, Ngo HQ, Upadhyaya S (2010, January). A data-centric approach to insider attack detection in database systems. In: Recent Advances in Intrusion Detection (pp. 382–401). Springer Berlin Heidelberg
11. Newman AC (2011) Intrusion detection and security auditing in Oracle. Application Security Inc. White paper
12. Pietraszek T (2004, January). Using adaptive alert classification to reduce false positives in intrusion detection. In Recent Advances in Intrusion Detection (pp. 102–124). Springer Berlin Heidelberg
13. Pietraszek T, Tanner A (2005) Data mining and machine learning – towards reducing false positives in intrusion detection. Inform Secur Tech Rep 10(3):169–183
14. Rao UP, Sahani GJ, Patel DR (2010) Clustering based machine learning approach for detecting intrusions in RBAC enabled databases. IJCNS 2(6)
15. Spalka A, Lehnhardt J (2005) A comprehensive approach to anomaly detection in relational databases. In: Data and Applications Security XIX (pp. 207–221). Springer Berlin Heidelberg
16. Srivastava A, Sural S, Majumdar AK (2006) Database intrusion detection using weighted sequence mining. J Computer 1(4)
17. Transaction Processing Council. Decision support benchmark TPC-H, www.tpc.org/tpch
18. Treinen JJ, Thurimella R (2006, January). A framework for the application of association rule mining in large intrusion detection infrastructures. In: Recent Advances in Intrusion Detection (pp. 1–18). Springer Berlin Heidelberg
19. Yu Z, Tsai JP, Weigert T (2007) An automatically tuning intrusion detection system. IEEE T Syst Man Cy 37(2)

Crowdsourcing Large-Scale Ecological Monitoring: Identifying Design Principles to Motivate Contributors

Christoph Schneider and Frederik von Briel

Abstract Addressing the impact of humans on the environment is arguably one of the biggest challenges society faces, and large-scale ecological monitoring is needed to reliably assess the impact and establish relevant policies. However, such large-scale monitoring is often infeasible, primarily owing to resource limitations. Recently, organizations have started to use information technologies to enable public participation in such efforts. One major problem is how to motivate people to contribute and, more importantly, to encourage sustained participation. In this conceptual paper, we integrate research from crowdsourcing, human–computer interaction, and motivational affordances to propose design principles enhancing the intrinsic motivation of contributors to large-scale ecological monitoring projects. Specifically, drawing on research on crowdsourcing and motivational affordances, we argue that instantiating design principles addressing people's needs for autonomy, competence, and relatedness can increase participants' motivation and present recommendations for designers of systems supporting such projects.

1 Introduction

One of the biggest issues society faces is the impact of humans on our environment [23]. Particularly, researchers have noted that our limited ability to assess and predict the impacts of human activity on the ecosystems we live in is one of the main factors hindering conservation efforts [4, 41]. In particular, a challenge is being able to separate the impacts arising from natural changes from impacts created by human action [35];

C. Schneider (✉)
City University of Hong Kong, Kowloon Hong Kong SAR

F. von Briel
Department of Information Systems, City University of Hong Kong, Kowloon, SAR, Hong Kong
e-mail: christoph.schneider@cityu.edu.hk; frederik@von-briel.net

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*, DOI 10.1007/978-1-4614-7540-8_39,

predicting such impacts necessitates repeatedly measuring the ecosystems' responses to disturbances [10, 12], so as to create a continuous feedback loop, helping to establish, prioritize, and adapt policies and conservation efforts [3, 36]. For example, researchers are interested in recording and analyzing changes to fauna and flora, temperatures, or pollutant concentrations. In order to assess long-term trends on a large geographical scale, large-scale ecological monitoring is needed [16, 39].

Yet, as such large-scale ecological monitoring can quickly become very costly, it is conducted less frequently than desired [3], and examples show that even well-funded programs often fail [4]. Recently, researchers have called for harnessing the power of the crowds for large-scale ecological monitoring. Termed "citizen science," this form of crowdsourcing uses ordinary citizens for collecting field data [34] that can be analyzed by researchers. Advances in mobile technologies have given further rise to the unique and promising opportunity of collecting field data of various nature (including sensor data, geospatial data) [25]. For example, the "Clean Marine" smartphone app encourages users to report marine debris; once reported, a related photo (paired with GPS coordinates) is transferred to regulatory authorities. Similarly, the "Project Noah" app allows smartphone users to participate in various field missions, so as to contribute to research projects. Thus, there clearly is much potential in using mobile apps for sustainability-related purposes; yet, it remains unclear how to best design systems shaping beliefs and influencing actions about environmental issues, how issues related to the context influence people's intention to use sustainability-related information systems, or which approaches or design methodologies help build the most effective systems [23]. Especially in the context of crowdsourcing ecological monitoring, a major question is how to motivate people to contribute [26, 29] and, in particular, how to sustain people's motivation to contribute. Therefore, the objective of this paper is to identify and conceptually link design principles to motivational factors that are relevant in this particular context.

By drawing on research on crowdsourcing and motivational affordances, we propose design principles that can serve to enhance the motivation of contributors in large-scale ecological monitoring projects. Therefore, we first provide an overview of related research, followed by a discussion of how to instantiate specific design principles. Finally we will provide a conclusion and suggest opportunities for future research.

2 Related Research

In this section, we provide a brief overview of crowdsourcing, followed by a discussion of participants' motivation in such projects and a description of the concept of motivational affordance.

2.1 Crowdsourcing

Business organizations frequently outsource tasks, for example, to reduce costs or make use of the best talent available [19]. Recently, advances in information technology have given rise to crowdsourcing or outsourcing tasks to anyone who is willing to work on the task (i.e., the crowds [14]). For companies, such crowdsourcing can create an ad hoc labor network that can be accessed as needed [15]. Similar to drawing on computing resources from the cloud, using crowdsourcing, companies can easily access human computing resources [47].

Drawing on a large number of people with diverse backgrounds and a broad range of expertise allows for utilizing collective intelligence, i.e., the notion that large crowds with diverse perspectives can outperform individual experts [24, 38]. This use of collective intelligence has been shown to enable the creation of outputs of very high quality, as shown, for example, in the context of open source software or the online encyclopedia Wikipedia [20], where the quality of the output is based on the contributions from the crowd (e.g., [31]). In addition to collective intelligence (or wisdom of the crowds [38]), other mechanisms to harness the power of the crowds include crowd creation, the purpose of which is creating products or render services (e.g., Amazon's Mechanical Turk or iStockphoto), crowd voting (e.g., prediction markets), and crowdfunding, the purpose of which is raising funds (e.g., micro-lending) [13, 15].

As a result, crowdsourcing has become very popular for micro-tasks, i.e., tasks that do not require specific expertise and that can be performed within a relatively short time period (typically, within several minutes) [20]. This makes crowdsourcing especially useful for citizen science applications [43] and, in particular, for large-scale ecological monitoring, as the associated tasks (or observations) are largely independent and can be performed with relatively little effort. For example, ecological monitoring could include recording data about bird populations, the flowering of trees, or changes in temperature throughout a day.

2.2 Motivating Participation

One of the key success factors of crowdsourcing initiatives is the motivation of participants. Typically, researchers distinguish between intrinsic motivation, which is engagement without apparent external incentives, and extrinsic motivation, which is characterized by external incentives, such as monetary rewards (e.g., [7]). Examples of extrinsic motivators include extrinsic rewards and enhanced reputation or image, whereas sense of self-worth, learning, or enjoyment in helping others are considered intrinsic motivators [37]. In addition, social motives such as social identity, reciprocity, or community advancement can help motivating participation [37].

In the context of crowdsourcing, most research known to us focuses on extrinsic motivation (e.g., [21]); as notable exceptions, two recent studies examined the role of intrinsic motivation on contributors' participation behavior. Specifically, Kaufmann et al. [18] developed a model of worker motivation in crowdsourcing; based on motivation theory, work motivation theory, and open source systems development literature, the authors classified enjoyment-based and community-based motivation as intrinsic motivation and classified immediate payoffs, delayed payoffs, and social motivation as extrinsic motivation. Using a survey of workers on

Amazon Mechanical Turk (a popular micro-task market for crowdsourcing projects), the authors found intrinsic motivations to be stronger than extrinsic motivations; specifically, the factors task autonomy, skill variety and task identity played an important role in determining workers' motivation. The only extrinsic motivation that appeared to influence motivation to a similar extent was human capital development, a delayed payoff that includes learning and skill development.

Zheng et al. [46] investigated the factors motivating participants in crowdsourcing contests on taskcn.com, a Chinese crowdsourcing platform. Similar to Kaufmann et al. [18], their results indicate that intrinsic motivation is more important than extrinsic motivation, and especially task autonomy and variety are strong motivations.

Relatedly, research in the area of citizen science found that people tend to join projects due to their inherent interest in the subject matter, for fun or for altruistic reasons [27]. However, especially for initiatives of longer duration, such as large-scale ecological monitoring, not only initial participation is important but also sustained participation [37, 40]. Research on open-source software development or contribution to Wikipedia shows that intrinsic motivators can help create sustainable digital ecosystems (e.g., [40]). This is especially opportune, as resource limitations typically prohibit remunerating participants of large-scale ecological monitoring efforts. Thus, the design of systems supporting such efforts should focus not only on factors creating initial interest but also on factors encouraging people to *continue* to participate. In the context of virtual communities, Sun et al. [37] stressed the importance of intrinsic motivation on continuance intention. We believe that research on motivational affordances can provide guidance in identifying factors that can help to sustain participants' intrinsic motivation.

2.3 Motivational Affordances

In recent years, researchers have called for applying principles of positive psychology to achieve desired effects from the use of information systems [2]: by designing systems that satisfy the users' basic psychological, social, cognitive, and emotional needs, positive states and outcomes can be created [44]. Following this premise, Zhang [44, 45] proposed ten design principles to enhance a system's "motivational affordance" and called on researchers to develop guidelines for specific usage contexts. We believe that these design principles can be very useful in the context of mobile apps for large-scale ecological monitoring, where participation is entirely voluntary, external rewards are typically infeasible, and sustained participation is highly desired.

3 Instantiating Design Principles

In the next sections, we draw on Zhang's [44, 45] design principles and propose how they can be successfully instantiated, with a specific focus on the promises and potential of mobile apps. The mobile context has a number of distinct

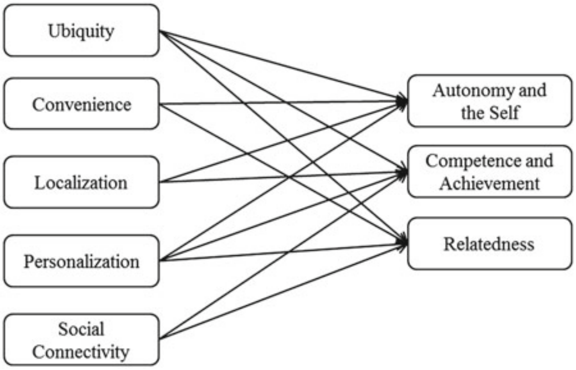


Fig. 1 Mobile phone characteristics addressing human needs

characteristics, specifically *ubiquity* (accessibility from almost any location), *convenience* (reduced constraints in terms of time and space), *localization* (ability to provide information based on the user's location), and *personalization* (ability to target specific individual users) [5]. Together, these characteristics further enable capabilities generally associated with Web 2.0 [30]; for example, permanent reachability increases people's ability to connect with others, facilitating *social connectivity*. Given the specific, unique characteristics of our context, especially the design principles addressing needs related to *autonomy and the self*, *competence and achievement*, and *relatedness,* appear to be most useful for enhancing the intrinsic motivation of contributors (see Fig. 1). Arguably, there are other unique characteristics of the mobile context, such as platform variation, or different needs for requirements specification (e.g., [1]); however, these are not directly related to user motivation and are thus beyond the scope of this paper.

3.1 Design Principles Focusing on Autonomy and the Self

In their seminal work on self-determination theory, Deci and Ryan [7] argued that people have the basic need to experience choice, rather than external pressure; as people desire choice, they can derive pleasure from challenges, leading to greater engagement [32]. Following this logic, Zhang [44] proposed the design principle *support autonomy*, arguing that flexibility and the ability to self-direct tasks and outcomes can support autonomy-related human needs. Mobile apps are especially well suited to support such needs, as they naturally allow people to contribute to tasks whenever it suits their individual circumstances. Specifically, mobile phones have the capability to determine time, location, or task, so that ubiquity, convenience, and localization help to harness "kairos," or the most opportune moment to contribute to a task [9].

Relatedly, motivational theories suggest that people desire to define and create their own self, as well as relate the self to the social context [32]; consequently, Zhang [44] suggests that information and communication technologies should seek to *promote creation and representation of self-identity*. Especially in the context of crowdsourced large-scale ecological monitoring projects, this principle appears to be promising, as participants, though they are not directly interacting, are embedded within a group of like-minded contributors. A successful instantiation of this design principle can be seen in the online encyclopedia Wikipedia, where contributors can create highly personalized user pages; these user pages often reflect a user's distinct personality, featuring information about the contributor, favorite projects, pictures, and so on. For monitoring projects, this provides a good example of implementing this particular design principle. We propose that providing users with means to create an own identity (such as going beyond user names and providing the ability to create personalized user pages) can promote the creation and presentation of self-identity, thereby increasing motivational affordance.

3.2 Design Principles Focusing on Competence and Achievement

Flow theory, originally proposed by Csikszentmihalyi [6], suggests that people perform best when an optimal balance between skills and challenges is provided. In other words, an excessively challenging task creates anxiety, whereas an overly simple task induces a state of boredom. Consequently, Zhang [44] proposes the principle *design for optimal challenge*, implying that people are motivated by new and different challenges. Research in the area of goal setting and group collaboration has shown that designing for optimal challenge can effectively enhance performance [17]. Yet, the challenges have to be interesting. A case in point is the once-successful social networking game Farmville: whereas Farmville once boasted 80 million monthly active users, this number has dwindled, partly due to the nature of the game, where the tasks were typically just more of the same. Yet, especially a mobile setting can be used effectively to provide interesting challenges; specifically, ubiquity, localization, and personalization can help to offer different, interesting tasks and provide different challenges for each individual user. One way to provide such optimal challenge is to offer the ability to participate in various different projects, as is the case with the Project Noah community, where users can select tasks from a wide array of projects [28].

Closely associated with designing for optimal challenge is Zhang's [44] principle *provide timely and positive feedback*. This principle is based on people's inherent need to excel in competition with a task, the self, or others [11]; in order to satisfy this need, people need to be able to evaluate their performance [22]. In a group collaboration context, researchers have shown that providing real-time performance feedback can significantly increase performance [17]. One key criterion is that the performance feedback is meaningful. Research in gamification [8] suggests that leaderboards and scorecards can effectively provide such feedback on a user's performance (typically as compared to others). Other types of feedback, using aspects of social connectivity and personalization, could further enhance motivation.

Table 1 Exemplary implementations of design principles

Design principle	Exemplary implementation	Mobile characteristics
Support autonomy	Mobile apps, "kairos"	Ubiquity, convenience, localization, personalization
Promote creation and representation of self-identity	Customizable user profiles and pages	Personalization, social connectivity
Design for optimal challenge	Different tasks to choose	Ubiquity, localization, personalization, social connectivity
Provide timely and positive feedback	Timely feedback	Ubiquity, personalization
Represent social bonds Facilitate human–human interaction	Display of participant's contribution within overall project	Social connectivity

Given the individual, yet collective nature of the tasks in large-scale ecological monitoring projects, feedback could be provided to show how an individual user's contributions fit within the overall project. Relatedly, users can be provided with not only meaningful feedback but also meaningful rewards. Proponents of gamification at times propose providing simple (and cheap) rewards in the form of badges or stars, an approach sometimes criticized as "pointsification," as meaningless rewards over time tend to lose their effectiveness in motivating people [33]. In contrast, Project Noah provides an interesting approach to provide more meaningful rewards: users can upload images of organisms unknown to them and can receive help from the community in identifying the species; this can serve as a reward for contributing to the aims of the project [28].

3.3 Design Principles Focusing on Relatedness

As stated by Zhang [44], people have the desire to belong and to form emotional bonds. Hence, providing information systems with the capabilities to form and display social bonds can help to increase an individual's motivation. Zhang thus proposes that information systems should *facilitate human–human interaction* and *represent human social bonds*. Ubiquity, convenience, and social connectivity provided by mobile devices can be used to instantiate these design principles. Specifically, paired with the provision of individualized user pages, apps supporting crowdsourced ecological monitoring should provide means for users to interact, including commenting functions, discussion forums, and the like. By creating such community-oriented functionality, managers of such projects can enhance sustained participation, as contributors are likely to come back and interact with like-minded others. Further, especially the ubiquity and convenience offered by mobile devices can be used to further enhance motivation, as it allows people to connect in real time, sharing observations, questions, or concerns. This, in turn, can strengthen the community, leading to greater buy-in. Table 1 summarizes the design principles, exemplary implementations, and characteristics of the mobile context supporting human needs.

4 Conclusion and Future Research

One of the big challenges society faces is the protection of our ecosystems. In order to formulate relevant and effective policies, an assessment of the impacts of human activity on the environment is needed. However, large-scale ecological monitoring projects to assess such impacts are often infeasible due to resource constraints [3]. Recently, advances in mobile technology have enabled the use of crowdsourcing for such monitoring projects, such that individuals can provide inputs using their mobile phones. Given that funding for such projects is typically limited, a challenge in this context that remains is the motivation of individual contributors.

In this conceptual paper, we proposed that instantiating certain design principles can help to enhance the motivational affordance of such systems, potentially leading to greater sustained participation. We built the bridge between motivating factors in crowdsourcing and related design principles, thereby laying the foundation for future research on crowd motivation in the context of large-scale ecological monitoring from a design science perspective. Given the conceptual nature of this paper, we call on researchers to implement and empirically test these design principles. Further, we call on researchers to theorize and test mechanisms to address other challenges related to crowdsourcing. For example, as with any crowdsourced project, the quality and reliability of data can be a challenge [42]. It stands to reason that increasing the intrinsic motivation of contributors is also likely to increase the data quality; however, this has to be tested and therefore offers additional potential for future research. An additional fruitful area of research is related to the effects of participation in ecological monitoring projects on users' beliefs and attitudes; specifically, in addition to providing useful input into ecological monitoring projects, the use of mobile devices could also help to increase interest, thereby promoting greater ecological awareness.

From a practitioner's perspective, the application of these design principles in crowdsourcing large-scale ecological monitoring projects seems to be promising, especially since the mobilization of participants through intrinsic motivation can help to overcome resource limitations of such projects. Overall, leveraging mobile technologies and the crowds for large-scale ecological monitoring offers the potential to positively impact ecological monitoring, and we expect that instantiating the design principles presented in this paper supports motivating current and potential participants.

Acknowledgments The work described in this paper was partially supported by a research grant from City University of Hong Kong (Project No. 7002626).

References

1. Andersson B, Henningsson S (2012) Accentuated factors of handheld computing. In: Pooley R, Coady J et al (eds) Information systems development. Springer, New York
2. Avital M, Lyytinen K et al (2006) Design with a positive lens: an affirmative approach to designing information and organizations. Commun Assoc Inform Syst 18(1, Article 25)

3. Burbidge AA (1991) Cost constraints on surveys for nature conservation. In: Margules C, Austin M (eds) Nature conservation: cost effective biological surveys and data analysis (Chapter 1). CSIRO, Victoria
4. Carlson M, Schmiegelow F (2002) Cost-effective sampling design applied to large-scale monitoring of boreal birds. Conserv Ecol 6(2):11
5. Clarke I III (2001) Emerging value propositions for m-commerce. J Bus Strat 18(2):133–148
6. Csikszentmihalyi M (1990) Flow: the psychology of optimal experience. Harper and Row, New York, NY
7. Deci EL, Ryan RM (1985) The general causality orientations scale: self-determination in personality. J Res Pers 19(2):109–134
8. Deterding S, Dixon D, Khaled R, Nacke L (2011) From game design elements to gamefulness: defining "gamification." Proceedings of the 15th international academic MindTrek conference: envisioning future media environments, MindTrek 2011, ACM, New York, NY, USA, pp 9–15
9. Fogg BJ (2003) Persuasive technology: using computers to change what we think and do. Morgan Kaufmann, San Francisco, CA
10. Halbert CL (1993) How adaptive is adaptive management? Implementing adaptive management in Washington State and British Columbia. Rev Fish Sci 1(3):261–283
11. Heckhausen H (1967) The anatomy of achievement motivation. Academic, New York, NY
12. Hinds WT (1984) Towards monitoring of long-term trends in terrestrial ecosystems. Environ Conserv 11(1):11–18
13. Hopkins R (2011) What is crowdsourcing? In: Sloane P (ed) A guide to open innovation and crowdsourcing: advice from leading experts in the field. Kogan Page Publishers, London, pp 15–21
14. Howe J (2006) The rise of crowdsourcing, June 2006. *Wired.com*. http://www.wired.com/wired/archive/14.06/crowds.html. Retrieved 18 Apr 2011
15. Howe J (2009) Crowdsourcing: why the power of the crowd is driving the future of business, 2nd edn. Crown Publishing Group, New York
16. Johnson S (1993) The Earth Summit: the United Nations conference on environment and development (UNCED). Graham & Trotman/Martinus Nijhoff, London
17. Jung JH, Schneider C, Valacich J (2010) Enhancing the motivational affordance of information systems: the effects of real-time performance feedback and goal setting in group collaboration environments. Manage Sci 56(4):724–742
18. Kaufmann N, Schulze T, Veit D (2011) More than fun and money. Worker motivation in crowdsourcing—a study on mechanical turk. Proceedings of the seventeenth Americas conference on information systems, Detroit, MI
19. King WR (2004) Outsourcing and the future of IT. Inf Syst Manage 21(4):83–84
20. Kittur A, Kraut RE (2008) Harnessing the wisdom of crowds in Wikipedia: quality through coordination. Proceedings of the 2008 international ACM conference on supporting group work, San Diego, CA. ACM, New York, NY, pp 37–46
21. Leimeister J, Huber M, Bretschneider U, Krcmar H (2009) Leveraging crowdsourcing: activation-supporting components for IT-based ideas competition. J Manage Inf Syst 26(1):197–224
22. McClelland DC, Atkinson JW, Clark RA, Lowell EL (1953) The achievement motive, Century psychology series. Appleton, East Norwalk, CT
23. Melville NP (2010) Information systems innovation for environmental sustainability. MIS Quart 34(1):1–21
24. Page SE (2007) The difference: how the power of diversity creates better groups, firms, schools, and societies. Princeton University Press, Princeton, NJ
25. Paulos E, Honicky R, Hooker B (2009) Citizen science: enabling participatory urbanism. In: Foth M (ed) Urban informatics: community integration and implementation. IGI Global, Hershey, PA, pp 414–436
26. Prestopnik NR, Crowston K (2011) Gaming for (citizen) science: exploring motivation and data quality in the context of crowdsourced science through the design and evaluation of a social-computational system. 2011 IEEE seventh international conference on e-Science Workshops (eScienceW), pp 28–33

27. Prestopnik NR, Crowston K (2012) Citizen science system assemblages: understanding the technologies that support crowdsourced science. Proceedings of the 2012 iConference, pp 168–176
28. Project Noah | Networked Organisms and Habitats. Project Noah. http://www.projectnoah.org. Retrieved 11 Jul 2012
29. Raddick MJ, Bracey G et al (2009) Citizen science: status and research directions for the coming decade. Astro2010: the Astronomy and Astrophysics Decadal Survey, Paper 46
30. Rana J, Kristiansson J, Hallberg J, Synnes K (2009) An architecture for mobile social networking applications. First international conference on computational intelligence, communication systems and networks, 2009. CICSYN'09, pp 241–246
31. Raymond ES (2001) The cathedral and the bazaar: Musings on Linux and open source by an accidental revolutionary. O'Reilly, Sebastopol, CA
32. Reeve J (2005) Understanding motivation and emotion, 4th edn. Wiley, Hoboken, NJ
33. Robertson M (2010) Can't play, won't play. Hide & Seek, October 6. http://www.hideandseek.net/2010/10/06/cant-play-wont-play/
34. Silvertown J (2009) A new dawn for citizen science. Trends Ecol Evol 24(9):467–471
35. Spellerberg IF (2005) Monitoring ecological change. Cambridge University Press, Cambridge
36. Stork NE, Samways MJ (1995) Inventorying and monitoring. In: Heywood VH, Watson RT (eds) Global biodiversity assessment. Cambridge University Press, Cambridge, pp 453–543
37. Sun Y, Fang Y, Lim KH (2012) Understanding sustained participation in transactional virtual communities. Decis Support Syst 53(1):12–22
38. Surowiecki J (2004) The wisdom of crowds. Doubleday, New York, NY
39. Urquhart NS, Paulsen SG, Larsen DP (1998) Monitoring for policy-relevant regional trends over time. Ecol Appl 8(2):246–257
40. Wagner C, Liu L et al (2009) Creating a successful professional virtual community: a sustainable digital ecosystem for idea sharing. Proceedings of the third IEEE international conference on digital ecosystems and technologies, Istanbul, Turkey, pp 163–167
41. Walters CJ, Holling CS (1990) Large-scale management experiments and learning by doing. Ecology 71(6):2060–2068
42. Wiggins A, Crowston K (2010) Distributed scientific collaboration: research opportunities in citizen science. CSCW, Savannah, GA
43. Wiggins A, Crowston K (2011) From conservation to crowdsourcing: a typology of citizen science. 2011 44th Hawaii international conference on system sciences (HICSS), pp 1–10
44. Zhang P (2008) Toward a positive design theory: principles for designing motivating information and communication technologies. In: Avital M, Boland RJ, Cooperrider DL (eds) Designing information and organizations with a positive lens, advances in appreciative inquiry. JAI, Amsterdam, pp 45–74
45. Zhang P (2008) Motivational affordances: reasons for ICT design and use. Commun ACM 51(11):145–147
46. Zheng H, Li D, Hou W (2011) Task design, motivation, and participation in crowdsourcing contests. Int J Electron Commerce 15(4):57–88
47. Zittrain J (2008) Ubiquitous human computing. Phil Trans R Soc A Math Phys Eng Sci 366(1881):3813–3821

An Analysis of Enterprise Architecture Frameworks from a Coherent Enterprise Description Viewpoint

Vladimir Selcan and Alena Buchalcevova

Abstract Enterprise architecture (EA) is present at every company; however, just few organizations have their architecture formalized and manage it to meet their strategic goals. EA creates opportunity for an effective interaction between business and ITC world. Moreover, EA serves as the blueprint for a company and the process which defines it. This paper formulates problem with current approach describing an enterprise and shows that service orientation is a key to coherent enterprise architecture description. We briefly discuss enterprise architecture cohesion and propose method how to measure the level of cohesion within enterprise architecture layers. Main contribution of this paper consists in a comparison and analysis of the chosen frameworks (ArchiMate, Zachman, TOGAF, and DoDAF) based on cohesion viewpoint. Conclusions made by authors could serve as guidance for particular framework extension with the aim to gain coherent enterprise description.

1 Introduction

Pervasive globalization impacted businesses to be cost-effective and at the same time competitive and innovative. Organizations operate within a rapidly changing and evolving network of business partners, relying on state-of-the-art technology to facilitate their cooperation. Many enterprises have been heavily involved in effort to successfully manage this dependency. In recent years, several scientific papers were published and research done to describe the approaches and methods to solve the problem. These include the ICT[1] assurance, risk management, increased requirements

[1] Information and communication technologies

V. Selcan (✉) • A. Buchalcevova
Department of Information Technology, University of Economics,
W. Churchill Sq. 4, Prague, Czech Republic
e-mail: vladimirselcan@hotmail.com; buchalc@vse.cz

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_40, © Springer Science+Business Media New York 2013

for control and efforts to standardize ICT processes and services [2, 5], enterprise architecture approach, and enterprise architecture frameworks [6, 12, 16, 19]. However, organizations often fail to meet the business requirements [13], and a lack of harmonization and enterprise description coherence is considered as the key reason [3, 15–17]. Achieving alignment between all parts of the enterprise and integrated approach to all aspects of the enterprise is needed. Enterprise effectiveness is not obtained by local optimization, but is realized by well-defined interaction of organization components [6].

Enterprise architecture is an important tool to address whole-enterprise integration. Institute For Enterprise Architecture Developments' definition of enterprise architecture states that it is "a complete expression of the enterprise; a master plan which "acts as a collaboration force" between aspects of business planning such as goals, visions, strategies and governance principles; aspects of business operations such as business terms, organization structures, processes and data; aspects of automation such as information systems and databases; and the enabling technological infrastructure of the business such as computers, operating systems and networks" [11]. However, in practice these architecture domains are not approached in a coherent way. Every domain speaks its own language, draws its own models, and uses its own techniques and tools. Communication and decision making across domains is seriously reduced [14].

We use coherence as a term that defines a logical, orderly, and consistent relation of parts to the whole [8]. We define coherent enterprise as comprehensive enterprise picture that preserves enterprise knowledge, models the enterprise, and aligns technologies to business. As a result enterprise is effective, efficient, and well understood and shows three core outcomes, business alignment, agility, and assurance that through coherency information provide real knowledge. This paper uses coherent enterprise definition as a central part, which is necessary in presenting ideas how to design complex enterprise that must adapt to continuous change.

We attempt to identify key concept of a coherent enterprise. We will explore potential of service orientation to support enterprise cohesion. We put our main effort to business alignment, which provides top-down alignment to segment[2] (business, application, and technology) architectures. Service orientation promises to provide interoperability, agility, and flexibility through reuse of existing services, improved maintainability of service portfolios, and higher customer satisfaction. Service orientation enables organizations to concentrate on improving their business and creates opportunity for cooperation with other business partners. The service concept that plays central role in service orientation is in our view intended to be generic—no solely related to ICT. We consider service as "a unit of functionality that some entity (e.g., a system, organization, or department) makes available to its environment, and which has some value for certain entities in the environment [15]."

In order to establish a common ground for a research focused on a determination how current enterprise architecture frameworks support an enterprise cohesion, we selected the most used enterprise architecture frameworks. According to [1] the most

[2] In this paper, we use term segment architectures and enterprise architecture layers as synonym.

used are TOGAF with 22 %, DoDAF 8 %, and Zachman 7 % of the market share. We decided to extend selected set of frameworks with enterprise modeling language ArchiMate,[3] which was created with the aim to allow a coherent enterprise architecture description. ArchiMate encompasses models that provide baseline for our analysis. Then we defined a method for the frameworks comparison based on their support for the coherent enterprise description. Analysis and frameworks comparison presented in this paper enable understanding of the particular framework's support for the coherent enterprise description.

In the following section we discuss an enterprise architecture framework concept. The Analysis Methodology section defines the way how to compare and analyze frameworks. The section on Enterprise Architecture Frameworks Analysis presents the study of four enterprise architecture frameworks. The Discussion section analyzes framework limitations in service description and proposes solutions. Finally, we conclude the paper and outline future work.

2 Enterprise Architecture Frameworks

Creating the enterprise architecture is a complex process, in which architectural framework plays crucial role. The standard for software intensive systems architecture description ISO/IEC/IEEE 42010 defines an architectural framework as follows "the conventions and common practices for architecture description established within a specific domain or stakeholder community" [10].

This standard also provides additional features of the architectural framework:

- Should establish standards for architectural thinking.
- Serve as a basis for development in the field of architecture.
- Provide resources and tools for architectural description, which serves the stakeholders of described system.

The architectural framework is a communication model for developing the enterprise architecture. It's not architecture itself, but rather a set of models, principles, services, procedures, standards, design concepts, components, and visual representation that help to create a specific aspect of the architecture [18].

The architectural framework should be aligned to enterprise strategic decisions. An architectural framework helps communicate the architecture to all stakeholders. The different views must provide the necessary information to the interested users. In essence, the primary problem of enterprise architecture, which architectural framework is trying to solve is the communication and consistency of a described enterprise [9].

To solve abovementioned problems, it is necessary to create a comprehensive picture of the company with an emphasis on its architecture. As the architecture is

[3] ArchiMate is an open and independent enterprise architecture modeling language that offers a support for describing the construction and operation of business processes, organizational structures, information flows, IT systems, and technical infrastructure [14].

subject to constant change, it is necessary to analyze the impact of the changes in planning future developments. Often, an enterprise architect has to rely on existing methods and techniques from disparate layers, without being able to create the "big picture" that puts these layers together. To have the holistic view, it requires an integrated set of methods and techniques for the specification, analysis, and communication of enterprise architecture that fulfills the needs of the different types of stakeholders involved [15].

ISO/IEC/IEEE 42010 standard [10] shows that precisely defined architectural framework should be in every architectural description. Moreover, standard identifies that the coherent description, which allows explicitly capture relationships between models in an architectural description, significantly improves communication among stakeholders and the consistency of a holistic description of the company.

3 Analysis Methodology

As we pointed before, an enterprise architecture framework should be used for proper description of the current and future states of an enterprise [10]. Therefore, we based our analysis on an existing support of frameworks for a coherent enterprise architecture. In particular, we have analyzed two key areas. At first, we had to understand how service orientation helps to describe the enterprise in a coherent way. Second, to understand how selected enterprise architecture frameworks support enterprise coherence, we had to identify models that support services. Hence, we analyzed selected enterprise architecture frameworks on their service orientation support, which has led to a definition of the following research questions:

Q1: How Service Orientation Supports Cohesion of Enterprise Architecture Models?

To answer the question we define coherent enterprise architecture models description as description with focus on interlayer relations. Coherent enterprise architecture models description must be able to model any global structure within each layer, showing the main elements and their dependencies, in a way which is understandable for stakeholders. Moreover, it must express relevant relations between layers [17].

The coherent enterprise architecture models description provides insight and overview, enables communication among different stakeholders, and guides complex process changes. Using the coherent enterprise description enables architects to avoid communication confusion. After all, enterprise architecture is the mean by which architects communicate with different stakeholders.

Most of the enterprise architecture frameworks are very precise in selecting which elements should be part of the enterprise architecture. However, to keep the enterprise architecture coherent during its lifecycle, the adoption of a certain framework may not be sufficient. The relation between different types of domain, views, or layers of the enterprise architecture must remain clear, and any change must be carried through methodically in all of them [14].

Service orientation plays a major role in a coherent enterprise description [7, 14, 15, 21]. Service concept is applicable at each architecture layer: services in the business layer represent relation to internal and external customers; application layer services represent relation to business layer or the other applications, and the technology layer services form cohesion among applications. The service orientation aspect puts strong emphasis on relations between the different enterprise architecture layers. Integrating interlayer relations is essential for providing the coherent enterprise architecture description.

Q2. How Service-Oriented Models Are Supported in Enterprise Architecture Frameworks?

In order to answer this question, we designed the research method. It is based on weighted sum model (WSM) of multi-criteria decision analysis method [20]. The proposed method is presented in four steps:

1. We defined set of n criteria $C_j = \{C_1, \ldots, C_n\}$ for each m selected enterprise architecture framework alternative A_i (A_1, ArchiMate 2.0; A_2, Zachman 2.1,;A_3, TOGAF 9.0; and A_4, DoDAF 2.02). Criteria identification for comparing alternatives is defined as union of all service-oriented models within each alternative. Models have been identified using literature analysis and for each defined enterprise architecture layer (business, application, and technology).
2. Relative weight w_j denotes to importance of the model from cohesion viewpoint.
   (a) $w_j = 0.1$ model only supports models that provide cohesion, i.e., model shows support for service description; however it does not support cohesion directly.
   (b) $w_j = 0.3$ model supports cohesion within layer.
   (c) $w_j = 0.5$ model supports relevant cohesion between layers.
3. Next we assign a_{ij} <0, 100> subjective criteria scores from expert opinion for each criterion C_j of alternative A_i. Score value represents the level of cohesion the model provides within each architecture layer and it based on our personal experiences. The distinction between relative weight w_j and score value a_{ij} in our method is that w_j evaluates model cohesion from external view while a_{ij} values model cohesion from internal view.
4. Finally, we compute importance of each alternative A_i denoted as $A_i^{\text{WMS-score}}$, for each architecture layer based on defined formula:

$$A_i^{WSM-score} = \sum_{j=1}^{n} w_j a_{ij}, \quad for\, i = 1, 2, 3, \ldots, m.$$

The best alternative is the one that scores maximum total score value.

We selected the most used architecture frameworks that are well known in the relevant communities [1, 4]. The intention of this study is not to report details of each architecture framework but to analyze and compare their similarities and differences in their cohesion support on enterprise level.

4 Enterprise Architecture Frameworks Analysis

This section provides multi-criteria decision analysis of three major enterprise architecture frameworks and enterprise modeling language ArchiMate based on the models that capture services. We had devoted main effort to analyze the models that support services for the particular layer in the enterprise architecture as a means to provide coherent enterprise.

Table 1 shows an overview of models that are provided by enterprise architecture frameworks from a service viewpoint. This table shows models that define set of criteria C_j for each enterprise architecture layer and also define set of alternatives A_i (ArchiMate 2.0, Zachman 2.1, TOGAF 9.0, and DoDAF 2.02). Models aligned in one row do not necessarily have the identical coverage, but from the service viewpoint, these models provide the same description and as a result they have the same cohesion score a_{ij}.

For instance, within technology layer we considered following set of criteria for evaluating alternatives; see Table 2.

Next, we apply our method to compare support for an enterprise coherent description and summary is shown in Table 3. For each criterion within each architectural layer, we have assigned relative weight w_j as defined in step 2 of our research method. Then based on our personal experience (we evaluated internal structures and relations within model with coherent enterprise as a goal), we assigned a_{ij} score value for each alternative framework and for each model within each layer of enterprise architecture. Finally, we computed $A_i^{\text{WMS-score}}$, based on formula defined in step 4.

To illustrate how we assess score for criteria, we use a case often used in the enterprise architecture domain, the ArchiSurance. ArchiSurance is a fictitious company that provides home, travel, and legal aid insurance. It sells its services through a network of intermediaries [7]. Let us evaluate two criteria in the application layer in particular Application Behavior c_{a1} and Application Usage models c_{a3}.

Application Behavior model describes internal behavior of Home & Away Policy application component, data object it uses (Insurance request, Insurance policy, and Customer file) as it realizes Policy creation service. On the other hand, Application Usage model describes how ArchiSurance application components support Handle Claim business process, by describing the services (Scanning service, Customer administration service, Claim administration service, Printing service, and Payment

Table 1 Overview of framework models from a service viewpoint

		A_1	A_2	A_3	A_4
		ArchiMate 2.0	Zachman	TOGAF 9.0	DoDAF 2.02
Business layer	c_{b1}	Introductory	–	–	–
	c_{b2}	Actor cooperation	–	Business use-case	–
	c_{b3}	Business process	–	Business service/ function	–
	c_{b4}	Business process cooperation	–	Business interaction	–
	c_{b5}	Product	–	Business footprint	–
	c_{b6}	Service realization	–	Business service/ information	–
	c_{b7}	Layered	–		–
	c_{b8}	Landscape map	–		–
Application layer	c_{a1}	Application behavior	–	System use-case	Services context
	c_{a2}	Application cooperation	–	Application interaction	Service-services
	c_{a3}	Application Usage	–	Data dissemination	Operational activity to services traceability matrix
	c_{a4}	–	–	–	Services rules model
	c_{a5}	–	–	–	Services resource flow
	c_{a6}	–	–	–	Service technology and skills
	c_{a7}	–	–	–	Service measure
Technology layer	c_{t1}	Infrastructure	–	–	–
	c_{t2}	Infrastructure usage	–	Processing diagram	Systems-services

Table 2 Criteria assessment for technology layer

Index	Criteria	Data source	Formula
c_{t1}	Infrastructure	Expert assessment	Scale of 1–100
c_{t2}	Infrastructure usage	Expert assessment	Scale of 1–100

service) that are used by the business process and can be used by other applications and/or application services.

It is clear that c_{a1} only supports cohesion within application layer, whereas c_{a3} provides cohesion between business and application layer. That is the reason we assigned relative weight for c_{a1} to 0.3 and for c_{a3} to 0.5. Moreover, c_{a3} shows more

Table 3 Comparison method summary table

			A_1	A_2	A_3	A_4
		alts.	ArchiMate 2.0	ZACHMAN 2.1	TOGAF 9.0	DoDAF 2.02
Business layer	c_{b1}	0.5	15	0	0	0
	c_{b2}	0.5	10	0	10	0
	c_{b3}	0.3	10	0	10	0
	c_{b4}	0.5	10	0	10	0
	c_{b5}	0.5	15	0	15	0
	c_{b6}	0.5	10	0	10	0
	c_{b7}	0.5	15	0	0	0
	c_{b8}	0.5	15	0	0	0
	$A_i^{\text{WMS-score}}$		35.5	0	20.5	0
Application layer	c_{a1}	0.3	10	0	10	10
	c_{a2}	0.3	10	0	10	10
	c_{a3}	0.5	20	0	20	20
	c_{a4}	0.3	0	0	0	15
	c_{a5}	0.5	0	0	0	20
	c_{a6}	0.3	0	0	0	15
	c_{a7}	0.1	0	0	0	10
	$A_i^{\text{WMS-score}}$		16	0	16	36
Technology layer	c_{t1}	0.3	10	0	0	0
	c_{t2}	0.5	15	0	15	15
	$A_i^{\text{WMS-score}}$		10.5	0	7.5	7.5

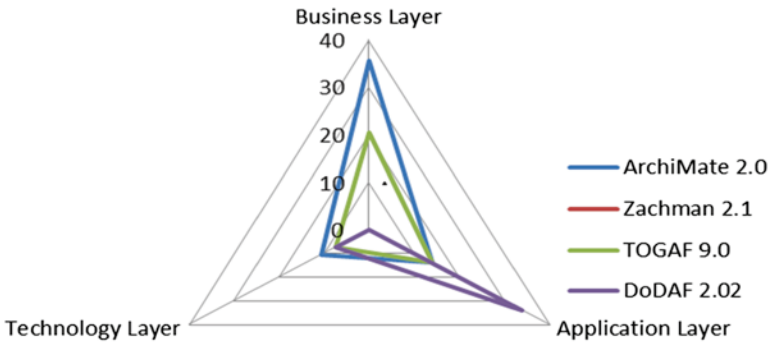


Fig. 1 Results of the frameworks comparison

benefits (relations among application components, relation among application services) as c_{a1}, from internal point of view. That is why we assigned values 10 and 20, respectively (Fig. 1).

Discussions of each framework are made in the following sections where models that support service orientation are analyzed and compared.

5 Discussion

This study compares and analyzes three architectural frameworks and enterprise modeling language ArchiMate 2.0 from the enterprise coherency viewpoint. To enable the analysis, we adjusted on weighted sum model (WSM) of multi-criteria decision analysis method.

As results of our analysis show, the Zachman v2.1 framework does not provide straightforward support for service models. The state of the Zachman framework is likely given by the fact that at the time Zachman framework was promoted, service orientation was not been identified as one of the driving trends in an architectural design. The problem with looking at the enterprise architecture as defined by Zachman framework is that it provides artifacts categorization by contrast, instead of providing an overview of the internal relations among models. This statement is also supported by our analysis results shown in Table 1, with result that Zachman does not provide any support for service description models.

Therefore, in order to provide coherent description in Zachman framework, one must tailor it to support service orientation or put it other way around, put service orientation in the context of the overall Zachman framework. Based on top-down approach and ArchiMate models, we suggest, as starting point for adding service orientation to Zachman, to apply ArchiMate business layer models to Zachman conceptual layer; in the next step, to apply ArchiMate application layer models to Zachman logical layer; and finally, to apply ArchiMate technology layer models to Zachman physical layer.

The analysis shows that DoDAF 2.02 framework provides strong support in IT-related layers of an enterprise architecture and offers models in application and technology layer but abstains at a business layer. However, lack of service-oriented models in the business layer results in poor alignment support between application and business layer of the enterprise architecture. Organization only profits from IT-related benefits of service orientation. Yet again, you can use top-down approach and extend DoDAF business layer with models provided by ArchiMate business layer.

In our analysis TOGAF 9.0 is the only enterprise architecture framework, which supports service orientation at all layers of the enterprise architecture. However, the service orientation is driven from inside out and not vice versa. That means that even though TOGAF business layer covers many ArchiMate business layer service models, some of them are uncovered especially those that ArchiMate provides to align stakeholders with an enterprise.

The analysis results highlight the fact that the ArchiMate enterprise modeling language supports some service-oriented models that are not included in any compared framework. Most of the models are placed in the business layer. We believe that ArchiMate has the potential to be a straightforward solution, which can be used to extend the existing frameworks with models that describe the services. This way extended frameworks would provide improved coherence and better integrated models. In general, enterprise modeling based on service orientation can improve the enterprise architecture as a whole. Moreover, extended enterprise architecture frameworks remove reluctance and failure to deploy those frameworks to production and even with the increased complexity because coherent model provides clear and unduplicated description during enterprise modeling phase.

6 Conclusion and Future Work

Architecture as a concept is not new in the field of information technologies. However, the complexity and dynamic enterprise changes emphasize the search for new ways on how to align whole enterprise to defined strategy. Enterprise architecture, by harmonizing all aspects and layers of the organization, should provide a holistic view of organization. New trends in architecture standardization (new standard ISO 42010: 2011) suggest that a well-defined architecture framework should be the key component of any architecture description. Moreover, a coherent model description should be provided which results in improved integrity of enterprise description.

Importance of enterprise architecture frameworks and the enterprise architecture as a tool for fulfilling an organization strategy is rapidly increasing. Nevertheless, key problem in the enterprise architecture domain seems to be inconsistency in describing different parts of the enterprise.

This paper provides a comparison and analysis of the most used enterprise architecture frameworks, on the level of models that provide service description. The results show that the TOGAF 9.0, Zachman 2.1, and DoDAF 2.02 frameworks poorly or do not at all provide service models at all architecture layers, resulting in poor cohesion of the enterprise. ArchiMate 2.0 provides set of service-oriented models that would help to create a consistent, integrated enterprise view. Thus, adding complementary models in top-down fashion provided by ArchiMate to selected enterprise architecture frameworks will reinforce their coherent enterprise description.

Our methodology accepts uncertainty for criteria scores, and as a future research we will include an investigation of alternative scoring method or finding some ways to make process of criteria scores definition more robust.

As part of our future work, we aim to create lightweight enterprise architecture framework for midsize corporations, in which service orientation will play a main role. We will also investigate how service orientation in enterprise allows organizations smoothly integrate outsourced services bought on the market for fixed prices, which will change the way how business and enterprise IT operates.

Acknowledgments This research has been supported by GACR (Grant Agency, Czech Republic) grant No. P403/10/0303, Enterprise Architecture as Management Principle for SMEs.

References

1. Blowers M (2011) iServer Solutions: using TOGAF9 and industry models to avoid "reinventing the wheel". http://www.orbussoftware.com/downloads/reports-and-publications/ovum-group-white-paper-iserver-solutions. Accessed 28 Nov 2011
2. IT Governance Institute (2006) Board briefing on it governance, 2nd edn. http://www.isaca.org/Content/ContentGroups/ITGI3/Resources1/Board_Briefing_on_IT_Governance/26904_Board_Briefing_final.pdf. Accessed 10 Dec 2011

3. IBM (2007) Building SOA solution using the rational SDP. IBM Corp, Riverton NJ
4. Chen P, Han J, Tang A (2004) A comparative analysis of architecture frameworks. In: Proceedings of the 11th Asia-Pacific software engineering conference, IEEE, Piscataway, NJ, pp 640–647, 30 Nov–3 Dec
5. ISACA (2011) COBIT, control objectives for information and related technology (COBIT), 4.1 edn. IT Governance Institute, Rolling Meadows, IL. http://www.isaca.org/cobit.htm. Accessed 13 Dec 2011
6. DoDAF (2011) DoDAF: DoD Architecture Framework Version 2.02. http://cio-nii.defense.gov/sites/dodaf20/products/DoDAF_v2-02_web.pdf. Accessed 2 Dec 2011
7. Doest H, Iacob M, Lankhorst E et al (2004) Viewpoints functionality and examples. https://doc.telin.nl/dsweb/Get/Document-35434/. Accessed 10 Jan 2012
8. Doucet G, Gøtze J et al (2008) Coherent management: using enterprise architecture for alignment, agility, and assurance. http://siteresources.worldbank.org/EXTEDEVELOPMENT/Resources/JEA_May_2008_Coherency_Management.pdf?resourceurlname=JEA_May_2008_Coherency_Management.pdf. Accessed 3 Jul 2012
9. Shah H, Kourdi ME (2007) Frameworks for enterprise architecture. It Professional 9(5):36–41
10. Emery D, Hilliard R (2009) Every architecture description needs a framework: expressing architecture framework using ISO/IEC 42010*. http://mysite.verizon.net/rfh2/writings/emery-hilliard2009.pdf. Accessed 28 Dec 2011
11. Institute for enterprise architecture developments (2010) Enterprise architecture standards overview. Institute for enterprise architecture developments. http://www.enterprisearchitecture.info/EA_Standards.htm. Accessed 26 Dec 2011
12. The Chief Information Officers Council (1999) FEAF (Federal Enterprise Architecture Framework), version 1.1. The Chief Information Officers Council. http://www.cio.gov/Documents/fedarch1.pdf. Accessed 25 Nov 2011
13. Grembergen W (2005) Measuring and demonstrating the value of IT. IT Governance domain practices and competencies. IT Governance Institute, Rolling Meadows, IL
14. Lankhorst M, Drunen H (2007) Enterprise architecture development and modelling. Via Nova Architectura. http://www.via-nova-architectura.org/files/magazine/Lankhorst.pdf. Accessed 15 Nov 2011
15. Lankhorst M (2005) Enterprise architecture at work—modelling, communication, and analysis. Springer, Berlin
16. McDavid D (2000) The Business-IT gap: a key challenge http://www.almaden.ibm.com/coevolution/pdf/mcdavid.pdf. Accessed 16 Nov 2011
17. The Open Group (2012) ArchiMate® 2.0 Specification. Open Group Standard http://pubs.opengroup.org/architecture/archimate2-doc/. Accessed 20 Mar 2012
18. Sessions R (2007) A comparison of the top four enterprise-architecture methodologies. MSDN Architecture Center. http://msdn.microsoft.com/en-us/library/bb466232.aspx. Accessed 15 Sep 2010
19. The Open Group. TOGAF: The Open Group Architecture Framework. The Open Group. http://www.opengroup.org/architecture/togaf9/downloads.htm. Accessed 11 Dec 2011
20. Triantaphyllou E (2000) Multi-criteria decision making: a comparative study. Kluwer Academic Publishers (now Springer), Dordrecht
21. Velitchkov I (2008) Cohesion of enterprise models. Via Nova Architectura. http://www.via-nova-architectura.org/magazine/magazine/cohesion-of-enterprise-models-3.html. Accessed 14 Feb 2012

Query-Driven Method for Improvement of Data Warehouse Conceptual Model

Darja Solodovnikova, Laila Niedrite, and Aivars Niedritis

Abstract We propose a query-driven method that elicits the information requirements from existing queries on data sources and their usage statistics. Our method presumes that the queries against the source database reflect the analysis needs of users. We use this method to recommend changes to the existing data warehouse schemata. In our method, we take advantage of the schema versioning approach to reflect all changes that occur in the analysed process, and we analyse the activity of users in the source system, rather than changes in physical data structure, to infer the necessary improvements to the data warehouse schema.

1 Introduction

Data warehouses are based on multidimensional models which contain facts (the goal of the analysis), measures (quantitative data), dimensions (qualifying data) and dimension attributes. Dimension attributes form classification hierarchies.

Developed, maintained and used the data warehouses after some time period are subject to changes. New information requirements appear; data structures of data sources vary. Different approaches exist on how to gather the information requirements to construct the most appropriate data warehouse conceptual model. At the same time there are lots of solutions on how to maintain the changes of a data warehouses. Further, a short overview of approaches to the development of conceptual models and data warehouse evolution is given.

The approaches to the development of data warehouse conceptual models are usually classified as supply driven (also known as data driven) and demand driven. More detailed groups can be distinguished according the way how the demands or

D. Solodovnikova (✉) • L. Niedrite • A. Niedritis
Faculty of Computing, University of Latvia, Raina boulv.19, Riga, Latvia
e-mail: Darja.Solodovnikova@lu.lv; Laila.Niedrite@lu.lv; Aivars.Niedritis@lu.lv

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_41,

requirements are elicited—e.g. user driven [1], process driven [2] and goal driven [3]. In data-driven approaches, e.g. [4], a data warehouse model is obtained by transforming models of data sources. More new approaches and methods, e.g. pattern based [5], are proposed to overcome the shortcomings of existing approaches. In the data-driven methods, the constructed conceptual model may not reflect all analysis needs. In the demand-driven approaches, the obtained model depends on knowledge of the interviewed users and the existing data supply is underestimated.

Data warehouse conceptual models tend to evolve, because of changes in data sources and information requirements of users. To reflect this evolution, it is possible to adapt the existing data warehouse schemata and ETL processes, but this solution can cause a loss of history. This is why it is preferable to keep track of the evolution. This can be realized by data warehouse schema versions. According to [6], "schema version is a schema that reflects the business requirements during a given time interval, called its validity, that starts upon schema creation and extends until the next version is created".

According to the design science approach [7], the problem addressed in our research is how to reconcile new information requirements with the existing conceptual model of the data warehouse. The prior approaches to the development of conceptual models of data warehouse and data warehouse schema evolution are reviewed in the related work section. To solve the research problem, we suggest a new type of demand-driven method—query-driven method that elicits the requirements from existing queries on data sources and their usage statistics, so combining the needs of users with existing data supply. We use this method to recommend changes to existing data warehouse schemata.

The method is developed for the case when the data warehouse evolution is implemented by schema versions described with metadata (see Sect. 4). The steps of the proposed method are explained in Sect. 6. The research evaluation step is still in progress and the current state is described in conclusions section.

2 Related Work

Our method for eliciting user information requirements presumes that the queries against the source database reflect the interests and needs of users. Some other works also exploit similar ideas.

Cube design based on queries of users against an existing data warehouse is presented in [8]. The goal of the authors is to improve the query performance by recommending appropriate sub-cubes. Authors of [9] treat queries on operational databases as the requirements. The structure of queries is analysed to obtain elements of a multidimensional model. The method shows that authors presume that the queries implemented in source systems are equal in representing the users' analytical needs, regardless of the real usage of the queries.

A query-driven design framework is presented in [10]. The framework uses prospective user queries for data warehouse schema construction. Users present their queries in natural language form. This approach uses expected queries instead of real queries. Authors of [11] propose an XML data warehouse that is developed based on frequent query patterns. The central issue of the work is the data mining technique to find the query patterns, but not the development of the conceptual model.

Among the above mentioned, the most similar to our method is given in [9], but we also take into account the usage statistics, and the method is tailored to a definite purpose to recommend the necessary changes in existing data warehouse schema.

In the literature there are various solutions for the data warehouse evolution problems that involve data warehouse schema versioning approach. The main idea in [6] is to store augmented schemata together with schema versions to support cross-version querying. In [12] a method to support data and structure versions of dimensions is proposed. The method allows tracking history and comparing data using temporal modes of presentation that is data mapping into the particular structure version. In [13] metadata management solutions in a multiversion data warehouse are proposed.

The above-mentioned papers do not address the problems of the data warehouse adaptation after changes in data sources directly. Several approaches have been proposed for solving these problems, e.g. [14]. Such approaches are based on mappings or transformations that specify how one schema is obtained from the other schema. This specification is used to adapt one schema after changes in the other schema. These approaches do not support evolution history and maintain only the actual schema. Besides, only changes in data structure of the source schema are considered.

In our method, we take advantage of the schema versioning approach to reflect all changes that occur in the analysed process, and we analyse the activity of users in the source system, rather than changes in physical data structure, to infer the necessary improvements to the data warehouse schema.

3 Motivating Example

As an example, let us consider a data warehouse that stores information about students' status changes at the university. The star schema of the status data warehouse is depicted in Fig. 1. For simplification, the example schema contains only the subset of all dimension attributes, which are essential for the illustration of the proposed method. This data warehouse contains a fact table Student_Status_Change, which records the changes in time of statuses of persons, who are students of any study program. The dimension Student_Status stores statuses, which are characterized by attributes: Study_Semester, Funding (state or private), Study_Status (studying, study break, finished studies, etc.), Graduation_Status (graduated with a degree, graduated with a qualification, finished studies without degree or qualification, etc.) and Exam_Status (directed to final exams or passed final exams).

The status data warehouse is updated daily, and the data are obtained from the university data management system. The fragment of the model of the university data management system related to the changes in student statuses is depicted in Fig. 2.

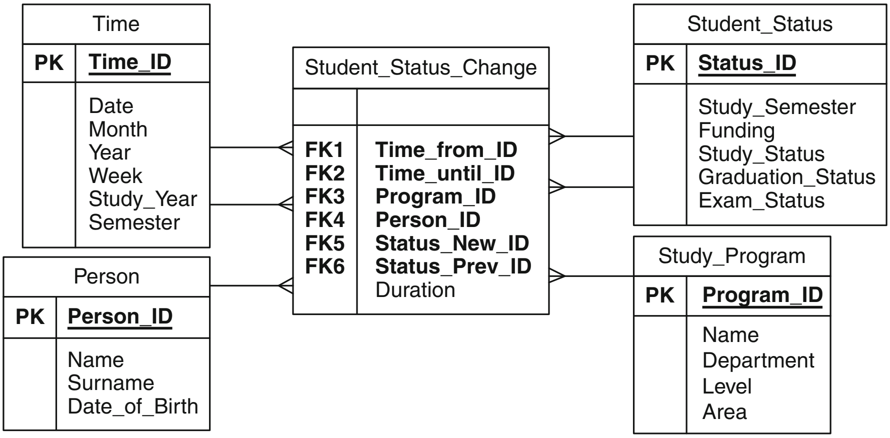


Fig. 1 Example data warehouse

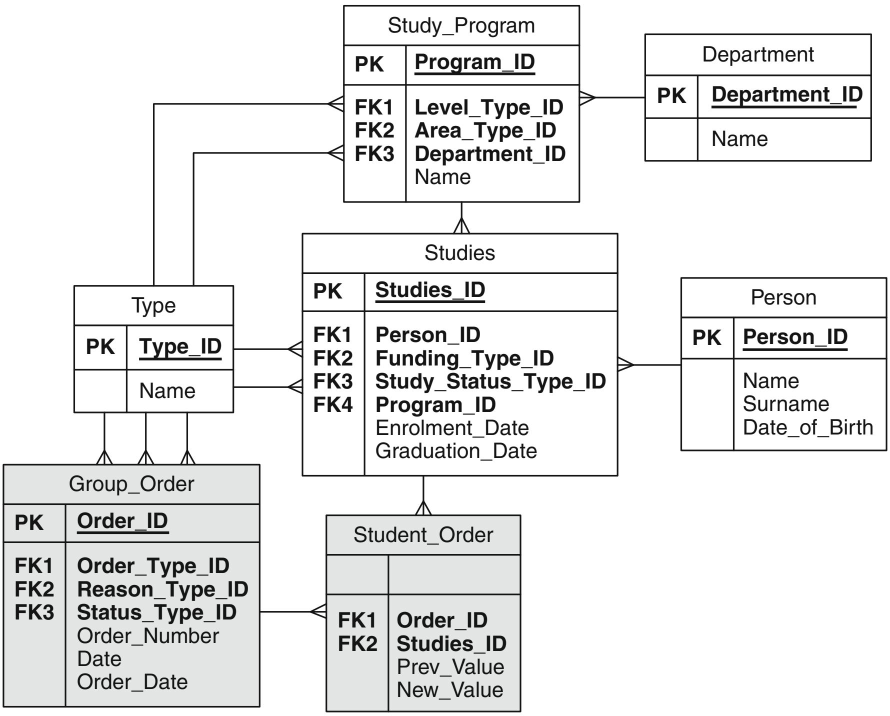


Fig. 2 Data warehouse source database model

The tables used in the extraction, transformation and loading processes of the data warehouse are the following:

- The table Person stores data about all persons related to the university, including students.
- The table Department stores information about departments of the university.

- The table Study_Program stores data about study programs, where students are enrolled.
- The table Studies stores information about enrolment of students into study programs, including funding and student's study status, which includes study semester, graduation information and final examination information.
- The table Types stores names of different types, i.e. levels and areas of study programs, funding options and study statuses of students.

After the development of the student status data warehouse, the university management system evolved due to changes in business requirements of users. The student status changing process was redesigned in such a way that every change of student status is registered as an order. Before, the history of changes of student statuses was not retained in the system, but the actual student status was stored in the table Studies. After the student status changing process was redesigned, two new tables were created:

- The table Group_Order was added to store data about order, which can include changes in status of several students: order number, type of an order (i.e. enrolment into study program of new students, change in study semester of existing students, direction of students to final examinations, graduation order, change of funding), type of an order reason (i.e. not meeting obligations), order status (prepared, executed or cancelled), date of an order and date when an order was registered in the system.
- The table Student_Order was added to store information about change of status of a particular student, where Prev_Value is the previous status value and New_Value is the changed status value.

Due to evolution of the student status changing process, operations performed by users of the university management system also changed. Therefore, it was decided to evolve the student status change data warehouse basing on the analysis of the activity of users at the data warehouse source system. The results of the application of the proposed method to the improvement of the student status change data warehouse are discussed in Sect. 6.6.

4 Metadata for Evolution Support

In our method, we take advantage of the existing data warehouse schema and mappings of its elements to data source elements, as well as existing reports defined on the data warehouse schema. In our approach, a data warehouse is a part of the data warehouse evolution framework [15].

The operation of all tools of the framework is based on the data warehouse repository (Fig. 3), which stores data warehouse metadata. Data warehouse repository stores schema metadata, which are data about data warehouse schema versions that are necessary for definition and execution of reports, including links between different versions. Data warehouse schema is described by three interconnected metadata

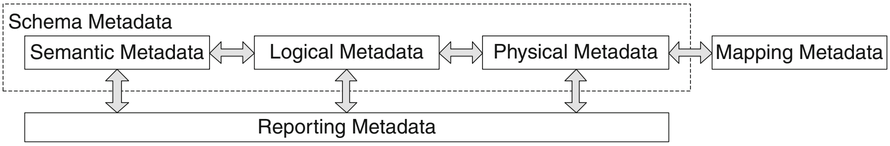


Fig. 3 Data warehouse repository

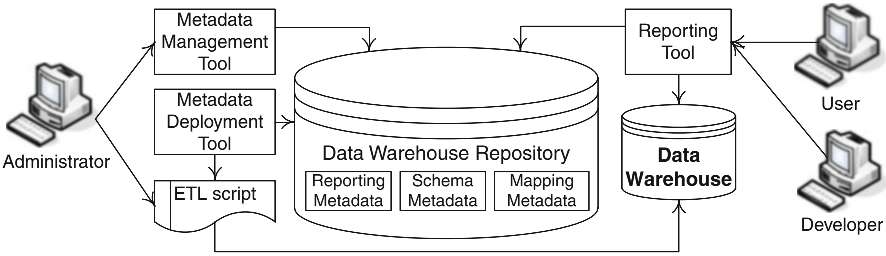


Fig. 4 Data warehouse evolution framework (part)

layers [16]. Logical metadata are used to describe the multidimensional data warehouse schemata and their versions. Physical metadata describe storage of a data warehouse in a relational database. Semantic metadata describe data stored in a data warehouse and data warehouse elements in a way that is understandable to users.

In addition, in the data warehouse repository, there are also the reporting metadata, which describe structure of reports predefined by the developers.

Data warehouse repository contains also mapping metadata, which define the logics of ETL processes. The mapping metadata describe data source elements (such as tables and columns in a relational source database or files) and the way how these source elements are used for data extraction, transformation and loading into target data warehouse tables. It is possible to infer dependences between data warehouse elements and source elements from the mapping metadata [17].

Different kinds of metadata are interconnected. Semantic metadata describe report's items from the reporting metadata and data warehouse schema elements from the logical metadata. Data warehouse schema elements from the logical metadata correspond to tables and table columns described in the physical metadata. Items of reports defined in the reporting metadata are obtained from table columns described in the physical metadata and correspond to data warehouse schema elements from the logical metadata. Mapping metadata define the correspondence between data source table columns and data warehouse table columns described in physical metadata.

The parts of the data warehouse evolution framework essential for discovering of improvements of a data warehouse schema and reports are shown in Fig. 4.

One of the basic elements of the framework is the reporting tool. Data warehouse users work with the reporting tool that allows to create ad hoc reports or to display reports predefined by developers as tables and graphs and analyse report data using hierarchies and other OLAP features. Using links between data warehouse schema versions in the schema metadata, the reporting tool can run queries on data, which correspond to multiple data warehouse schema versions or one version. Data warehouse repository is administered by the metadata management tool used to design a data warehouse schema and specify ETL processes. ETL process scripts are generated by the metadata deployment tool that uses the mapping metadata from the metadata repository. The data warehouse administrator executes generated ETL scripts.

5 Information Sources for Query Usage Study

In the operational systems different types of log files are used for operational goals: e.g. for recovery, auditing and optimization. In the data warehousing world, logging is used often for detecting changes in data of source systems. Every change of source data can be immediately captured by means of source applications, by triggers in source databases, by transaction logs, or afterwards by detecting changed records based on data fields or by comparing two copies of the same data source. Some of these techniques can be applicable also, when the usage practice of the data from source systems is analysed. We will examine a source system that collects different logs about functions and data usage.

The function usage is logged by a built-in feature of the application system that is always invoked when a certain function is performed. The format of such log consists of the following attributes: *Function_name, user* and *date*. This log can exhibit the most frequently used functions. Further investigation is needed to analyse the data manipulation statements that are used within these functions.

The data changes are logged by triggers. Each table has its own trigger for logging insert, updates and delete statements. We assume that triggers log the changes in one log table with the following structure: *Table_name, user, date, change_type* and *change_text*. The statistics about changes for each table and attribute allows discovering the leading ones.

A special kind of functions is reports with a wide range of parameters that can be used to get a specific report, to filter necessary data. The queries in reports represent also the information requirements of users. Thereby, logging of report usage is particularly interesting for the study. We assume that the format of such log is the following*: Function_name, user, date* and *parameter_text*, where *parameter_text* contains values in the format *p1=<value1… pN=<valueN*>, where *p1…pN* are in the format <*table_name.column_name*>.

In addition to the log files, the structure of data manipulation statements used in functions or procedures of source systems examined together with the function usage statistics can also provide information about users' actions. Two kinds of information sources can be prepared: the first one about insert and update statements and the second one about select statements.

The structure of the first source is the following: *Function_name, Action_Type, List_of_columns* and *Where_condition*, where *Action_Type* is update or insert.

List_of_columns and *Where_condition* contain the column names in the format <*table_name.column_name*> **.**

The structure of the second source is the following: *Function_name, Joins, Where_condition, Group_By, Columns_Select* and *Aggregation,* where *Joins, Where_condition, Columns_Select* and *Group_By* contain columns in the format <*table_name.column_name*>. *Aggregation* contains column name in the format <*table_name.column_name*> and the aggregation function used, e.g. *SUM, COUNT*.

6 Query-Driven Method

The proposed query-driven method for recommending data warehouse schema changes and new reports is composed of five steps described in this section.

6.1 Preprocessing of Information Sources

The first step of the method is preprocessing of the data available about system usage identified in Sect. 5. As a result of this step, two usage tables are constructed.

The first one is the *data modification table*, which contains data about the frequency of insert and update statements performed by functions with tables. The columns of the data modification table are the following: *Function_name, List_of_columns* and *Where_condition, count*. Since the precision of date columns in the function usage log and in the insert, update and delete log table is not less than 1 s, we assume that in one second one user could not insert or update data in one table using different functions. Besides the structure of data modification statements used in functions is available. This is why it is possible to relate the particular function, which was launched by the user, with the particular insert or update statement registered in the log table by trigger. The number of data modification statements with the same columns of the same table is summed up for the analysed period of time for all users.

The second usage table is the *data selection table*, which contains data about the frequency of select statements performed by functions with tables. The columns of the data selection table are the following: *Function_name, Joins, Where_condition, Group_By, Columns_Select, Aggregation, count*. To generate the records of the data selection table, data from the report usage log are combined with data about structure of report functions. The number of data selection statements is summed up for the analysed period of time for all users.

Two kinds of previous usage tables allow to restrict the set of tables taking into account only the functions used by a particular business process or some related processes that will be the focus of prospective measurement by means of data analysis performed within a data warehouse.

6.2 Analysis of Data Modification Statements

After the usage tables are prepared, data in these tables are analysed to obtain potential data warehouse schema elements. At first the data modification table is analysed. For each column, we calculate the number of times it was updated or inserted, i.e. we summarize *count* for every occurrence of the column in *List_of_columns* of the data modification table. Since *List_of_columns* of this table stores updated or inserted columns, we choose the Top-N columns with the biggest number of times they were updated or inserted and consider such columns as potential measures. Several potential measures belonging to one source table are united into a potential data warehouse fact table.

Then we identify the potential dimension attributes from the Top-N columns with the biggest number of times they were used in *Where_condition* in the data modification table. We assume that potential fact measures could be analysed together with potential dimension attributes if they are used together in one data modification statement. So if a potential measure is included in *List_of_columns* and a potential attribute is used in *Where_condition* in one data modification statement, we connect this potential dimension attribute to the potential fact table. Several potential attributes belonging to one source table are united into a potential data warehouse dimension table.

6.3 Analysis of Data Selection Statements

The information in the data selection table is analysed with the purpose to discover potential dimension attributes, possibly measures and connections between dimension tables and fact tables. During this analysis, we consider the Top-N columns, which are used the most frequently in *Where_condition* or *Group_By* as potential dimension attributes. The Top-N columns with the biggest number of times they were used in *Aggregation* are considered the potential measures.

We also analyse source columns included in *Columns_Select*. If such column is used in a query, which contain also the aggregation functions recorded in *Aggregation*, then such column is a potential dimension attribute. Besides, source columns used in *Columns_Select* in queries without aggregation functions are considered as potential measures if their data type is numeric, otherwise such columns are considered as potential attributes.

Several potential measures belonging to one source table are united into a potential data warehouse fact table, and several potential attributes belonging to one source table are united into a potential data warehouse dimension table. Source columns included in *Joins* are usually primary key or foreign key columns used to join tables. A join between two tables can reveal a connection between a potential fact table and a dimension table as well as a connection between two dimension tables. In the latter case, this connection can be considered as a link between hierarchy levels of the same dimension.

6.4 Conflict Resolution

Analysing usage data, it is possible that the same source column can be identified as potential fact measure and as a potential dimension attribute. In such case we analyse the data type of the column. If the data type of the column is numeric, then we consider such column as a measure. If the data type of the column is character or date, then we refer to such column as to an attribute, but the change of value of this attribute should be considered as a potential measure.

6.5 Generation of Potential Improvements

In the last step of the method, the potential data warehouse schema elements are compared with the existing data warehouse schema. Mappings between source table columns and existing target data warehouse elements are used in this comparison. If a potential data warehouse schema element is already included into the existing data warehouse schema, then it is ignored. Non-existing potential data warehouse schema elements are recommended to be added to the existing data warehouse schema. As a result, the potentials improvements are generated as data warehouse schema changes supported by the data warehouse evolution framework, such as addition of a new dimension attribute or fact measure, addition of a new dimension or fact table, connection of a dimension to a fact table, addition of a new dimension hierarchy or new hierarchy level. If a data warehouse administrator decides to accept a potential change, it is processed according to the predefined procedure [16], which makes the necessary modifications to the physical data structure of the data warehouse, the logical, physical and mapping metadata. As a result of such modifications in physical data structure and metadata, a new data warehouse schema version is created.

Separately, select statements, which are executed the most frequently at the data warehouse source system, are obtained from the data selection table and are recommended to the data warehouse developer as potential reports.

6.6 Case Study: Improvements for the Motivating Example

The proposed method was applied to the student status change data warehouse introduced in Sect. 3. After the analysis of student status changing activity of users in the university management system, the following table columns are encountered frequently in *Where_Condition, Group_By* and *Columns_Select* of the data selection table:

- Name of the table Type
- Order_Date and Order_Number of the table Group_Order
- Prev_Value and New_Value of the table Student_Order

These columns are considered as potential dimension attributes. So the proposed method recommends to extend the data warehouse schema depicted in Fig. 1 with the dimensions Group_Order, Student_Order and Type. However, after the manual revision of recommended improvements, it was decided to unite dimensions Group_Order, Student_Order and Type in one dimension Order with the following attributes: Order_Date, Order_Number, Prev_Value and New_Value. Besides, the column Name of the table Type was added as three attributes of the dimension Order: Order_Type, Reason_Type and Status_Type.

As a result of the predefined dimension and attribute addition procedures, the new schema version of the student status change data warehouse was created, which, in addition to the data structures depicted in Fig. 1, contains also the new dimension Order with attributes obtained during the analysis of users' activity in the university management system.

7 Conclusions and Future Work

The proposed method allows to keep track of actual analysis tasks performed in operational data sources of data warehouses and adjust the existing data warehouse schema according to the most frequent queries. The method is used to recommend necessary changes for a new data warehouse schema version.

The method in the part of analysing the query usage and discovering the elements of candidate schema for a data warehouse can be used alone, without the versioning support. In this case some evaluation of candidate schemata by users is needed. In the case of an existing data warehouse, its schema serves as a basis of comparison to filter completely different elements of schema not to be included in a new version of schema. They can be stored for later manual processing.

A data warehouse schema also serves as a representation of the existing data warehouse analysis needs that our proposed method allows to extend with new schema elements (after manual revision and acceptance). Therefore, the influence of usage of operational reports of data source systems on data warehouse schema changes is minimized.

During the case study, we have applied the proposed method for the improvement of the student status change data warehouse. We have identified the necessary changes for the data warehouse schema evolution. After discussion, evaluation and refinement of these changes, the new improved schema version of the data warehouse was developed.

We are working on implementation of the method and evaluation of constructed models to be convinced that the new version of the data warehouse is better adjusted to real analysis needs. Also usage patterns of existing data warehouse elements and user profiles in source systems can be examined to improve the method.

Acknowledgments This work has been supported by ESF project No. 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044.

References

1. Westerman P (2001) Data warehousing using the Wal-Mart model. Morgan Kaufmann, San Francisco
2. Kaldeich C, Oliveira J (2004) Data warehouse methodology: a process driven approach, vol 3084, LNCS. Springer, Berlin, pp 536–549
3. Giorgini P, Rizzi S, Garzetti M (2005) Goal-oriented requirement analysis for data warehouse design. In: 8th ACM international workshop DOLAP, pp 47–56
4. Golfarelli M, Maio D, Rizzi S (1998) The dimensional fact model: a conceptual model for data warehouses. Int J Coop Inf Syst 7(2&3):215–247
5. Jones ME, Song I (2005) Dimensional modeling: identifying, classifying and applying patterns. In: Proceedings of DOLAP '05. ACM, New York, pp 29–38
6. Golfarelli M, Lechtenbörger J, Rizzi S, Vossen G (2006) Schema versioning in data ware-houses: enabling cross-version querying via schema augmentation. Data Knowl Eng 59(2):435–459
7. Kuechler B, Vaishnavi VK (2008) On theory development in design science research: anatomy of a research project. Eur J Inf Syst 17(5):489–504
8. Niemi T, Nummenmaa J, Thanisch P (2001) Constructing OLAP cubes based on queries. In: Proceedings of the 4th ACM international workshop on data warehousing and OLAP. ACM Press, pp 9–15
9. Pang C, Taylor K, Zhang X, Cameron M (2004) Generating multidimensional schemata from relational aggregation queries. In: Zhou X et al (eds) WISE'2004, vol 3306, LNCS. Springer, Berlin, pp 584–589
10. Nair R, Wilson C, Srinivasan B (2007) A conceptual query-driven design framework for data warehouse, vol 19, Transactions on engineering, computing and technology., pp 141–146
11. Zhang J, Wang W, Liu H, Zhang S (2005) X-warehouse: building query pattern-driven data warehouse for XML data. In: Proceedings of the international world wide web conference, pp 896–897
12. Body M, Miquel M, Bedard Y, Tchounikine A (2002) A multidimensional and multiversion structure for OLAP applications. In: ACM 5th international workshop on data warehousing and OLAP. ACM, McLean, VA, pp 1–6
13. Wrembel R, Bębel B (2005) Metadata management in a multiversion data warehouse. In: On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE (pp. 1347–1364). Springer Berlin Heidelberg
14. Velegrakis Y, Miller RJ, Popa L (2003) Mapping adaptation under evolving schemas. In: 29th international conference on VLDB. Morgan Kaufmann, Berlin, Germany, pp 584–595
15. Solodovnikova D (2007) Data warehouse evolution framework. In: Proceedings of the Spring Young Researcher's Colloquium on Database and Information Systems (SYRCoDIS'07), Moscow, Russia. http://ceur-ws.org/Vol-256/submission_4.pdf
16. Solodovnikova D (2010) Metadata to Support Data Warehouse Evolution. In Information Systems Development (pp. 627–635). Springer US
17. Solodovnikova D, Niedrite L (2006) Data warehouse adaptation after the changes in source schemata. In: 7th international Baltic conference on databases and information systems, pp 52–63

Capabilities and Affordances of Virtual Worlds for People with Lifelong Disability

Karen Stendal and Judith Molka-Danielsen

Abstract Using technology is of great value for people with lifelong disability (PWLD). The Internet can help PWLD to be more independent, to be socially active, and to participate in a range of activities. Virtual worlds (VWs) offer an environment with technology capabilities for interaction, rendering, and communication. The ability to take advantage of these capabilities may depend on the technology and the ability of the person utilizing it. Using a qualitative study we aimed to explore the differences of ability required to use these capabilities and make capabilities into affordances for PWLD. We found there were differences in PWLD's ability to utilize the capabilities offered and conclude there is a need for standards for universal design to create sustainable VWs suitable for all.

1 Introduction

Technology offers a variety of capabilities for socialization and inclusion. One of the latest technologies to offer communication, rendering, and interaction capabilities is metaverses [7]. Metaverses, also known as virtual worlds (VWs), are immersive, online 3D representations of the physical world where individuals interact through avatars. An avatar may have a resemblance to the user creating it or can be created to represent how the user wishes to present him/herself [1]. There are multiple reasons why people engage in VWs, including business, work, hobbies, socialization, and enjoyment.

K. Stendal (✉)
Molde University College, University of Agder, Service Box 422, 4604 Kristiansand, Norway
e-mail: karen.stendal@uia.no

J. Molka-Danielsen
Molde University College, Postbox 2110, 6402 Molde, Norway
e-mail: j.molka-danielsen@himolde.no

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_42,

A fairly new research area related to VWs is exploring how people with lifelong disability (PWLD) are using and experiencing metaverses. PWLD are active users of VWs [3], but there is little empirical research showing the challenges and opportunities offered to this group [20].

The United Nations (UN) Convention defines people with disability as follows: "*Persons with disabilities include those who have long-term physical*, *mental, intellectual, or sensory impairments which in interaction with various barriers may hinder their full and effective participation in society on an equal basis with others*" ([11], p. 1220).

Due to known participation and inclusion challenges, this study focuses on people with lifelong disability, where the impairment occurred before the age of 22 and the impact of the impairment is lifelong. Lifelong disability includes intellectual disability, autism, deafness, epilepsy, polio, and cerebral palsy [2]. To date VWs and their promised values have been researched in the context of education and rehabilitation [18]; however research concerning how PWLD are able to take advantage of the technology capabilities offered by VWs is yet limited. Psychologist J.J. Gibson defined affordances as all "action possibilities" latent in the environment, objectively measurable, and independent of the individual's ability to recognize them but always in relation to the actor and therefore dependent on their capabilities [8]. In the context of VWs, we understand this notion to mean that VWs offer a variety of capabilities which in turn may become an affordance when the user is able to take advantage of the capability promised. Motivated by the theoretically defined set of technological capabilities [7] and the potential of turning them into affordances, the aim of this paper is to identify the capabilities and affordances for PWLD when using VWs. The research question is:

What capabilities and affordances do virtual worlds offer people with lifelong disability?

This study contributes to the understanding of the challenges experienced by PWLD to benefit from the potential opportunities offered by virtual worlds. It also contributes to the theoretically defined technological capabilities by applying them in a different context than Davis et al. [7].

2 Technology Capabilities of Virtual Worlds

Metaverse capabilities have been defined as a set of capabilities for communication, rendering, interacting, and team process [7]. In the context of this paper, we consider the first three capability categories, while team process is not focused since our analysis does not include the virtual team level.

2.1 Communication

Communication is intrinsic to humans; the ability to connect and communicate with others is important in most environments and contexts [7]. VWs offer communication possibilities through both text chat and a voice feature. This enables people to communicate in real time and obtain immediate feedback on their comments or statements [6]. In the physical world, we rely on more than the spoken word alone, for example, by using body language and facial expression. The possibility to animate an avatar to show familiar body language supplies a great opportunity to obtain rich communication [16]. However, it may be difficult to use this multimodal communication in VWs, due to the complexity of such animation. In addition, online communities may have their own written language, which in every day spoken conversation is not used [4]. Through experience with the communication channels, the communication richness increases [5]. Communication support in the VW context means there are multiple ways to communicate. The VW Second Life (SL) offers public text chat, private instant messaging (IM), group IM, public voice chat, private voice chat, and group voice chat. However, to use these different communication tools effectively requires training, and a lack of training may discourage potential users from adopting the technology [7].

2.2 Rendering

Rendering is defined to be the process of creating or executing lifelike images on the screen [7]. One of the important factors offered by VWs is the embodiment which the avatar represents [13]. Through an avatar people in VWs are able to create and play with a representation of them in a way not available in the physical world. Many will create avatars that cultivate an extreme look, for example, with clothes and hair the individual would not wear in the physical world [14]. VWs are a three-dimensional reproduction of the physical world, and this is described as vividness [7].

2.3 Interaction

Interaction is defined as the extent to which users can participate in modifying the form and content of the VW in real time [7]. VWs, such as SL, enable users to create the content; in other words people are able to build, create, and modify objects, environments, and their own representation in real time [9]. VWs also provide the users with the ability to move around in an environment free from physical laws [19]. Consequently VW users can fly and teleport. Thus, for PWLD, this allows an individual who in the physical world uses a mobility aid to move around freely, to walk, run, or dance [21].

2.4 Virtual Worlds and People with Lifelong Disability

Accessibility to technology has been recently discussed [10] and is also discussed in the context of VWs. Universal design is a principle, which can be applied for

VWs in order to optimize a well-designed virtual environment for all people, regardless of their disability or the assistive technology needed [10]. There are practical limitations to implementing such solutions, such as cost and complexity. Also, the needs of PWLD are diverse; in other words a solution for one type of disability may not be applicable to another type of disability [10]. This can be seen by the implementation of the voice feature in the virtual world where people who are deaf once again are faced with their hearing impairment and cannot easily participate in the same activities as their nondisabled peers [6].

We have conducted an exploratory qualitative study to explore how PWLD experience the technology capabilities offered by VWs and how turning these into affordances may be a challenge for different groups within the disability group.

3 Research Method

This study includes two groups of participants, a group of novice users and a group of experienced users of Second Life (SL). All participants are over the age of 18, diagnosed with a lifelong disability, able to sign an informed consent form, and have access to and are able to use a computer with broadband. Aligned with previous research SL was chosen due to (1) the large number of users, (2) the range of activities available, and (3) the wide range of opportunities (i.e., business and education) [17]. Because of these particular features, SL is a suitable platform for both novice and experienced users to explore and prosper in the VW. All participants gave informed consent and were told they could withdraw from the project at any time with no reason given and no penalty. Ethical clearance from the Norwegian Social Science Data Services (NSD) for the research was obtained.

Table 1 shows a list of participants in this study. To ensure the anonymity of the participants, all avatar names in this paper are pseudonyms. Participants' disability

Table 1 List of participants in this study

Group	Participant	Disability	Location	Primary language
Group 1	Pevit Torana	Intellectual	Norway	Norwegian
Novice	Mix Mofat	Intellectual	Norway	Bulgarian/Norwegian
	Solvita Silka	CP	Norway	Norwegian
	Trinaka Lika	Intellectual	Norway	Norwegian
	Rolatina Endora	Intellectual	Norway	Norwegian
	Gjagra Gralt	CP	Norway	Norwegian
	Missara Melsa	Intellectual	Norway	Norwegian
	Siltar Siana	ASD	Norway	Norwegian
Group 2	Agonra Sircka	Physical	USA	English
Experienced	Kalnika Gublic	Physical	USA	English
	Sunger Alista	HI	USA	English
	Ahroun Wolf	ASD	USA	English
	Maria Butterfly	Physical	USA	English
	Kirana Merkini	HI	South Africa	Finnish/English
	Landira Crunge	HI	USA	English

included cerebral palsy (CP), Asperger's syndrome (AS), hearing impairment (IM), mild to moderate intellectual disability (ID), and other physical disability (PD).

With the first group we aimed to explore how novice users with lifelong disability experience the VW. This group consisted of eight participants recruited through organizations in their local community such as adult learning centers. Over a period of 8 weeks in 2011–2012, the eight participants met with the first author (hereafter referred to as "the researcher") in weekly sessions of 1½ h and engaged in different activities in SL. In the fourth and eighth week of the study, all participants were interviewed about their experiences in the VW. Interviews were conducted by phone, to ensure the SL sessions were solely used for activities and interaction.

The second group of participants consisted of seven experienced users. These users were recruited with the help of Virtual Ability Inc. (ref virtualability.org), which operates an island in SL to support people with disability entering into the VW. During two in-world presentations of the project, people attending the presentation were invited to participate in this study. The experienced users participated in a longer in-depth interview, approximately 2 h, exploring their experiences with the virtual world. All interviews with the experienced users were conducted in private IM or private voice chat. The active SL experience level of the participants in this group ranged from 1 to 7 years.

The focus of the data collection was to map the challenges and affordances offered by VWs for PWLD. Based on the capabilities presented by Davis et al. [7] and Gibson's definition of affordances, our data analysis identified the challenges and affordances offered by VWs to PWLD. Answers from different individuals to common questions were grouped together and along with observations were analyzed for different perspectives on the issues [15], which were considered important for understanding how people with lifelong disability experience the capabilities offered by VWs.

4 Findings: Affordances Offered by Virtual Worlds for PWLD

Observation and interviews conducted with the novice users of SL were in Norwegian, and any quotes from these participants have been translated to best reflect their meaning. The interviews with the experienced users of SL were conducted in English, and quotes are taken directly from the interviews. Findings are presented related to the technological capabilities presented by Davis et al. [7].

4.1 Communication

While SL offers multiple communication channels, depending on disability, utilization of these channels presents challenges. For the participants with hearing impairment, the voice feature was not an option. Although they were able to use the text chat function, they reported challenges when communicating in the virtual world. "*What bothers me the most, is their* [other avatars] *refusal to type. Especially if the only reason they refuse is because they don't want to type, not because they can't type*" (Landira Crunge). On the other hand, those who experience difficulty writing due to their physical or intellectual disability feel they are not able to provide the immediate feedback to others as they would like, if the voice feature is not available or malfunctioning. Maria Butterfly uses the software Dragon (see www.nuance.com) to convert voice into text in SL. However, this creates some lag in the conversation which in some cases gives her the feeling of not being able to expand and explain as well as she would like. If one person with literacy impairment and one with communication impairment try to communicate, there could be some tension; however one participant noted that there are ways to work around the challenge. "*If a person cannot type, I will however try to find someone in-world willing to translate what they hear and type it to me. That's if the person CAN not type…different from that they are unwilling to type to me*" (Sunger Alista).

When observing the novice group in SL, the researcher noticed that being able to hear how the participants reacted to different activities and situations enriched the data collection. Through laughter, cheers, and comments, the lack of body language did not appear to affect the experience of SL. Furthermore the Norwegian participants struggled with English. The fact that most communication in SL is in English discouraged some of the participants from trying to make contact with others.

Through sessions with the novice users, multiple communication channels were used to support the conversations. Public voice chat was used when available. In addition, public chat, private IM, and private group IM were used. The novice users expressed being distracted by the variety of communication support; on the other hand they also noted the variety of communication channels was a good thing. Once the researcher and Missara Melsa were talking in the public voice chat and were interrupted by a stranger who also spoke Norwegian. Missara Melsa found this experience a little frightening, because she did not know the person who started talking and could not see the avatar representing the person. Because of this encounter we started a private IM instead which made Missara Melsa feel more comfortable.

4.2 Rendering

All the participants, except Ahroun Wolf, use human avatars on a regular basis. However, the ability to be anonymous and create their avatar to their liking is of value to all. Ahroun Wolf chose an animal avatar because he identifies with the wolf.

It is important for participants to be able to choose what to disclose, and most participants did not disclose any disability in SL. Only two were open in showing or disclosing their disability. Sunger Alista has posted in her profile information that she has a hearing impairment and only communicates through text. Agonra Sircka chooses to have an avatar which uses a wheelchair, as he does in the physical world.

Kalnika Gublic explained that in the physical world, she has to "convince" people she is a person, and she avoids that in the virtual world by not displaying her disability. Also the novice users embrace the possibility for anonymity. "*It fits me very well* [to be anonymous]. *Because I have had problems with people in Facebook, so I like to be anonymous in places like this*" (Missara Melsa).

Objects in SL are 3D representations of physical world objects, which represent vividness in SL. This creates a rich sensory environment for users of SL. Some of the novice participants used available objects found in SL in unexpected ways. For example, when visiting Virtual Africa to view elephants, zebras, giraffes, and other animals present, the novice users decided to run around in the environment and sit on all the animals. For them the ability to sit on the animals was just as important as the animals being there and created pleasurable situations for them to enjoy.

4.3 Interaction

Being able to modify objects is dependent on the island where the object is placed. However, we noticed there is a required skill level that has to be accomplished before an object can be modified. Through the eight sessions with the novice users, they were able to modify their own avatar; however this is a multiple step operation and requires training and support. Some of the experienced users had developed the required skills. "*I also enjoy making things- well more modifying things. Like especially modifying them so that they are more easily usable*" (Sunger Alista).

SL also offers the opportunity to build and create new objects in the virtual environment. A few of our participants are skilled builders in SL and taking in-world classes to learn how to build objects. They noted this is another learning curve to overcome, but when they accomplish the basics, the ability to use their creativity in SL is something they enjoy very much. "*Because I'm retired, it gives me something to do and satisfies any creativeness I might want to try*" (Landira Crunge).

The findings of this section are summarized in Table 2.

5 Discussion

The capabilities presented by Davis et al. [7] were available for all participants; however dependent on participants' disability and experience level, some of the capabilities did not turn into an affordance based on Gibson's definition. Figure 1 shows how together technology capabilities and human ability create affordances for PWLD in VWs.

Communication is an important issue for all humans, and technology is proving to be an important tool for PWLD to overcome some of the barriers they encounter in the physical world. VWs offer the possibility to communicate through public IM/voice, private IM/voice, and group IM/voice. Although Maria Butterfly explained

Table 2 Summarized findings

Disability	Challenge in VW	Available solution	Assessment of available solution
HI	Unable to use voice feature	Voice-to-text software	Lag in conversation
		Text chat	Problem for people with low literacy
		Find a human translator	
ID	Unable to understand nonnative languages	Limit communication to native speakers	Limiting of new contacts
	Too many communication functions	Use only a few interface communication functions	
ID	Use of VW interface	Training	Initial support is a critical factor to avoid dropout
PD, CP	Use of keyboard/mouse control	Seek alternative controls or browsers	No universal design standards for virtual world browsers
All	Selective disclosure of private information	Personalization of avatar Information in public profile	Offers opportunity for anonymity

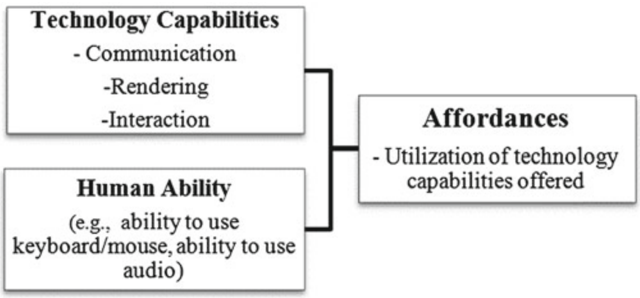


Fig. 1 Technology capabilities and affordances of VWs for PWLD

she uses voice-to-text software to be able to participate in text chat, common software of this kind does not interface well with VWs [10]. In addition, those participants with a hearing impairment struggle when other people they meet are unable or unwilling to use text chat when communicating with them. Carr [6] stated in earlier research educators need to be aware of the possible impact VW voice feature may have on deaf students. She noted deaf students may once again feel the impact of their impairment in the VW.

There were some difficulties using the different communication channels, such as when the researcher asked a question in the private group IM, the participants answered in the public chat. The variety of communication channels showed to be confusing at times for those participants with an intellectual disability. Through the observation of the novice users, we see their ability to take advantage of the offered communication channels increase and become richer over time. This can be related to channel expansion where richness of communication through different channels will increase by experience [5]. However, as mentioned in the concern of feedback, the range of cues and channels available to PWLD depends on both the type and level of disability experienced and the functionality incorporated in the VW. In situations where the voice feature was not available or functioning, the novice participants used short messages such as "haha" and "☺" to express their feelings about the situation. However, for those without knowledge about online language trends, having to express cues in textual form may be difficult [4].

Personalization of avatars is an important factor in VWs [14]. This is also true for the participants in this study, where only two of the participants chose to disclose their disability and one participant chose to represent himself as a wolf in the VW. Vividness in SL is represented by the environment being similar to the physical world, though without the feel and smell of the physical world [7]. The embodiment through an avatar still creates a feel of being present in the VW [13], which most of the participants experienced.

Mobility is an important issue for PWLD, particularly those with physical disability. In SL, both novice and experienced users noted this is an important affordance in VWs. The ability to walk in the VW may be important for those who use a wheelchair in the physical world. Walking allows them to experience the environment in a new way. Being able to visit new locations, which in the physical world may not be available to them, is seen as a great value. However, we observed that for the novice users the abilities to walk, fly, and teleport are operations which take time to master and support is needed. Since mobility in VWs is dependent on use of keyboard and mouse skills, those with a physical disability may find it difficult to accomplish this affordance [10]. Being able to do tasks and participate at a deeper level is important for the self-esteem [12]. However, the novice users did not acquire the level of skills within the eight sessions to be able to modify and create objects. Table 3 summarizes the findings from this study.

In addition to the specific technology capabilities, it is important to note that the learning curve for use of SL was steep for most participants. Maria Butterfly said she would not have continued using SL if she had not had help to overcome the entry barriers. Landira Crunge, at one point, stopped using SL due to the difficulties of entry. Through observing the novice users, we noticed the difficulty of using the application. Often multiple steps are required to perform tasks in-world for which a novice user with intellectual disability needs support. While the learning curve for VWs is steep for any individual, those with an intellectual disability may face an entry barrier difficult to overcome on their own [10], but not impossible. In summary, based on our assessment of the challenges, we identify a need for universal standards for interface design for VWs, such as W3C's accessibility standards (ref w3.org). We recommend adherence to this initiative in the design of VW interfaces.

Table 3 Capabilities, affordances, and challenges of VW for PWLD

	Capability	Affordances	Challenges
Communication	Feedback	Synchronous communication in both voice and text	Physical disability may cause text-based communication to be a challenge. Hearing impairment exclude the use of voice feature
	Multiplicity of cues and channels	Communication with a variety of users in different channels and different levels of privacy	For people with a hearing impairment or for others when voice feature malfunctions/ unavailable, text may not show underlying cues in communication
	Language variety	Communication with worldwide users of VWs	Slang and online language depends on experience with online medias
	Channel expansion	Richer communication through multiple channels	Steep learning curve to learn how to use communication channels
	Communication support	Multiple communication channels to support different needs	Multiple channels may be challenging and distracting for those with intellectual disability and novice users
Rendering	Personalization	Creating own avatar. Choice of disclosure, playing with identity and anonymity important to all participants	Takes time to master, especially for those with intellectual disability
	Vividness	Rich sensory representation of the physical world	Novice users use objects in unexpected ways
Interaction	Interactivity	Modify content in VWs	Dependent on experience and skill level
	Mobility	Teleporting/flying/ walking	Control the avatar through keyboard and/or mouse clicks
	Immediacy of artifacts	Building and creating content in VWs	Dependent on experience and skill level

6 Conclusion

This study is limited in that it is based on a small sample size of the representation of different impairments. We recommend a refinement for further work. We suggest that more in-depth empirical studies of a range or representatives from each disability group will facilitate the identification of specific challenges for each group. Additionally we suggest that eight sessions in SL may not be enough time for novice users to achieve the required skill level to fully experience the VWs.

In conclusion, this study contributes with an identification of the affordances offered by VWs to PWLD. As seen in the discussion, all of the technology capabilities presented by Davis et al. [7] represent potential affordances for PWLD. However, the different forms of disability may hinder PWLD to utilize the promised capabilities into affordances. We identified several work-around solutions of the participants that enabled VW capabilities to become realized affordances, although these solutions have limited success. Another possible solution or approach is to build sustainable VWs after universal design principles, that is, to design and develop VWs and their interfaces that are based on the abovementioned principles and which take the different challenges met by PWLD into account (e.g., voice-to-text technology, support for not being able to use keyboard, and intuitive interfaces). In summary, our focus on the experiences of PWLD contributes by extending the use of the theoretically defined technological capabilities to a different setting than reported in former research by Davis et al. [7].

References

1. Adrian A (2009) Civil society in second life. Int Rev Law Comput Technol 23(3):231–235
2. Ansello EF, O'Neill P (2010) Abuse, neglect, and exploitation: considerations in aging with lifelong disabilities. J Elder Abuse Negl 22(1/2):105–130
3. Babiss F (2009) Heron Sanctuary. Occup Ther Mental Health 25(1):1–3
4. Campbell R (2011) Online language: new messages in new media? Qual Market Res 14:1
5. Carlson JR, Zmud RW (1999) Channel expansion theory and the experiential nature of media richness perceptions. Acad Manage J 42(2):153–170
6. Carr D (2010) Constructing disability in online worlds: conceptualising disability in online research. Lond Rev Educ 8(1):51–61
7. Davis A, Murphy J, Owens D, Khazanchi D, Zigurs I (2009) Avatars, people, and virtual worlds: Foundations for research in metaverses. J Assoc Inf Syst 10(2):1
8. Jones KS (2003) What is an affordance? Ecol Psychol 15(2):107–114
9. Jung Y, Kang H (2010) User goals in social virtual worlds: a means-end chain approach. Comput Human Behav 26(2):218–225
10. Krueger A, Stineman MG (2011) Assistive technology interoperability between virtual and real worlds. J Virtual Worlds Res 4:3
11. Leonardi M, Bickenbach J, Ustun TB, Kostanjsek N, Chatterji S (2006) The definition of disability: what is in a name? Lancet 368(9543):1219
12. Lidström H, Ahlsten G, Hemmingsson H (2011) The influence of ICT on the activity patterns of children with physical disabilities outside school. Child Care Health Dev 37(3):313–321
13. Mennecke BE, Tripplett JL, Hassall LM, Conde ZJ (2010) Embodied social presence theory. Paper presented at the 43rd Hawaii international conference on system sciences, Hawaii, US
14. Oishi L (2007) Surfing second life. Technol Learn 27(11):54–64
15. Patton MQ (1990) Qualitative evaluation and research methods, 2nd edn. Sage Publications, Newbury Park
16. Saunders C, Rutkowski AF, van Genuchten M, Vogel DR, Orrego JM (2011) Virtual space and place: theory and test. Manag Inf Syst Quart 35(4):1079–1098
17. Schultze U, Leahy MM (2009) The Avatar-self relationship: enacting presence in second life. ICIS Proceedings. Paper 12 (Vol. Paper 12)
18. Standen PJ, Brown DJ (2005) Virtual reality in the rehabilitation of people with intellectual disabilities: review. Cyberpsychol Behav 8(3):272–282

19. Standen PJ, Brown DJ (2006) Virtual reality and its role in removing the barriers that turn cognitive impairments into intellectual disability. Virtual Reality 10(3–4):241
20. Stendal K, Balandin S, Molka-Danielsen J (2011) Virtual worlds: a new opportunity for people with lifelong disability? J Intellect Dev Disabil 36(1):80–83
21. Stewart S, Hansen TS, Carey TA (2010) Opportunities for people with disabilities in the virtual world of Second Life. Rehabil Nurs 35(6):254

A Triple-Helix Model of Sustainable Government Information Infrastructure: Case Study of the eProcurement System in the Indonesian Public Sector

Fathul Wahid

Abstract Lack of sustainability is one of the problems in information system (IS) implementation in developing countries, such as Indonesia. More specifically, this is the case in eGovernment implementation. Using an interpretive research stance in a case study of the eProcurement system in the public sector, the study reveals that the concept of information infrastructure (II) can be used to understand the sustainability issue. Findings suggest that applying the concept of II in IS development and implementation can improve the sustainability of an IS in the public sector, especially one that is used across government agencies and levels. The paper proposes a triple-helix model of a government II composed of legal, managerial, and technical factors and recommends that these factors should be taken into account in an orchestrated fashion.

1 Introduction

Many problems must be dealt with before the implementation of eGovernment, or more generally information systems (IS), can achieve success in developing countries [1–4]. Previous studies have identified various factors responsible for these problems, including lack of commitment from political leadership, fragmented and uncoordinated organizational structures, unrealistic ambitions, a design-reality gap, and more generally the problem of sustainability [e.g., 3–5].

A case in point is Indonesia. In this country, the problem of sustainability is often caused by uncoordinated initiatives. Most of the local governments developed and maintained their own ISs. This practice created fragmented and incompatible ISs

F. Wahid (✉)
Department of Information Systems, University of Agder, Kristiansand, Norway

Department of Informatics, Universitas Islam Indonesia, Yogyakarta, Indonesia
e-mail: fathul.wahid@uia.no

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_43,

among the local governments and hindered the smooth vertical aggregation of information from the local to the national levels. This uncoordinated practice also led to inefficient use of the government budget.

Despite this problem, some initiatives have proven to be successful in developing ISs that have been used nationally across local governments. One of these is the eProcurement initiative, which is nationally coordinated by the National Public Procurement Agency (*Lembaga Kebijakan Pengadaan Barang/Jasa Pemerintah* [LKPP]). Since its inception in 2008, the eProcurement system has provided service to 650 government agencies at local and national levels [6]. The system has gone through various adjustments along the way.

Here, the system is understood as an ensemble, "only one element of a 'package' which also includes the components required to apply that technical artefact to some socio-economic activity" [7]. Hence, the system is considered as a socio-technical artifact. Additionally, based on its scope and reach, the system is also considered as a national information infrastructure (II).

The study seeks to answer the following main research question: *how can the sustainability of the eProcurement system be understood in the context of Indonesia?* The concept of II is used as a lens to understand the phenomenon under study, as the most important attribute of an II is that it is a permanent initiative and designed to have a long life [8]; thus it is expected to be sustainable. Among the contributions of this paper is the introduction a triple-helix model consisting of three factors (i.e., legal, managerial, and technical) to understand a sustainable government II.

The rest of the paper is structured as follows. Section 2 describes the current status of eProcurement practice in the Indonesian public sector. Section 3 presents the theoretical foundation of the study. Section 4 describes the research method, followed by presentation of the findings in Sect. 5. Section 6 discusses the findings. Section 7 presents contributions and suggestions for further research.

2 Setting the Scene

eProcurement has been formally adopted by the Indonesian government since April 2007 as part of its effort to curb corruption. As in many other developing countries, corruption is believed to be one of the main barriers to development in Indonesia. Since 2008 the eProcurement system has been implemented in some government agencies in both national and local levels. One of the main strategic objectives of this initiative is to improve the transparency and accountability of public procurement. This objective has gained ground in Indonesia because budget leakage in public procurement has reached an alarming level, with 10–50 % of the budget being misappropriated [9].

In Indonesia, public expenditures account for 30–40 % of total national spending. In some government departments, public procurement expenditures may even reach 70 % of total annual budgets. Public procurement expenditure is therefore important, as are the sound processes and proper management that must accompany effective measures to reduce and minimize the potential for irregularities and misconduct [10]. According to the national data, the use of the system from 2008 until February 2011 saved 12 % of the national procurement budget [6]. At the end of 2010, another Presidential Instruction (PI), No. 11/2011, was enacted to provide clearer guidance on how to use the system to combat corruption. A clear target was set: in 2012 at least 75 % and 40 % of the total procurement budgets at the national and the local levels, respectively, should adopt the system.

3 Research Framework

This section provides a theoretical basis for this study. Two main concepts are presented in this section, namely, sustainable ISs and II.

3.1 Sustainable Information Systems

In this study, sustainability is conceptualized as an attribute of an IS. This stance is in keeping with the literature. For example, from a knowledge perspective, Maruster et al. [11] defined a sustainable IS as that which adapts to its environment, involves relevant stakeholders, and supports the knowledge lifecycle (i.e., knowledge creation, knowledge evaluation, and knowledge integration and application). In the context of developing countries, Jacucci et al. [12] summarized the possible factors responsible for IS unsustainability. These factors include inadequate focus on local expertise, too-narrow interventions (often a sustainable IS requires a parallel reform of the respective sector), and technical bias of projects (inadequate focus on human resource development).

Kumar and Best [13] provided a more comprehensive definition of sustainability as an attribute of IS implementation, which included five aspects, namely, financial, social, technological, political/institutional, and environmental sustainability. There is also another perspective to understand IS sustainability: supply sustainability and demand sustainability [14].

An essential premise in all these different perspectives of sustainable systems is that the system must survive in the long term. How to achieve this remains sustained issue in the literature. In this paper, one way to address this is proposed: conceptualizing the system as an II.

3.2 Information Infrastructure

Various definitions of information infrastructure (II) can be found in the literature. Star and Ruhleder [15:113] stated that “infrastructure is a fundamentally relational concept. It becomes infrastructure in relation to organized practices." When studying II in the health sector, Braa et al. [3:383] defined II as "the technological and human components, networks, systems, and processes that contribute to the functioning of the health information system." Hanseth and Lyytinen [16:4] provided another definition that includes the properties of II: II is "a shared, open (and unbounded), heterogeneous and evolving socio-technical system (which we call installed base) consisting of a set of information technology (IT) capabilities and their user, operations and design communities."

Using a different perspective, Bygstad [8] saw II as an organizational form. As such, II is a permanent initiative; it is enacted, reproduced, and changed through daily use, has (to some extent) supplanted hierarchy as the coordination mechanism, and has borders enabled by standards; the members of the II share some common objectives, and the transaction costs are very low and decrease with size.

When defining infrastructure, the notion of *when* is more important than the notion of *what* [15]. They pointed out that something is an infrastructure only in relation to some practices. Accordingly, when defining eProcurement infrastructure, this study took into consideration its importance in relation to eProcurement practice, specifically in the public sector.

This study mainly adopted Star and Ruhleder's definition [15] consisting of dimensions that could be used to understand a national II [15, 17]. Each of these dimensions is briefly explained in Table 1.

4 Method

An interpretive case study strategy was adopted since the problem under investigation is practice based, whereby the experiences of the actors are important and the context of action is critical [18]. Data were mainly collected through formal and informal interviews, which were carried out in July, August, and November 2011. The subjects were a variety of key players in eProcurement and/or eGovernment implementation at various levels: a mayor, heads/vice-heads of departments (*dinas/badan*), heads of divisions, eProcurement officers and administrators, and a developer of the eProcurement system. Twenty-two interviews were conducted. Interviews lasted 30–150 min; most of them were recorded. Examples of questions in the interview are attached in Appendix.

I also participated in a national meeting of eProcurement agencies attended by more than 600 participants throughout Indonesia. In this meeting, some informal discussions with eProcurement officers from various local governments were also conducted. In addition, to increase the validity, data were also collected from written documents/reports and other secondary data such as news presented in the media, presentation slides, official websites, and field observations in three local governments (i.e., Yogyakarta, Sragen, and Jembrana).

I used the dimensions of II as template when coding the data. Subsequently, by considering the eProcurement system as a socio-technical artifact [7, 16], I adopted axial coding strategy to identify technical and nontechnical factors emerged from the data.

Table 1 Dimensions of II

Dimension of II	Description
Embeddedness	An II is said to be embedded if it "is sunk into, inside of, other structures, social arrangements and technologies" [15:113]
Transparency	The transparency of an II should be understood "in the sense that it does not have to be reinvented each time or assembled for each task, but invisibly supports those tasks" [15:113]
Reach and scope	An II has reach and scope beyond a single event or one-site practice [15]
Learned as part of membership	New participants acquire a naturalized familiarity with an II as they become members. The potential adopters then see an II as an object about which they will learn [15]
Links with conventions of practice	As a socio-technical artifact [19], II both shapes and is shaped by the conventions of a community of practice [15]. Thus, an II is enacted, reproduced, and changed through daily use [8]
Embodiment of standards	Standards are defined as "shared and agreed upon specifications among a set of communities" [16:16]. The standards allow an II to be plugged into other structures in a standardized fashion [15]
Built on an installed base	An II does not grow de novo from scratch [15], but it is built on an installed base. An installed base is defined as "the interconnected practices and technologies that are institutionalized in the organization" [20:583]. The definition implies that there are technical and nontechnical (i.e., legal and managerial) installed bases
Becomes visible upon breakdown	An II, which normally possesses an invisible quality when it is working, becomes visible when it breaks, such as when the eProcurement server is down [15]
Fixed in modular increments	Since II is large, layered, and complex, changing it takes time, negotiation, and adjustment [17], and hence it is evolving [16]. II's evolution is path dependent and can be both linear and nonlinear [16]

5 Findings

In this section, the development and implementation of eProcurement in Indonesia are presented in the light of the dimensions of II presented above. The study found that in addition to technical factor that determined the sustainability of the system, other two nontechnical factors emerged: legal and managerial. Altogether, the three factors (legal, managerial, and technical[1]) emerged in almost all dimensions of II. If the eProcurement system was seen from two different levels (national and local government), these three factors related to both supply and demand sustainability [14]. Table 2 summarizes the intensive links between the dimensions of II and these factors.

[1] In this study the legal factors include all the regulations and policies enacted or made by both national and local governments; the managerial factors include all the internally and externally oriented practices in the public sector in general (i.e., budget allocation, interorganizational collaboration); the technical factors include the availability, administration, and use of IT in general and of the eProcurement system in particular.

Table 2 Detailed links between dimensions of II and the involved factors

Dimension of II	Involved factors: Legal	Managerial	Technical
Embeddedness	• PD No. 80/2003[a] • PD No. 106/2007[b] • Local regulations were enacted to legitimate and guide implementation of the system	• The system was attached to the existing public procurement practice both at national and local levels • Budget allocation to support eProcurement initiative	• The system has been developed and maintained by a permanent in-house IT team • The system has been and will be integrated with the existing IT infrastructure
Transparency	• PD No. 80/2003 • PD No.54/2010[c]	• The eProcurement procedures are transparent and should be followed in any public tenders	• The system developed by LKPP could be easily installed in other agencies
Reach and scope	• PD No. 54/2010 • PD No.11/2011[d]	• The system has been used by more than 650 agencies • The system has been used for more than 32,000 tenders	• The system has been installed in 320 agencies • The eProcurement infrastructure at both national and local levels has been improved
Learned as part of membership	• LKPP issued regulations concerning the adoption of the system in other government agencies • The adopters mode a letter of intention	• The adopters of the system established their eProcurement team • The team attended an eProcurement management training • The vendors attended a training on how to use the system	• The eProcurement officers/administrators attended technical trainings and annual meetings held by LKPP
Links with convention of practice	• PD No. 80/2003 • PD No.54/2010 • LKPP issued a set of regulations concerning various procurement aspects, such as bidding documents standardization and procurement training accreditation	• The eProcurement practice was build upon the existing public procurement and other (such as auditing) practice • The new practice demanded improvement in other practices, such as procurement planning • LKPP held two national meetings annually involving officers from all the agencies • LKPP invited other governmental agencies to collaborate	• The system was built by considering the existing public procurement practice and similar systems in other countries as the inputs • The system adopted the data encryption method developed by National Crypto Agency (*Lembaga Sandi Negara* [LSN]) and the auditing mechanism developed by Finance and Development Supervisory Agency (*Badan Pengawasan Keuangan dan Pembangunan* [BPKP])

Embodiment of standards	• PD No. 80/2003 • PD No. 54/2010 • PI No. 11/2011 • LKPP issued a set of regulations concerning various standards	• The officers should be certified by LKPP • The eProcurement processes followed the standards set in the national regulations	• The system was equipped with a standard auditing and data encryption mechanism • The configuration and installation of IT infrastructure to support the eProcurement system followed the standards set by LKPP
Built on an installed base	• PD No. 80/2003 • PD No. 54/2010 • Local regulations were enacted to provide a technical guidance	• The eProcurement practice was built upon the existing procurement practice • The existing certified public procurement officers were involved in this initiative	• The system was built upon the existing IT assets and capabilities
Becomes visible upon breakdown		• LKPP through their permanent in-house IT team, whenever possible, provides immediate response for emergency trouble reported by the agencies • LKPP holds a regular workshop to review the incoming "trouble tickets" and makes a priority list for the system improvement	• The vendors report troubles to the agencies • The agencies report troubles directly to LKPP through the trouble ticketing system • LKPP and the agencies discuss the general technical problems in the national meeting • The in-house IT team continuously improves the systems based on the inputs obtained
Fixed in modular increments	• National regulations guided the implementation of the system gradually, from PD No. 80/2003, through PD No. 54/2010, to PI No. 11/2011 • Local regulations were enacted to guide the incremental adoption of the system	• Incremental adoption of the system by developing a selection criteria of work packages at local level and later by considering a target set by the PI No. 11/2011	• The system has gone through several versions with improved features in the newer versions

Notes: [a]Presidential Decree No. 80/2003 concerns the public procurement
[b]Presidential Degree No. 106/2007 concerns the establishment of LKPP
[c]Presidential Decree No. 54/2010 concerns the adoption of the eProcurement system
[d]Presidential Instruction No. 11/2011 partly concerns the target of eProcurement implementation both at national and local levels

6 Discussion

The findings are discussed in the light of the research questions posed at the outset: how can the sustainability of the eProcurement system be understood in the context of Indonesia? As presented above (see Table 2), the emergence of legal factor in the discussion of II is new. Previous studies on II [e.g., 8, 15, 16] did not study this factor possibly because they did not deal with a government II that is mainly developed and used by the government agencies. Thus, it is clear here that the nontechnical factors of the government II should include the legal factor, for at least two reasons. First, the regulations can force the government agencies to use a government II, and hence it will foster the adoption of II. Second, enacting the regulations that have a coercive power becomes more important in the context of a highly volatile government, like Indonesia, where decisions made by one incumbent may be overturned by the next and can lead to projects being derailed or blocked [21].

The intensive links between the three factors provide evidence that in order to ensure or to improve the sustainability of systems, we should consider both technical and nontechnical factors (in this context, managerial and legal factors). A change in a factor leads to changes in other factors. Here is an example. Although many government agencies have adopted the system, during the seventh national gathering of eProcurement agencies in Bali (November 2011), the Head of LKPP stated in a speech:

> I propose that we should interpret 'some' tenders [in PD No. 54/2010] as 'all' tenders. Let us start campaigning that 'all' tenders in 2012 should use the eProcurement system, not 'some.'

In fact, the PD only stated that starting from 2012 "some" tenders should use the system. At that time, many participants, especially those who were under external pressure from their local governments, seemed to receive moral support. However, this was only an interpretation—or to be more precise, an expectation—and it did not have any coercive power. Only after a clear target was set by the most recent PI No. 11/2011 did the use of the eProcurement system gain a new momentum and a new legitimacy. This is an example of why legal factors should keep up with the pace of changes in technical and managerial factors.

These factors should go hand in hand to ensure the sustainability of the system. Failing to keep pace with the changes may lead to other problems [12]. I introduced a triple-helix model of a government II (Fig. 1) that can explain this situation. In the Fig. 1, for an illustrative purpose, only some critical events and states are shown in a timeline.

In addition, the dimensions of II provide a framework to understand the sustainability of the system in an Indonesian context. The presented findings thus indicate that one way to ensure the sustainability of the system is by viewing it as an II and incorporating all the dimensions of II into it.

Some practical suggestions can be provided here based on the aforementioned findings. From a legal viewpoint, although regulations provide a coercion power, taking specificity of the context into account could smooth their enactment. In a context with huge disparities in many aspects (such quality of IT infrastructure,

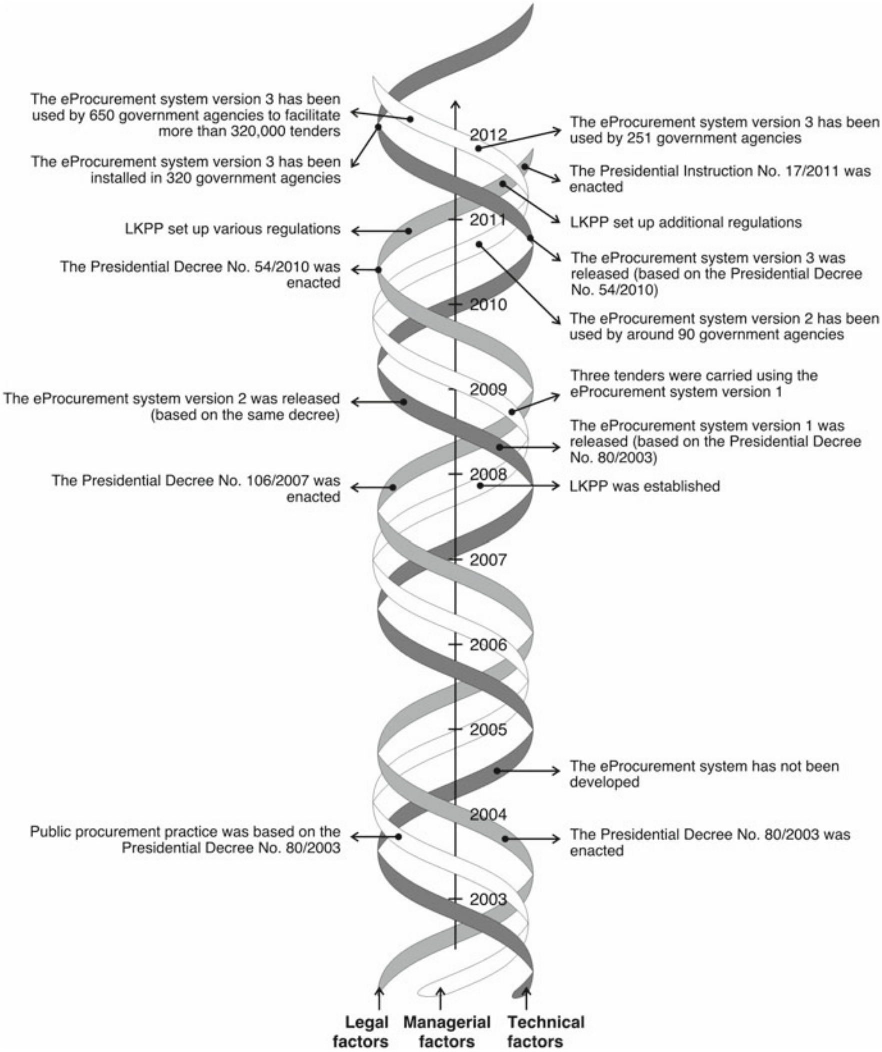


Fig. 1 A triple-helix model of a government II. Note: only some critical events and states are shown

readiness of the local governments) like Indonesia, the regulations made should address these issues. For example, incremental adoption of the eProcurement system was "legalized" by the PD No. 54/2010. Subsequently, the PI No. 17/2011 also set different targets for national and local government agencies, which obviously have different levels of readiness.

From a managerial standpoint, involvement of the community of practices, including relevant governmental agencies from various levels, in the development and improvement of II can ensure its sustainability. By doing so, LKPP can be attuned to real practices and to conventions of the community of practices. For example, from the perspective of IS development, the national gathering held by LKPP twice a year can be considered as a joint-application development (JAD) workshop [22]. JAD is believed to be an effective method of eliciting tacit knowledge from users. This is not a common practice in the Indonesian public sector. This mechanism will in turn address end users' requirements and make the diffusion even faster due to a high acceptability and hence will ensure the system sustainability.

Further, the involvement of the community of practices can also serve as a way to create a collective awareness and cultivate legitimacy. The legitimacy in this context is built by obtaining endorsement from other governmental agencies of the same or lower levels and authorization from the upper authority [23]. The support given by LSN and BPKP during the development of the system was an example of endorsement, while enactment of the new PD was an authorization. Again, this is not a common practice in the Indonesian public sector (or public sectors elsewhere), which is often influenced by a strong "sectoral ego" that hinders collaborative action between the agencies. Braa et al. [24] found that creating networks of action involving different parties was a determining factor in ensuring the sustainability of an IS. The Director of eProcurement in LKPP agreed with this assertion. In a personal communication, he stated:

> Sustainability as a movement only happens with a massive participation. This participation needs an approach that provides a room for all the parties to play their role. There is no dominating role in this kind of movement, although it needs a 'bond', to prevent it from an undirected movement. This is perhaps the factor that has been driving the movement so far.

From a technical perspective, a continuous incremental development of II was proven to be a successful strategy to make it sustainable. This was made possible mainly by the establishment of a permanent in-house IT team that was attuned to the real practice and responsive to the system breakdowns. This is very important, especially for developing and maintaining a system, which is intended to have a long life and to be used across governmental agencies and levels. In the Indonesian context, this strategy is not commonly adopted by government agencies. Often, they outsourced the system development with a short contract. Thus, the government agency became a hostage of the developer. The agency was unresponsive to changes, which led to costly maintenance.

7 Conclusion

The paper has shown how II dimensions can be used to understand the sustainability of the eProcurement initiative in Indonesia. These dimensions were grouped into three factors (legal, managerial, and technical factors), indicating that to create a sustainable II, these factors should be taken into consideration in an orchestrated fashion. These factors made a triple-helix model of a government II.

This study offers three contributions. Firstly, it offers a new perspective from which to understand the sustainability of an eGovernment initiative (eProcurement here), by applying II dimensions. This enhances our understanding of sustainability of systems by giving a framework in addition to those available in the literature [12, 24].

Secondly, it extends or reinterprets the concept of II in the context of eGovernment studies or more generally in the context of the public sector, by introducing three factors that go hand in hand during the development and implementation of the eProcurement system. The proposed triple-helix model of a government II is a theoretical contribution.

Thirdly, it offers a set of practical suggestions that may inform both eGovernment researchers and practitioners. Then, we may expect that what LKPP has done in developing the eProcurement system can be considered as a guideline that can be adapted in other developments of national ISs in the public sector. Such initiatives should thus be more II oriented with a permanent and long-life initiative in mind and not only project-oriented with a short life.

This study suffers from some limitations. Firstly, as an II is often context-specific, generally this single-case study may not capture all the variability that may emerge during the process of implementation of an II. A similar assessment of other national IIs, both successful and unsuccessful, could be conducted. This is an interesting future research avenue.

Secondly, the sustainability of the II under study was judged based on its short history (i.e., 4 years). A question about the time frame or the duration emerges here. For how long must an II be in place before it can be considered sustainable? Hence, assessment of the same eProcurement system in the future could be another interesting research domain to explore.

Thirdly, although the eProcurement system in Indonesia is considered sustainable, the study has not assessed its impact and the sustainability of the impact. It can be argued that a sustainable IS is a prerequisite for sustainable impacts. In context of this study, it can be understood that without a sustainable eProcurement system, it is difficult to find ways to sustain the impacts of eProcurement (such as enhanced transparency, better access for nonlocal bidders, better access for small and medium-sized enterprises, and corruption avoidance [10]). However, the question remains whether there is indeed a link between a sustainable IS and its sustainable impact. Some impacts can be easily measured, such as budget savings, but other impacts (e.g., public procurement transparency, SMEs empowerment) are intangible and difficult to measure. Thus, creativity is required in order to measure the impact. Taken together, this is another future research avenue.

8 Appendix: Examples of Questions in the Interview

1. What were the main objectives of this initiative?
2. How were the decisions made?
3. How have the objectives changed over time?
4. Who were involved in the implementation?
5. How was the communication between the involved parties made during the implementation?

6. What were the general reactions from the involved parties to this initiative implementation?
7. How the reactions affect the progress and outcome of the initiative?
8. How has the attitude of the involved parties towards the initiative changed over time?

References

1. Sein MK (2011) The "I" between G and C: e-Government intermediaries in developing countries. Electron J Inform Syst Dev Countries 48:1–14
2. Walsham G, Sahay S (2006) Research on information systems in developing countries: current landscape and future prospects. Inform Technol Dev 12:17–24
3. Braa J, Hanseth O, Heywood A, Mohammed W, Shaw V (2007) Developing health information systems in developing countries: the flexible standards strategy. MIS Quarterly 31:381–402
4. Heeks R (2002) Information systems and developing countries: failure, success and local improvisations. Inform Soc 18:101–112
5. Heeks R (2002) E-government in Africa: promise and practice. Inform Polity 7:97–114
6. (2012) Lembaga Kebijakan Barang/Jasa Pemerintah: e-Procurement implementation progress 2008–2012
7. Orlikowski WJ, Iacono CS (2001) Research commentary: desperately seeking the "IT" in IT research—a call to theorizing the IT artifact. Inform Syst Res 12:121–134
8. Bygstad B (2008) Information infrastructures as organization: a critical realist view. The 29th international conference of information systems (ICIS), Paris, France
9. Anonymous (2009) Corruption ingrained in procurement processes. The Jakarta Post
10. Transparency International (2008) Handbook for curbing corruption in public procurement. Transparency International, Berlin
11. Maruster L, Faber NR, Peters K (2008) Sustainable information systems: a knowledge perspective. J Syst Inform Technol 10:218–231
12. Jacucci E, Shaw V, Braa J (2006) Standardization of health information systems in South Africa: the challenge of local sustainability. Inform Technol Dev 12:225–239
13. Kumar R, Best ML (2006) Impact and sustainability of e-government services in developing countries: lessons learned from Tamil Nadu, India. Inform Soc 22:1–12
14. Furuholt B (2009) Bridging the digital divide: sustainable supply and demand of internet access in developing countries. PhD thesis. Aalborg University, Denmark
15. Star SL, Ruhleder K (1996) Steps toward an ecology of infrastructure: design and access for large information spaces. Inform Syst Res 7:111–134
16. Hanseth O, Lyytinen K (2010) Design theory for dynamic complexity in information infrastructures: the case of building internet. J Inform Technol 25:1–19
17. Star SL (2010) This is not a boundary object: reflections on the origin of a concept. Sci Technol Hum Values 35:601–617
18. Benbasat I, Goldstein DK, Mead M (1987) The case research strategy in studies of information systems. MIS Quarterly 11:369–386
19. Hanseth O (2002) From systems and tools to networks and infrastructures. Toward a theory of ICT solutions and its design methodology implications
20. Rolland KH (2000) Challenging the installed base: deploying a large-scale IS in a global organization. In: Hansen HR, Bichler M, Mahrer H (eds) Proceedings of the 8th European conference of information systems, pp 583–590
21. Furuholt B, Wahid F (2008) E-government challenges and the role of political leadership in Indonesia: the case of Sragen. In: Proceedings of the 41st Hawaii International Conference on System Sciences (HICSS) 2008

22. Friedrich WR, van der Poll JA (2007) Towards a methodology to elicit tacit domain knowledge from users. Interdiscipl J Inform Knowl Manag 2:180–193
23. Walker HA (2004) Beyond power and domination: legitimacy and formal organizations. In: Johnson C (ed) Research in the sociology of organizations, vol 22. Elsevier, Oxford, UK, pp 239–271
24. Braa J, Monteiro E, Sahay S (2004) Networks of action: sustainable health information systems across developing countries. MIS Quarterly 28:337–362

Usability Work in Agile Systems Development Practice: A Systematic Review

Adeola Wale-Kolade, Peter Axel Nielsen, and Tero Päivärinta

Abstract In this chapter we present the results of a systematic literature review of the recommendations in the existing research literature on usability work in agile systems development. The review contributes by summarizing the literature in light of seven claims about how to integrate usability work into an agile development project. By analyzing the claims we show how the previous literature provides grounds, warrants, backing, rebuttal, and qualification with regard to each of them. From this comprehensive overview of the literature we then discuss a research agenda with a particular focus on how situational factors for the claims must be researched and how this must encompass identified rebuttals and qualifications.

1 Introduction

Agile methods are increasingly used for information systems development (ISD). This trend is stimulated by today's dynamic business environment [18] and the growing preference among practitioners for these methods due to their perceived effectiveness [65]. However, agile methods have been challenged with regard to

A. Wale-Kolade (✉)
Department of Information Systems, University of Agder, Agder, Norway
e-mail: adeola.y.wale-kolade@uia.no

P.A. Nielsen
Research Centre for Socio-Interactive Design, Aalborg University, Aalborg, Denmark

Department of Information Systems, University of Agder, Agder, Norway
e-mail: pan@cs.aau.dk

T. Päivärinta
Department of Computer Science, Electrical and Space Engineering Computer and Systems Science, Luleå University of Technology, Luleå, Sweden
e-mail: tero.paivarinta@ltu.se

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_44,

their lack of focus on system usability and usability work [41]. This contradiction has led to sustained research interest in the phenomena [58].

Recent criticisms on user-centered design in agile development include contentions about the methodological claims being grounded on anecdotal evidence [38, 58]. Further, researchers have largely ignored situational factors and their influences on methods in use in positing solutions to the phenomena [26]. In tandem, differences between these theorized solutions have been found to be mostly superficial, as an examination of their structural foundation reveals them to be focused on common practices and artifacts [58].

If the suggested usability methods are assumed to consist of similar practices and artifacts, they could be expected to comprise similar recommendations on how usability efforts should be performed within agile development projects. Taking this proposition into account and viewing this against the backdrop of the highlighted criticisms of agile usability research, two questions emerge which motivate our review of the agile usability literature:

- What are the recommendations on how usability work should be executed within agile contexts?
- Are there situational factors that influence these, and what is the nature of such influences?

The rest of the chapter is structured as follows: In Sect. 2, we discuss our conceptual background, Sect. 3 details the review process, and Sects. 4–6 detail our analysis, discussion, and conclusion, respectively.

2 Conceptual Background

We base our review on the understanding of "usability" within agile contexts as going beyond being concerned only with the attributes of a system to also encompassing the work efforts through which such a system materializes [41]. Our stance is based on the logic that the three perspectives of usability, that is, the product, process, and experiential perspectives [5, 33, 57], all find commonality in the means through which they materialize, which is through an exertion of work. This encompasses the following:

1. How guidelines or design heuristics are applied as in the case of the product perspective [28]?
2. How processes are followed as in the process perspective [30]?
3. How user perceptions are measured as in the case of the experiential perspective [9]?

It is for this reason we have elected to use the term *usability work* as our working concept, which represents all the work practices (and practice abstractions) that relate to the abovementioned perspectives, being conducted in software and systems development processes.

The issues surrounding agile methods and usability work stem from the assertion that agile methods hinder the performance of usability work [41]. This has been linked to the manner in which it has been implemented so far. An example is the process of acquiring just-in-time system requirements, which has been identified as hindering the in-depth exploration of user requirements [11, 15]. Such observations have resulted in a plethora of solutions existing in the form of guidelines and methodologies [26] and which form the main focus of our review.

3 Review Methodology

Silva da Silva et al. [58] recently reviewed a body of knowledge with a particular focus on the development of a theoretical construct for the integration of user-centered design (UCD) and agile methods. Our review distinguishes itself from this work in three ways to seek answers to the research questions posed above. Firstly, we seek to identify the recommendations on the integration of usability work and agile methods. Secondly, we go beyond this to identify also situational factors influencing the integration and examining the nature of such influences. Thirdly, we have broadened the scope to usability work instead of UCD, for reasons highlighted in the preceding section.

Our focus on usability work influenced how we selected papers from the body of knowledge used in the Silva da Silva et al. review and the subsequent literature search we carried out in Google Scholar and Scopus. All in all, 38 papers in Silva [58] dealt with the issue of how usability work should be conducted in connection to an agile systems development approach. We have updated this list and added 11 papers to the 38 selected from the Silva da Silva et al. review. The 49 papers we have reviewed are listed alphabetically in Fig. 1. The search terms include usability, UCD, agile, Scrum, XP, user interface, and human computer interaction, which are effectively the same search terms used in Silva da Silva et al. [58]. Search on Scopus from 2011 yielded three additional papers. Further eight papers were added by searching the reference lists of the already included papers.

The analytical lens employed to integrate the literature is the Toulmin model [62]. Table 1 provides an overview of its concepts and how these were particularized to our data. The model provides a concise framework to assess the argumentative structure of the recommendations identified in the reviewed literature. Further, through the observed qualifications and rebuttals, we could identify pertinent situational factors and explicate the nature of their influences. We applied content analysis in a stepwise manner [36] to identify themes related to concepts of our selected model. This led to the identification of common recommendations and for each recommendation their grounds, rebuttals, and qualifiers. Our findings are presented in the next section.

Authors

- 1. Adikari et al., 2009
- 2. Albisetti, 2010
- 3. Ambler 2008
- 4. Armitage, 2004.
- 5. Barksdale and McCrickard, 2010
- 6. Belchev and Baker, 2009
- 7. Benigni et al., 2010
- 8. Beyer et al., 2004
- 9. Blomkvist, 2005
- 10. Bonacin et al., 2009
- 11. Broschinsky and Baker, 2008
- 12. Budwig et al.,2009
- 13. Chamberlain et al., 2006
- 14. Cho, 2009
- 15. Constantine and Lockwood, 2002
- 16. Detweiler,2007
- 17. Düchting et al., 2007
- 18. Esteves and Andrade, 2011

Authors

- 19. Evnin and Pries, 2008
- 20. Federoff and Courage, 2009
- 21. Ferreira et al., 2007
- 22. Ferreira et al., 2011
- 23. Fox et al., 2008
- 24. Gonçalves and Santos, 2011
- 25. Haikara , 2007
- 26. Hellmann et al., 2010
- 27. Hodgetts, 2005
- 28. Hosseini- Khayat et al., 2010
- 29. Humayoun et al., 2011
- 30. Hussain et al., 2009b
- 31. Illmensee and Muff, 2009
- 32. Kane, 2003
- 33. Kollmann et al., 2009
- 34. Krohn et al., 2009
- 35. Lee et al., 2009

Authors

- 36. Losada et al., 2011
- 37. Memmel at al., 2007
- 38. Meszaros and Aston, 2006
- 39. Miller, 2005
- 40. Najafi and Toyoshiba, 2008
- 41. Obendorf and Finck, 2008
- 42. Paelke and Nebe, 2009
- 43. Patton, 2002
- 44. Peixoto and da Silva, 2009
- 45. Singh, 2008
- 46. Sy, 2007
- 47. Ungar,2008
- 48. Williams and Ferguson, 2007
- 49. Winter et al., 2011

Fig. 1 Data sources

Table 1 The analytical lens based on Toulmin et al. [62]

Elements of an argument	As described by Toulmin et al. [62]	In this review
Claim (C)	Assertions put forward publicly for general acceptance	A recommendation for a usability work practice in agile systems development as advocated by the authors
Grounds (G)	Specific facts relied on to support a given claim	Assertions used to support the recommendation
Warrant (W)	The chain of reasoning that connects the grounds to the claim	Theories, principles, assumptions being appealed to by the authors
Backing (B)	Generalizations making explicit the body of experience relied on to support the warrant	Source(s) of the warrants (from empirical evidence, body of knowledge)
Qualification (Q)	Statements which limit the strength of the argument or which propose the conditions under which the warrant is true	Account of situational factors which qualify the scope of recommendations
Rebuttal (R)	Extraordinary or exceptional circumstances that might undermine the force of the supporting arguments	Account of situational factors which refute the recommendations made

4 Analysis

Our analysis identified seven claims on the integration of agile and usability work. Following the concept-centric approach advocated by Webster and Watson [64], these claims are categorized based on what aspect of usability work they describe.

4.1 Work Execution

Claim 1: *Conduct some upfront design activities prior to project start* (**6**, **8**, **9**, **11**, **13**, **16**, **17**, **19**, **23**, **25**, **27**, **30**, **31**, **34**, **40**, **41**, **45**, **46**, **48**).

Grounds: (1) Facilitates insight into use context, leading to increased understanding during the project life cycle (**27**).

Warrants: User's goals, tasks, and needs should all guide the development from the very beginning (**9**).

Backing: Principles of UCD (**9**, **15**).

Qualification: (1) Feasible when there is an absence of time and budget constraints (**18**, **23**, **25**, **27**, **29**, **31**, **33**). (2) Appropriate when it is certain it would reduce project risks (**21**). (3) Would require that designers have high physical and mental resilience in order to combat the fatigue associated with this effort (**31**).

Rebuttal: (1) End users are typically not able to contribute novel system design ideas (**15**, **42**). (2) Lengthy prerelease activities do not guarantee a usable system (**9**, **42**).

Claim 2: *Design low-fi prototypes as the basis for developing the system* (**1**, **8**, **9**, **10**, **11**, **13**, **16**, **17**, **23**, **24**, **26**, **38**, **41**, **42**, **45**, **47**, **48**).

Grounds: (1) Cheap, expedites exploration and evaluation, increases user understanding of the user interface and usage (**9**, **14**).

Warrants: Throughout the development life cycle, prototypes should be used to visualize and evaluate ideas and design solutions in cooperation with the end users (**9**).

Backing: Principles of UCD (**9**, **15**).

Qualification: (1) Suitable when used at the start of the project, especially in situations where there are time and budget constraints (**18**, **21**, **29**). (2) Suitable when the existing work structure is one which allows for seamless transmission of information between developers and designers, thus minimizing the lag between feedback and implementation (**13**).

Rebuttal: (1) Abstract and too unstructured to guide system design (**37**). (2) Limited capabilities for the user to evaluate (**21**, **39**, **48**). (3) Often results in impracticable designs (**21**, **22**).

Claim 3: *Perform testing in between iterations* (**3**, **8**, **9**, **14**, **23**, **26**, **28**, **33**, **39**, **40**, **42**, **45**) *and with end users* (**8**, **9**, **13**, **16**, **17**, **20**, **33**, **35**, **36**, **38**, **39**, **40**, **42**, **45**).

Grounds: (1) Enables refinements of the system being developed, reduces rework, and increases buy-in (**26**, **45**). (2) Only real users can articulate their needs (**17**).

Warrants: None

Backing: None

Qualification: (1) Qualified by the ease of assessing end users (**32**). (2) Depends on the level of influence of time and budget concerns (**32**). (3) Depends on the organizational profile (**20**, **49**). (4) Not suitable for systems with volatile requirements as recommendations made in between iterations may end up being obsolete (**30**).

Rebuttal: (1) Higher tendency for usability defects to be introduced in later iterations (**32**). (2) Cannot give the user a true picture of the system, as code generated during sprints is often too unstable to conduct tests on (**16**). (3) Does not guarantee user representativeness since users become increasingly familiar with the prototypes being tested (**30**).

Claim 4: *Designers and developers each work in parallel* (**8**, **11**, **12**, **15**, **16**, **18**, **20**, **23**, **24**, **31**, **33**, **35**, **36**, **37**, **39**, **40**, **46**, **47**).

Grounds: (1) Would enable designers perform their tasks ahead of the developers (**46**). (2) Eliminates the chances of developing designs that are not used and more cost effective, enables designers get more timely feedback, and maximizes coding time (**39**).

Warrants: None

Backing: None

Qualification: (1) Requires high headcount of designers (**3**, **11**, **12**, **13**). (2) More suitable for novel or large projects as opposed to rework on existing systems (**20**).

Rebuttal: (1) In reality designer work is driven by agile developers' work and not vice versa (**21**). (2) Prone to design drift (**13**, **21**, **35**). (3) May undermine consistency of the interface (**16**). (4) Limits input from designer to a "just one pass role" (**4**).

4.2 Work Organization

Claim 5: *Usability designers should be a part of the development project* (**1**, **5**, **8**, **9**, **10**, **11**, **12**, **13**, **14**, **15**, **16**, **17**, **18**, **19**, **20**, **21**, **23**, **27**, **30**, **31**, **33**, **35**, **37**, **39**, **40**, **41**, **42**, **44**, **45**, **47**, **48**).

Grounds: (1) Having a usability domain expert ensures that usability concerns are always at the fore (**17**).

Warrants: Usability experts should be involved early on and continuously throughout the development life cycle (**9**).

Backing: Principles of UCD (**9**).

Qualification: (1) Depends on whether this role is present in the organization (**3**, **11**, **13**).

Rebuttal: (1) Developers with some knowledge of usability work are also able to function in this capacity (**2**, **38**, **43**).

Claim 6: *Usability designers should be fully integrated into the development team* (**3**, **9**, **11**, **14**, **16**, **18**, **20**, **27**, **35**, **37**, **39**, **40**, **45**).

Grounds: (1) Ensures that project team focus and self-organizing nature of agile teams is maintained (**27**). (2) Necessary to share the results of user research and testing and to clarify any misinterpretations of designs (**40**). (3) Ensures that usability concerns are always at the fore (**17**).

Warrants: None
Backing: None
Qualification: (1) Depends on whether this role is present in the organization (**3**, **11**, **13**). (2) Depends on whether management reveres the cross-specialization skills the designer would need to have to function in such a context (**3**).

Rebuttal: (1) Undermines the quality of work from each of the disciplines (**21**). (2) Designers may lose their objectivity and user focus as they begin to identify very closely with the goals and needs of their respective project teams (**16**).

Claim 7: *End users or their proxies should be involved in the project life cycle* (**8**, **9**, **10**, **13**, **17**, **38**, **48**).
Grounds: Enables end users express their opinions about activities, practices, tasks, and usage context (**10**).
Warrants: Representative users should actively participate, early on and continually throughout the entire development process (**9**).
Backing: Principles of UCD (**9**).
Qualification: (1) Qualified by the accessibility of end users (**11**, **39**). (2) Depends on the organizational profile (**11**, **39**).
Rebuttal: (1) Unable to cater for the different mental model of users regarding usability (**32**). (2) Characterized by diminished user representativeness with increasing user involvement (**8**, **32**).

4.3 Summary of Findings

The identified claims and their qualifications have been re-categorized based on the framework by Clarke and O'Connor [17] of situational factors which influence the software development process. This has been done to relate our findings to existing research on these situational factors. For an overview of this, see Table 2. We notice that none of the qualifications we have highlighted addresses the situational factor "technology." The existence of this gap should be seen as an impetus for agile usability researchers to examine how considerations related to the technology employed affect the claims we have identified and in effect how usability work is executed within agile contexts.

As a progression from Table 2, Fig. 2 represents a high-level abstraction of the relationships between the seven identified claims, their rebuttals, and situational factors.

5 Discussion

The agile usability literature has been criticized for providing overtly rationalized solutions to problems and concerns and for presenting an impression of balance and harmony [26]. Previous reviews [58, 60] on a subset of the literature we review here

Table 2 Factors influencing the software development process mapped to the claims' qualifications (CnQm denotes the mth qualification of the nth claim)

Situational factor	Description and considerations	Mapping of qualification
Business	Strategic and tactical business considerations—time constraints, project drivers	C1Q1, C2Q1, C3Q2
Personnel	Constitution and characteristics of personnel—team size, team capability	C1Q3, C4Q1
Requirements	Characteristics of requirements—degree of risk, requirement feasibility	C1Q2
Operation	Operational consideration and constraints—number of users outside the organization, organizational policies	C2Q2, C3Q1, C7Q1
Organization	Organization profile—structure, size	C3Q3, C5Q1, C6Q1, C7Q2
Application	Characteristics of application under development type, complexity	C3Q2, C4Q2
Management	Management constitution and characteristics—governance structure, project planning capability	C6Q2
Technology	Profile of technology being used for the development process—knowledge of technology, familiarity	None

have also followed this pattern by developing integrated methods and frameworks based on observed commonalities. Our work deviates from this path by summarizing seven claims made about usability work practices and examining the hitherto reported validity of these claims with the Toulmin model [62].

By particularizing the model's concepts in our examination of these claims, we have been able to identify key situational factors which affect the validity of these claims by rebutting or qualifying them. Thus, our findings extend the ongoing discussions within agile usability research beyond the present focus on the lack of empirical studies [58], and calls that have been made for the exploration of the relationship between situational factors and methods in use [26], by providing cogent reasons as to why these observations are justified. Further as some of the identified claims are based on knowledge derived from the wider usability literature, their weak grounding underscores the need for a reexamination of the theoretical form of usability work. It has been asserted that the ideas of practice characterized in the theoretical form are obsolete [51, 56]; whether this is responsible for the weak grounding observed is another issue that should be investigated.

The following three propositions are based on our theoretical framework in Fig. 2 and further expound our discussions on the nature of influences of situational factors on the usability work claims identified.

- *Situational factors exist as modifiable factors which qualify these usability work claims and non-modifiable factors which rebut these usability work claims*— In Fig. 2, the factors identified as qualifying the usability work claims are seen to

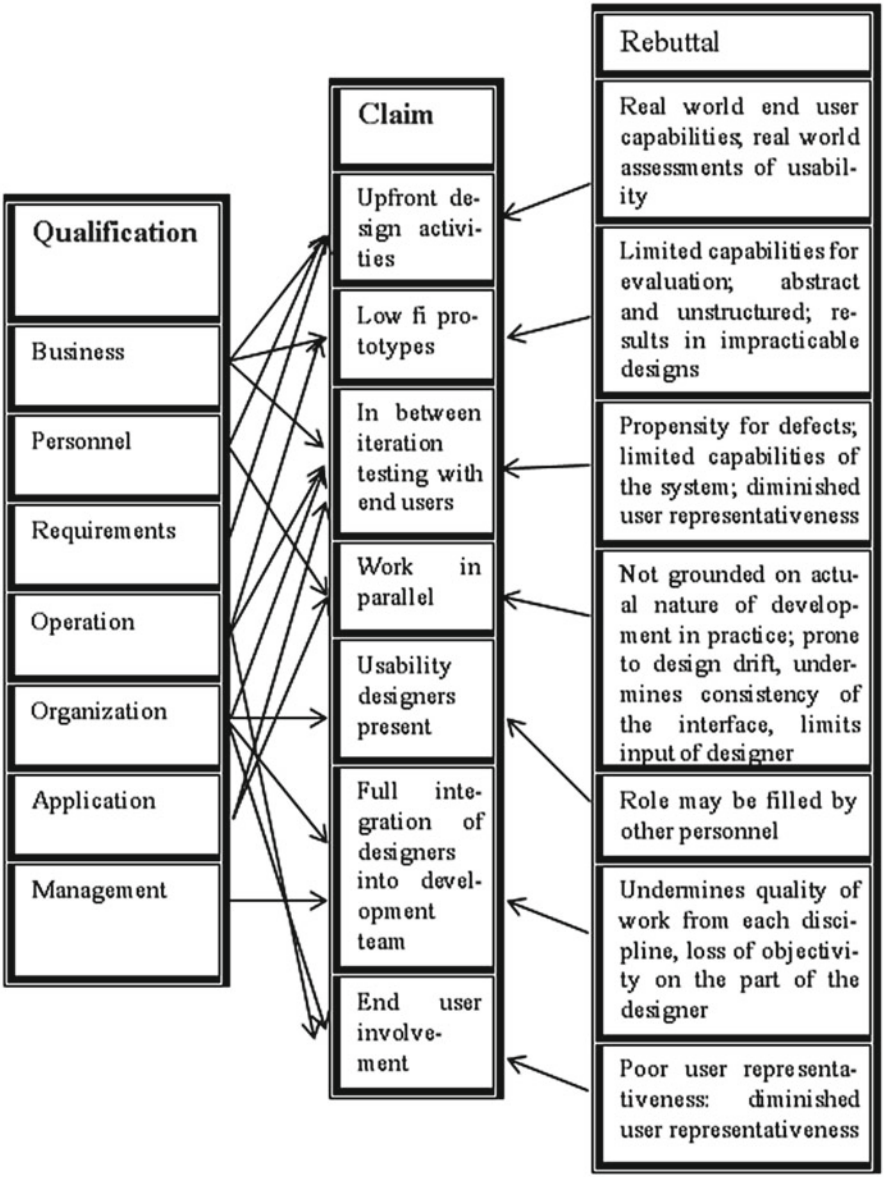


Fig. 2 Factors, claims, and rebuttals

be factors that can be controlled by the organization, while the rebuttals highlight factors which are beyond the control of the organization.

- *Modifiable situational factors influence the efficiency of the recommendations*— These qualifications highlight the conditions under which these usability work

claims might be made feasible and thus determine how efficient the process would be.
- *Non-modifiable factors influence the effectiveness of these recommendations*—They do this by highlighting considerations which would prevent the projected outcomes of these recommendations from emerging.

Relating these considerations to the realm of usability work in agile development, our framework in Fig. 2 in this regard forms a theoretically solid basis for further empirical examination of the phenomena, as it highlights situational factors that need to be taken into account during such examinations. Further, it reveals associations between situational factors and claims that should be empirically assessed and tested. This exercise is significant as it would enable development of usability work strategies that are aligned to the situational factors which characterize the particular organizations being investigated.

Our findings also extend discussions within the ISD literature on the situational factors influencing the development processes [17], as we elucidate more on the nature of these influences. Further we introduce another dimension along which these influences might be examined beyond the present focus on modifiable factors that can be controlled by the organization.

A further observation is that despite the call by Kane [42] for more work on the development of heuristic and style guidelines for interaction design for use in agile contexts, these issues have not been fully taken into account. We only observed one paper which came close to examining this issue [55]. One of the papers examined explicitly stated that they did not perceive that it was the role of usability work to specify system architecture [59].

In our current version, we searched the literature systematically with focus on articles published in 2011. We also focused on related articles that had not been covered by Silva da Silva et al. [58] but had been noted in the papers we reviewed. As any systematic search suffers from the search procedure applied however it is designed, a possible outcome of the approach we employed is that key articles may have been overlooked.

An additional limitation of our analysis is that the claims we have identified are formulated in a non-operational manner. This is in part due to our attempting to describe these claims in a manner that concisely captures their essence and reflects the Toulmin's model depiction of a claim. It is also due to space restrictions. Moreover our overarching goal in identifying these claims was not to provide a prescriptive approach as to how agile and usability work should be integrated, rather it was to provide a platform for the analysis of these claims. We have done this by highlighting the various situational factors which could qualify or rebut these claims. However though our actions based on the preceding arguments may be deemed justified, we acknowledge the difficulties it poses to the operationalization of our findings, hence the need for further research and refinement.

6 Conclusion

In this chapter we have provided insight into recommendations that have been made in the literature regarding how agile and usability work should be organized. Further we have employed the Toulmin model [62] to analyze the extent to which these recommendations have been validated in the systems and software development literature. The absence of a genuine practice perspective is demonstrated in the existence of rebuttals and the absence of warrants, which reduce the strength of these recommendations. This points to the need for a more situated approach to research on usability work in agile development and calls for a more informed and reflexive approach to an examination of these issues. Further the identified rebuttals and qualifications accentuate that there are other considerations far beyond theorized views of usability work which also need to be examined. We have in this regard provided a framework in Fig. 2 that highlights situational factors whose influences need to be taken into account during such examinations. Our paper is however characterized by two limitations. Firstly the subset of literature analyzed may not be entirely exhaustive. Secondly our claims have been presented in a non-operational manner. These are issues we aim to resolve in future research.

References

1. Adikari S, McDonald C et al (2009) Little design up-front: a design science approach to integrating usability into agile requirements engineering. In: Jacko J (ed) Human-computer interaction. New trends, vol 5610. Springer, Berlin/Heidelberg, pp 549–558
2. Albisetti M (2010) Launchpad's quest for a better and agile user interface. Agile processes in software engineering and extreme programming, pp 244–250
3. Ambler SW (2008) Tailoring usability into agile software development projects. In: Law EL-C, Hvannberg ET, Cockton G (eds) Maturing usability. Springer, London, pp 75–95
4. Armitage J (2004) Are agile methods good for design? Interactions 11(1):14–23
5. Balázs Polgár P, Biró M (2011) The usability approach in software process improvement systems, software and service process improvement. In: O'Connor RV, Pries-Heje J, Messnarz R (eds) Springer, Berlin/Heidelberg, vol 172. pp 133–142
6. Barksdale JT, McCrickard DS (2010) Concept mapping in agile usability: a case study. In: Proceedings of the 28th of the international conference extended abstracts on human factors in computing systems. ACM, Atlanta, Georgia, USA, pp 4691–4694
7. Belchev B, Baker P (2009) Improving obama campaign software: learning from users. IEEE
8. Benigni G, Gervasi O et al (2010) USABAGILE_Web: a web agile usability approach for web site design computational science and its applications—ICCSA 2010. In: Taniar D, Gervasi O, Murgante B, Pardede E, Apduhan B (eds) vol 6017. Springer, Berlin/Heidelberg, pp 422–431
9. Bevan N (2009) What is the difference between the purpose of usability and user experience evaluation methods. In: Proceedings of the workshop UXEM'09 (INTERACT'09), Uppsala, Sweden
10. Beyer H, Holtzblatt K et al (2004) An agile customer-centered method: rapid contextual design. In: Zannier C, Erdogmus H, Lindstrom L (eds) Extreme programming and agile methods—XP/Agile Universe 2004, vol 3134. Springer, Berlin/Heidelberg, pp 527–554

11. Blomkvist S (2005) Towards a model for bridging agile development and user-centered design. In: Seffah A, Gulliksen J, Desmarais MC (eds) Human-centered software engineering—integrating usability in the software development lifecycle, vol 8. Springer, Netherlands, pp 219–244
12. Bonacin R, Baranauskas MCC et al (2009) An agile process model for inclusive software development. In: Filipe J, Cordeiro J (eds) Enterprise information systems, vol 24. Springer, Berlin/Heidelberg, pp 807–818
13. Broschinsky D, Baker L (2008) Using persona with XP at LANDesk software, an avocent company. Agile Conference, 2008. Agile '08. IEEE
14. Budwig M, Jeong S et al (2009) When user experience met agile: a case study. In: Proceedings of the 27th international conference extended abstracts on human factors in computing systems. ACM, Boston, MA, USA, pp 3075–3084
15. Chamberlain S, Sharp H et al (2006) Towards a framework for integrating agile development and user-centred design. In: Abrahamsson P, Marchesi M, Succi G (eds) Extreme programming and agile processes in software engineering, vol 4044. Springer, Berlin/Heidelberg, pp 143–153
16. Cho L (2009) Adopting an agile culture a user experience team's journey. Agile conference, 2009. AGILE '09. IEEE
17. Clarke P, O'Connor RV (2012) The situational factors that affect the software development process: towards a comprehensive reference framework. Inform Software Technol 54(5): 433–447
18. Conboy K (2009) Agility from first principles: reconstructing the concept of agility in information systems development. Inform Syst Res 20(3):329–354
19. Constantine LL, Lockwood LAD (2002) Usage-centered engineering for Web applications. IEEE Software 19(2):42–50
20. Detweiler M (2007) Managing UCD within agile projects. Interactions 14(3):40–42
21. Düchting M, Zimmermann D et al (2007) Incorporating user centered requirement engineering into agile software development. In: Jacko J (ed) Human-computer interaction. Interaction design and usability, vol 4550. Springer, Berlin/Heidelberg, pp 58–67
22. Esteves M, Andrade V (2011) Designing interaction concepts, managing customer expectation and mastering agile development in rich application product development. In: Jacko J (ed) Human-computer interaction design and development approaches, vol 6761. Springer, Berlin/Heidelberg, pp 54–62
23. Evnin J, Pries M (2008) Are you sure? Really? A contextual approach to agile user research, agile conference, 2008. Agile '08. IEEE
24. Federoff M, Courage C (2009) Successful user experience in an agile enterprise environment. In: Smith M, Salvendy G (eds) Human interface and the management of information. Designing information environments, vol 5617. Springer, Berlin/Heidelberg, pp 233–242
25. Ferreira J, Noble J et al (2007) Up-front interaction design in agile development. In: Concas G, Damiani E, Scotto M, Succi G (eds) Agile processes in software engineering and extreme programming, vol 4536. Springer, Berlin/Heidelberg, pp 9–16
26. Ferreira J, Sharp H et al (2011) User experience design and agile development: managing cooperation through articulation work. Software Pract Ex 41(9):963–974
27. Fox D, Sillito J et al (2008) Agile methods and user-centered design: how these two methodologies are being successfully integrated in industry. In: Proceedings of the Agile 2008, IEEE Computer Society, pp 63–72
28. Folmer E, Bosch J (2002) Architecting for usability: a survey. J Syst Software 70(1–2):61–78
29. Gonçalves J, Santos C (2011) POLVO-software for prototyping of low-fidelity interfaces in agile development. In: Human-computer interaction. Design and development approaches, pp 63–71
30. Gulliksen J, Göransson B et al (2003) Key principles for user-centred systems design. Behav Inform Technol 22(6):397–409
31. Haikara J (2007) Usability in Agile software development: extending the interaction design process with personas approach. In: Concas G, Damiani E, Scotto M, Succi G (eds) Agile

processes in software engineering and extreme programming, vol 4536. Springer, Berlin/Heidelberg, pp 153–156
32. Hellmann TD, Hosseini-Khayat A, Maurer F (2010, April) Supporting test-driven development of graphical user interfaces using agile interaction design. In: Software Testing, Verification, and Validation Workshops (ICSTW), 2010 Third International Conference on (pp. 444–447). IEEE
33. Hertzum M, Clemmensen T (2012) How do usability professionals construe usability? Int J Hum Comput Stud 70(1):26–42
34. Hodgetts P (2005) Experiences integrating sophisticated user experience design practices into agile processes, Agile conference, 2005. Agile '05. IEEE
35. Hosseini-Khayat A, Hellmann TD et al (2010) Distributed and automated usability testing of low-fidelity prototypes, Agile conference, 2010. Agile '10. IEEE
36. Hsieh H-F, Shannon SE (2005) Three approaches to qualitative content analysis. Qual Health Res 15(9):1277–1288
37. Humayoun S, Dubinsky Y et al (2011) A three-fold integration framework to incorporate user–centered design into agile software development. In: Kurosu M (ed) Human centered design, vol 6776. Springer, Berlin/Heidelberg, pp 55–64
38. Hussain Z, Slany W et al (2009) Investigating Agile user-centered design in practice: a grounded theory perspective. In: Holzinger A, Miesenberger K (eds) HCI and usability for e-inclusion, vol 5889. Springer, Berlin/Heidelberg, pp 279–289
39. Hussain Z, Milchrahm H et al (2009) Integration of extreme programming and user-centered design: lessons learned. In: Maurer F, Abrahamsson P, Marchesi M et al (eds) Agile processes in software engineering and extreme programming, vol 31. Springer, Berlin/Heidelberg, pp 174–179
40. Illmensee T, Muff A (2009) 5 users every friday: a case study in applied research. Agile conference, 2009. AGILE '09
41. Jokela T, Abrahamsson P (2004) Usability assessment of an extreme programming project: close co-operation with the customer does not equal to good usability. In: Bomarius F, Iida H (eds) Product focused software process improvement, vol 3009. Springer, Berlin/Heidelberg, pp 393–407
42. Kane D (2003) Finding a place for discount usability engineering in agile development: throwing down the gauntlet. Agile development conference, 2003. ADC 2003. Proceedings of the conference on Agile development
43. Kollmann J, Sharp H et al (2009) The importance of identity and vision to user experience designers on Agile projects. Agile conference, 2009. AGILE '09
44. Krohn T, Kindsmüller M et al (2009) User-centered design meets feature-driven development: an integrating approach for developing the web application myPIM. In: Kurosu M (ed) Human centered design, vol 5619. Springer, Berlin/Heidelberg, pp 739–748
45. Lee JC, Scott McCrickard D et al (2009) Examining the foundations of agile usability with eXtreme scenario-based design. Agile conference, 2009. AGILE '09
46. Losada B, Urretavizcaya M et al (2011) Efficient building of interactive applications guided by requirements models. San Sebastian. 5648 LNCS: 481–484
47. Memmel T, Gundelsweiler F et al (2007) Agile human-centered software engineering. In: Proceedings of the 21st British HCI group annual conference on people and computers: HCI… but not as we know it—volume 1. British Computer Society, University of Lancaster, UK, pp 167–175
48. Meszaros G, Aston J (2006) Adding usability testing to an agile project. Agile conference, 2006
49. Miller L (2005) Case study of customer input for a successful product, agile conference, 2005. Agile '05. IEEE
50. Najafi M, Toyoshiba L (2008) Two case studies of user experience design and agile development, agile conference, 2008. Agile '08. IEEE
51. Norman D (2006) Why doing user observations first is wrong. Interactions 13(4):50
52. Obendorf H, Finck M (2008) Scenario-based usability engineering techniques in agile development processes. CHI '08 extended abstracts on human factors in computing systems. ACM, Florence, Italy, pp 2159–2166

53. Paelke V, Nebe K (2009) Dynamic maps for future navigation systems: agile design exploration of user interface concepts. In: Human-computer interaction. Ambient, ubiquitous and intelligent interaction, pp 169–178
54. Patton J (2002) Designing requirements: incorporating usage-centered design into an agile SW development process. In: Wells D, Williams L (eds) Extreme programming and agile methods—XP/agile Universe 2002, vol 2418. Springer, Berlin/Heidelberg, pp 95–102
55. Peixoto CSA, da Silva AEA (2009) A conceptual knowledge base representation for agile design of human-computer interface. Intelligent information technology application, 2009. IITA 2009. Third international symposium on
56. Scott KM (2009) FEATURE: is usability obsolete? Interactions 16(3):6–11
57. Seffah A, Djouab R et al (2001) Comparing and reconciling usability-centered and use case-driven requirements engineering processes. IEEE Software 0, pp 132
58. Silva da Silva T, Martin A et al (2011) User-centered design and agile methods: a systematic review. Agile conference (AGILE) 2011. Salt Lake City, UT
59. Singh M (2008) U-SCRUM: an agile methodology for promoting usability. Agile, 2008. AGILE '08. Conference
60. Sohaib O, Khan K (2010) Integrating usability engineering and agile software development: a literature review. Computer design and applications (ICCDA), 2010 international conference on
61. Sy D (2007) Adapting usability investigations for agile user-centered design. J Usabil Stud 2(3)
62. Toulmin SE, Rieke RD et al (1984) An introduction to reasoning. Macmillan, New York
63. Ungar J (2008) The design studio: interface design for agile teams, agile conference, 2008. Agile '08. IEEE
64. Webster J, Watson R (2002) Analyzing the past to prepare for the future: writing a literature review. MIS Quarterly 26:13–23
65. Williams H, Ferguson A (2007) The UCD perspective: before and after agile. AGILE 2007
66. Winter J, Rönkkö K et al (2011) Meeting organisational needs and quality assurance through balancing agile and formal usability testing results software engineering techniques. In: Huzar Z, Koci R, Meyer B, Walter B, Zendulka J (eds) vol 4980. Springer, Berlin/Heidelberg, pp 275–289

Data and Information Quality Research in Engineering Construction Projects: A Review of Literature

Soffi Westin

Abstract This article presents a review of the research on Data and Information Quality (DQ/IQ) assessment in engineering construction. Through a review of 445 articles on the topic, only nine were found in the context of engineering construction. The analysis of these nine articles revealed six challenges in performing DQ/IQ assessment in this context: the iterative nature of concurrent engineering, the uniqueness of engineering data, lack of integration between processes, lack of integration between systems, lack of timely information, and lack of relevant DQ/IQ assessment frameworks and tools. The specific contributions of this paper are the identification of DQ/IQ challenges in engineering construction, their consequences, and implications for the development of relevant DQ/IQ assessment frameworks and tools. Additionally, it also identifies an area where Information Systems research can contribute, thereby extending the reach of the discipline.

1 Introduction and Related Research

Within the field of Information Systems (IS), Data and Information Quality (DQ/IQ) is a familiar concept. This paper presents a review of the research on DQ/IQ assessment in the context of engineering construction. The following presentation of related research in the areas of engineering construction and DQ/IQ will also provide the motivation for this paper.

S. Westin (✉)
University of Agder, P.O. Box 422, Kristiansand, 4604, Norway
e-mail: Soffi.Westin@uia.no

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_45,

1.1 Engineering Construction

Delays and cost overruns are commonplace in construction projects [13, 20, 23], and several researchers have reported delays as the most costly problem [5, 6, 12]. The problem has occurred in many countries, contributing to a high cost of construction [15]. One example of the magnitude of these issues is from Norway, where a committee was appointed to investigate reasons for this problem in the offshore development projects on the Norwegian continental shelf. The analysis showed 29.9 billion NOKs (≈5 billion USD) in cost overrun for 13 projects between 1994 and 1998 [5]. Project management issues (e.g., use of parallel activities) rather than technical issues were seen as the main reasons for the problem.

The effects of construction delays are not confined to the construction industry only but could also influence the overall economy of a country. In the United Arab Emirates, construction plays a major role and contributes 14 % to the GDP (gross domestic product) [6]. In Norway, the oil and gas industry contributed 25 % to the GDP in 2006 [31]. Therefore, it is essential to define the most significant causes of delays in order to avoid or minimize their impact [6].

Several causes for delays and cost overruns have been identified, and errors in drawings remain a severe challenge. Toor and Ogunlana [23] presented 75 problems encountered in a number of countries, including "errors and omissions in design documents." Dai et al. [3] identified errors in drawings as the second most significant problem and lack of needed information related to drawings as another top ten factor in construction projects located in the United States. These findings are in line with a study by Tulacz [25], which found construction workers commonly to have only 70 % of the necessary data on-site. Westin and Päivärinta [29] identified that DQ/IQ problems related to documents and drawings had the most negative impact on the profit margins of the projects executed by a European engineering and construction company. They also point out that the drawings are mainly extracted from engineering databases and conclude that the problems encountered with the drawings are related to the low level of DQ/IQ in the data sources.

1.2 Data and Information Quality

There is a lack of consensus on the definitions of data quality and information quality that distinguish the two. Data quality often (but not always) refers to technical issues, while information quality usually refers to nontechnical issues [14]. This paper does not distinguish and uses the term DQ/IQ, adopting Wang and Strong's [27] definition as "fitness for use" in relation to the recognized purposes of information use and user groups. In the IS literature, the cost of an insufficient level of DQ/IQ has been found to be huge [19, 22], and problems have been observed across numerous corporations [19, 26]. Awareness of these issues has grown rapidly over recent years, and research is now covering several topics [14] and contexts [8]. Data warehousing, health-care registers, and retailing- and Internet-related systems (e.g., booking, online stores) are all contexts that have been studied to a certain extent. Several assessment frameworks exist and their similarities and differences have been compared by Batini et al. [1]. Common for the mentioned contexts is the assumption of consistency and accuracy of the data records from the moment they are inserted, and most frameworks have been developed to support these assumptions.

To address the issue of persistence of DQ/IQ problems, IS researchers have called for more investigation specifically in new and different contexts [14]. Researchers also called for providing practitioners with tools that can assess the level of DQ/IQ and provide comparative assessment over time to senior management [18].

Based on the above discussion, the following literature review is focused on to two questions:

What are the challenges related to DQ/IQ assessment in the context of engineering construction?

What do these challenges imply for the development of DQ/IQ assessment frameworks and tools?

2 Research Method

Literature reviews are conducted for a variety of reasons; learning the breadth of research in a topic is one of them [17]. The purpose of this literature review was to identify DQ/IQ research performed in the context of engineering construction projects; hence, one focus was the breadth of research on this topic. The method chosen was based on the eight-step guideline proposed by Okoli and Schabram [17] and the concept matrix suggestions by Webster and Watson [28].

To cover a broad range of fields, three literature databases—EBSCO, Inderscience Publishers, and Scopus—were chosen. These databases include fields such as engineering construction, project management, and IT/IS. Scopus alone includes 428 engineering journals of which several are ranked among the top journals (e.g., "Journal of Construction Engineering and Management" and "Automation in Construction").

To focus specifically on IT/IS, 18 publication outlets were chosen for searches based on two criteria: (1) journals and conferences frequently referred to in DQ/IQ literature and (2) significant journals and conferences in the IS field. These outlets are:

- ACM Computing Surveys
- AMCIS (Americas Conference on Information Systems)
- CAIS (Communications of the Association for Information Systems)
- ECIS (European Conference on Information Systems)
- European Journal of Information Systems
- ICIQ (International Conference on Information Quality)
- ICIS (International Conference on Information Systems)
- Information Systems Journal
- Information Systems Research

Table 1 Search terms

IT/IS outlets	All outlets (EBSCO, Inderscience Publishers, and Scopus)
"data quality"	
("data quality" and "engineering")	("data quality" and "engineering project")
("data quality" and "construction")	("data quality" and "construction project")
"quality of data"	
("quality of data" and "engineering")	("quality of data" and "engineering project")
("quality of data" and "construction")	("quality of data" and "construction project")
"information quality"	
("information quality" and "engineering")	("information quality" and "engineering project")
("information quality" and "construction")	("information quality" and "construction project")
"quality of information"	
("quality of information" and "engineering")	("quality of information" and "engineering project")
("quality of information" and "construction")	("quality of information" and "construction project")

- International Journal of Information Quality
- Journal of Data and Information Quality
- Journal of Information Technology
- Journal of Management Information Systems
- Journal of Strategic Information Systems
- Journal of the Association for Information Systems
- MIS Quarterly
- PACIS (Pacific Asia Conference on Information Systems)
- SJIS (Scandinavian Journal of Information Systems)

All search terms are listed in Table 1.

An important criterion for the types of studies to include was that the search terms would be found in either abstract, title, or keywords. All outlets were searched for all years. Some studies were rejected due to various reasons: misfit of context; a promising abstract in English, then the rest of the article available only in a foreign language (e.g., Chinese); and some "studies" were proposals. The remaining studies were submitted to full-text analysis, after which several more studies were eliminated. Backward and forward (Bw/Fw) searches were then conducted based on the identified studies. The searches were performed from late July 2011 to September 2011.

3 Results

Table 2 displays the search result for the IT/IS outlets, the non-IT/IS outlets, and a final Bw/Fw search.

In total for all search terms, 445 studies were found; however, after the practical screening only eight studies qualified as DQ/IQ research in the context of engineering construction projects. Bw/Fw searches resulted in the addition of one study.

Table 2 Search results

Search objects	# of unique hits	# of relevant studies
IT/IS outlets	408	6
Non-IT/IS outlets	37	2
Bw/Fw search	–	1
Total	445	9

A total of nine studies fulfilled the requirements. All studies were peer reviewed, except from Dobson and Martinez [4], which is an ABB review and not a scientific publication. However, practice could provide necessary information on the topic of this literature review, and this paper was included for that reason, as well as fulfilling the search criteria.

With the first research question in mind, these nine studies were analyzed with the aim to identify any reported challenges related to DQ/IQ assessment. Based on this analysis six challenges were identified. These challenges were the iterative nature of concurrent engineering, the uniqueness of engineering data, lack of integration between processes, lack of integration between systems, lack of timely information, and lack of relevant DQ/IQ assessment frameworks and tools in the context of engineering construction projects. Table 3 below lists the nine studies and the challenges they elaborated.

3.1 Challenge 1: The Iterative Nature of Concurrent Engineering

The globalization of engineering leads to shorter delivery schedules and forces a break in the traditional linear engineering model [4]. The resulting iterative nature of concurrent engineering creates challenges for DQ/IQ assessment. Many of the processes in concurrent engineering are interdependent. This fact forces the engineers to proceed with partial information, incomplete knowledge, and subjective interpolations [2]. Late changes in requirements or changes caused by design errors—erroneous or missing information—lead to rework [24]. Rework initiates more iteration, which, in turn, leads to delays. Delays—in terms of waiting for information—reduce the profit margins of the projects [29]. To further complicate the situation, the engineering asset management process, which consists of several interdependent stages, not only is iterative but also requires collaboration with other processes in the organization, such as supply chain management and supplier relationship management [10, 16].

3.2 Challenge 2: The Uniqueness of Engineering Data

One of the unique characteristics of engineering data is that the “correct answer” for the values to be inserted into the engineering asset data record is not known at the

Table 3 Identified challenges

Article	Context details	Challenges: The iterative nature of concurrent engineering	The uniqueness of engineering data	Lack of integration between processes	Lack of integration between systems	Lack of timely information	Lack of relevant DQ/IQ assessment frameworks and tools
Lee et al. [9]	Assessed the process of remanufacture; an aircraft fuel pump			X	X		
Neely et al. [16]	Assessed current DQ/IQ tools in the realm of engineering asset management	X	X	X	X		X
Dobson and Martinez [4]	Considerations on globalization in plant engineering (large and global engineering projects)	X			X		
Lin et al. [10]	Assessed engineering asset management in Australian engineering organizations	X	X	X	X		X
Lin et al. [11]	Explored DQ/IQ issues in engineering asset management (AM) and developed and tested a framework for AM	X	X	X	X		X
Blechinger et al. [2]	Proposed how to support production of high-quality data in concurrent plant engineering	X	X	X	X	X	X
Wingkvist et al. [30]	Assessed quality in technical documentation of: A mobile phone Parts of a warship						X
Tribelsky and Sacks [24]	Assessed information flows in 14 civil engineering projects, all part of a major airport Construction project	X		X		X	
Westin and Päivärinta [29]	Ranked problems experienced in engineering Construction projects within one organization (offshore installations)	X		X		X	

time of insertion [16]. Physical engineering assets and the related asset data need not necessarily be present at the same time. For example, in engineering design, engineering assets may exist only as referenced logical/functional elements in data registers and data models. In addition, data are collected in a variety of formats and have to be shared among various technical and business systems. Some of the data is textual and, as such, unverifiable by the current automated processes [10, 16]. This is different from typical business environments, where data often is provided in fixed formats and shared only among relevant business systems [10].

3.3 Challenge 3: Lack of Integration Between Processes

Lack of integration between processes leads to difficulties in connecting various information [9]. This could lead to a time-consuming search for existing but not easily accessible information. Tribelsky and Sacks [24] identified a positive correlation between the quality of information flows and the effectiveness of design documents (e.g., less rework due to more accurate drawings) and found that unstable information flows were associated with unpredictable project outcomes. Westin and Päivärinta [29] identified a similar problem; one of the top-ranked problems reflected missing information flows between teams contributing to the projects, which led to drawings that were not congruent. This problem was also identified by Blechinger et al. [2], who indicated the importance of notifying users of work copies when changes in the master Information Management Systems (IMS) occurred. Data captured in a variety of formats, processed in isolation, and stored in legacy systems is usually process dependent, which makes it difficult to reuse data in other processes [16]. Management of all engineering asset data processes has to be performed across all significant lifecycle stages, and there is a need for synchronization of asset lifecycle management with other enterprise processes [10].

3.4 Challenge 4: Lack of Integration Between Systems

Several DQ/IQ problems stem from a lack of integration between systems: different values and different numbers of replicated items in different Information Management Systems [2]; data that cannot be exported because they are captured with sensors only readable by specially designed monitoring systems [16]; specialized systems bought from multiple vendors leading to a very difficult integration job for the end users [10]; and lack of integration between business systems and technical systems making it difficult to achieve a comprehensive overview of status, resulting in significant DQ/IQ consequences.

3.5 Challenge 5: Lack of Timely Information

Lack of timely information is identified as a major challenge in three of the studies [2, 24, 29]. The resultant waiting causes delays in progress of the project [2] and leads designers to shift attention to other projects, which in turn leads to extra time spent on recapitulation when work on the project is resumed [24]. A delay in receiving necessary information causes the design to be based on incomplete information [29], which in turn could lead to rework. This is similar to what happens when engineers are forced to proceed with partial information due to process interdependencies and short delivery schedules, as mentioned earlier.

3.6 Challenge 6: Lack of Relevant DQ/IQ Assessment Frameworks and Tools

Existing DQ/IQ solutions and tools focus on typical business systems, such as financial systems, customer databases, and CRM systems [16]. These tools can be categorized as (1) auditing tools, which aim to create a variance report where data that do not conform to the business rules can be manually examined; (2) cleansing tools, which aim to automatically verify and correct data; and (3) migration tools, which physically move data from one location to another [16]. These tools are primarily directed at the quality dimension of accuracy; hence, the "correct answer" has to be known in order to perform the comparison of inserted data with real-world data. However, this is not sufficient for DQ/IQ assessment in engineering construction [2, 10, 16]. A more relevant framework has been developed and tested by Lin et al. [11]. Their framework contributes to the lifecycle of engineering asset management (AM) and addresses the challenges identified from an AM lifecycle point of view.

4 Discussion

In order to perform DQ/IQ assessment involving large amounts of data, a relevant tool is needed. This review reveals that most existing DQ/IQ assessment frameworks and tools are developed on the assumptions that the "correct answer" is known by the time of data insertion, which is not the case in engineering construction [16]. The AM framework [11] represents an exception. The framework considers the identified challenges and contributes extensively to the AM part of engineering construction. However, the poor level of DQ/IQ in drawings mentioned in the introduction is related to engineering design as a whole (not solely AM). Hence, engineering construction projects are still in need of relevant DQ/IQ assessment frameworks and tools. The question that arises is: which concepts need to be included in such a framework to make it suited to this specific context?

When analyzing the findings of this review, an interesting issue emerged: some of the identified challenges seem to be unavoidable in engineering construction. Therefore, an assessment tool should be developed on the assumption that these challenges are unavoidable and their consequences need to be handled appropriately. Below is a discussion on which identified challenges are unavoidable and associated implications for the development of relevant DQ/IQ assessment frameworks and tools.

The iterative nature of concurrent engineering is unavoidable. This is also the case for the uniqueness of engineering data. Lack of integration between processes and lack of integration between systems are core foci for an IS researcher. However, it is far too risky to change processes and systems within a single project [2]. These changes and alignments are better done at a higher, organizational level. The implication is that within a single project these challenges are also unavoidable. Apart from the identified lack of relevant DQ/IQ frameworks and tools, only one challenge remains: lack of timely information. Assuming several reasons exist for *why* information is lacking, *elements* of the answer could include the relationship between the identified challenges. This is elaborated below:

(1) The iterative nature of concurrent engineering forces the engineers to proceed with partial information [2]. This implies a lack of necessary information. (2) The uniqueness of engineering data is characterized by the fact that the "correct answer" is not always known at the time of data insertion [16]. This means the information is not there when first needed (i.e., there is a lack of timely information). (3) Lack of integration between processes could lead to time-consuming searches for existing but not easily accessible information [9], which could also lead to lack of timely information. (4) Lack of integration between systems could lead to difficulties with exporting data [16], and the resulting delay would again contribute to lack of timely information. Taken together, it appears that lack of timely information could also be viewed as unavoidable, at least in some situations.

What are the consequences then? Since these four challenges could all lead to lack of timely information, engineers may have to proceed with only partial information [2]. The consequences will be incompleteness and/or inconsistencies in information used in the projects. Research findings indicate that engineering drawings are mainly extracted from such systems [29]. It was also found that errors in drawings comprise one of the most significant problems related to productivity factors [3]. Hence, proceeding with partial information will have negative impact on the level of DQ/IQ in the data sources and the extracted drawings. The connections between the different challenges and the consequences are summed up in Fig. 1, where dotted arrow means "could lead to" and solid arrows mean "leads to."

A DQ/IQ framework has to include concepts and guidelines that consider these issues and help overcome the challenges.

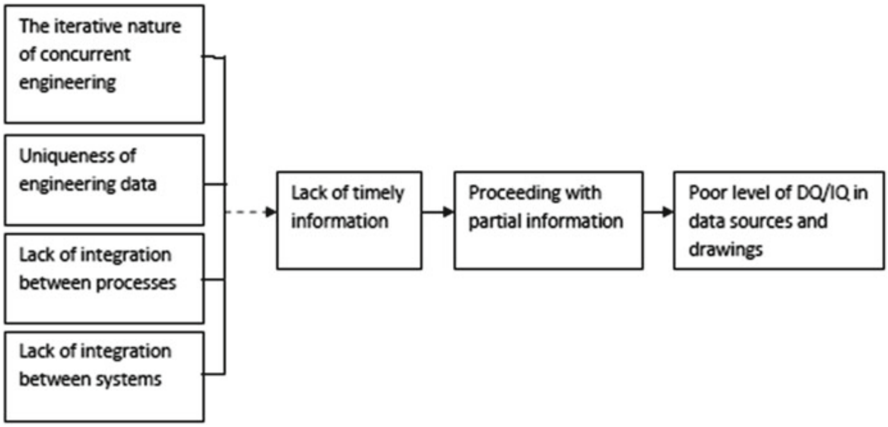


Fig. 1 Connections between identified challenges and level of DQ/IQ

5 Implications and Conclusion

This review has identified several challenges for DQ/IQ in the engineering construction context. The key challenge now is to develop a relevant assessment framework and a tool to implement that framework. In doing so, it is important that such tools have to diverge from the common assumption in existing DQ/IQ research (i.e., that the "correct answer" is known at the time of data insertion). In the present context, the assumption is the opposite: the "correct answer" is not always known. This has some implications: (1) DQ/IQ assessment frameworks and tools have to allow for partial information and, consequently, incompleteness and inconsistencies in the data sources during early phases of engineering construction projects, and (2) thereafter, further guidelines would have to suggest a way to mitigate the unavoidable poor level of DQ/IQ in later project phases.

Developing a DQ/IQ assessment tool requires expertise in the domain of engineering construction. That requires close collaboration between the researcher and the engineers; hence, access to an engineering construction organization is needed. Since the goal is to develop a tool, suitable methods could be design research (DR) or action design research (ADR). Both methods emphasize close collaboration between researchers and organizations. DR posits that it is through the building and application of such a tool (the artifact) that the problem domain is understood and its solution achieved [7]. ADR emphasizes that such a tool (the artifact) emerges from interaction with the organizational context [21]. No matter which method is chosen, engineering construction is in need of the development of a DQ/IQ assessment tool. Finally, the surprising finding that only nine papers are found on this topic is in itself intriguing. This calls for more research in this area. IS research can contribute by building up a research agenda in DQ/IQ in engineering construction. Findings from general DQ/IQ research in IS will be very useful for this agenda.

References

1. Batini C, Cappiello C, Francalanci C, Maurino A (2009) Methodologies for data quality assessment and improvement. ACM Comput Surv 41(3):1–52
2. Blechinger J, Lauterwald F, Lenz R (2010) Supporting the production of high-quality data in concurrent plant engineering using a MetaDataRepository. In: Americas conference on information systems, Lima, Peru, 12–15 Aug 2010
3. Dai J, Goodrum PM, Maloney WF (2009) Construction craft workers' perceptions of the factors affecting their productivity. J Constr Eng Manage 135(3), 217–226 doi:10.1061/(asce)0733-9364(2009) 135:3(217)
4. Dobson D, Martinez R (2007) Integrated engineering. ABB Review (SPEC. REP.), pp 48–52
5. Dyrhaug Q (2002) A generalized critical success factor process model for managing offshore development projects in Norway. Doctoral thesis, Norwegian University of Science and Technology (NTNU), Trondheim, Norway
6. Faridi AS, El-Sayegh SM (2006) Significant factors causing delay in the UAE construction industry. Constr Manage Econ 24(11):1167–1176
7. Hevner AR, March ST, Jinsoo P, Ram S (2004) Design science in information systems research. MIS Quart 28(1):75–105
8. Lee YW (2004) Crafting rules: context-reflective data quality problem solving. J Manage Inform Syst 20(3):93–119
9. Lee Y, Allen T, Wang R (2001) Information products for remanufacturing: tracing the repair of an aircraft fuel-pump. In: International conference on information quality, 2001, Citeseer, pp. 77–82:
10. Lin S, Gao J, Koronios A, Chanana V (2007) Developing a data quality framework for asset management in engineering organisations. Int J Inf Qual 1(1):100–126
11. Lin S, Gao J, Koronios A (2008) A data quality framework for engineering asset management. Aust J Mech Eng 5(2):209–219
12. Ling FYY, Pham VMC, Hoang TP (2009) Strengths, weaknesses, opportunities, and threats for architectural, engineering, and construction firms: case study of Vietnam. J Constr Eng Manage 135(10):1105–1113
13. Long ND, Ogunlana S, Quang T, Lam KC (2004) Large construction projects in developing countries: a case study from Vietnam. Int J Proj Manage 22(7):553–561
14. Madnick SE, Wang RY, Lee YW, Zhu H (2009) Overview and framework for data and information quality research. J Data Inf Qual 1(1):1–22
15. Memon AH, Abdul Rahman I, Abdul Azis AA (2011) Preliminary study on causative factors leading to construction cost overrun. Int J Sustain Constr Eng Technol 2(1)
16. Neely MP, Lin S, Gao J, Koronios A (2006) The Deficiencies of current data quality tools in the realm of engineering asset management. In: Americas conference on information systems, Acapulco, México, 4–6 Aug 2006: Proceedings of the 12th Americas conference in information systems
17. Okoli C, Schabram K (2010) A guide to conducting a systematic literature review of information systems research. Sprouts: working papers on information systems 26(10): 1–49.
18. Pipino LL, Lee YW, Wang RY (2002) Data quality assessment. Commun ACM 45(4ve):211–218
19. Ramaswamy M (2006) On the phenomenon of information dilution. Issues Inform Syst VII(2):289–292
20. Sambasivan M, Soon YW (2007) Causes and effects of delays in Malaysian construction industry. Int J Proj Manage 25(5):517–526
21. Sein MK, Henfridsson O, Purao S, Rossi M, Lindgren R (2011) Action design research. MIS Quart 35(1):37–56
22. Strong DM, Lee YW, Wang RY (1997) Data quality in context. Commun ACM 40(5):103–110. doi:10.1145/253769.253804
23. Toor S-U-R, Ogunlana SO (2008) Problems causing delays in major construction projects in Thailand. Constr Manage Econ 26(4):395–408

24. Tribelsky E, Sacks R (2011) An empirical study of information flows in multidisciplinary civil engineering design teams using lean measures. Archit Eng Design Manage 7(2):85–101
25. Tulacz GJ (2005) Worrying about labor shortage. Engineering News Record, 14 Nov 2005, www.enr.com. Accessed 29 Sep 2011
26. Wand Y, Wang RY (1996) Anchoring data quality dimensions in ontological foundations. Commun ACM 39(11):86–95
27. Wang RW, Strong DM (1996) Beyond accuracy: what data quality means to data consumers. J Manage Inform Syst 12(4):5–33
28. Webster J, Watson RT (2002) Analyzing the past to prepare for the future: writing a literature review. [Editorial]. MIS Quart 26:xiii–xxiii
29. Westin S, Päivärinta T (2011) Information quality in large engineering and construction projects: A Delphi Case Study. In: European conference on information systems, Helsinki, Finland, 9–11 June 2011
30. Wingkvist A, Ericsson M, Lincke R, Löwe W (2010) A metrics-based approach to technical documentation quality. In: International conference on information quality, 2010, IEEE, pp 476–481
31. Wolf C, Pollitt M (2009). The welfare implications of oil privatization: a cost-benefit analysis of Norway's statoil. Cambridge Working Paper in Economics No. 0907, EPRG Working Paper No. 0905

Aligning Communication Analysis with the Unifying Meta-model for Enterprise Modeling

Iyad Zikra, Sergio España, Marcela Ruiz, Oscar Pastor, and Janis Stirna

Abstract Enterprise Modeling (EM) captures and represents organizational knowledge in models that cover different views of the enterprise. The models can be leveraged in the development of information systems. Investigating how to use them as input to model-driven development (MDD) is an open challenge. This paper explores how a holistic EM approach, represented by a unifying meta-model, can benefit from integrating with Communication Analysis, a communication-oriented business process modeling and requirements engineering method. As a first step towards an integrated EM-enabled MDD approach, the unifying meta-model and Communication Analysis meta-model are aligned. A set of guidelines for transforming models conforming to the unifying meta-model to Communication Analysis models are formulated. The approach is illustrated using a lab demo.

1 Introduction

Enterprise Modeling (EM) is a process whereby organizational intentions, behavior, and design are captured and represented in terms of business goals, processes, concepts, actors, and high-level information system (IS) requirements using conceptual models.

Research is partially supported by the Spanish Ministry of Science and Innovation project PROS-Req (TIN2010-19130-C02-02), the Generalitat Valenciana project ORCA (PROMETEO/2009/015), and the Santiago Grisolía grant (GRISOLIA/2011/005) and cofinanced by ERDF structural funds and Jubileumsdonationen K & A Wallenbergs Stiftelse.

I. Zikra (✉) • J. Stirna
Department of Computer and Systems Sciences, Stockholm University, Stockholm, Sweden
e-mail: iyad@dsv.su.se; js@dsv.su.se

S. España • M. Ruiz • O. Pastor
Centro de Investigación PROS, Universitat Politècnica de València, València, Spain
e-mail: sergio.espana@pros.upv.es; lruiz@pros.upv.es; opastor@pros.upv.es

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8_46,

It helps in shaping the business vision of an organization and identifying suitable means to attain it [1]. Enterprise models can also be the input for IS development. Model-driven development (MDD) is a relatively recent IS development approach where the focus is shifted from programming code to models as the primary development artifact [2]. Formal models describe the system on several abstraction layers, and concrete models are derived from more abstract models until executable code is reached. The advent of the Unified Modeling Language (UML) [3] as a de facto standard language and the model-driven architecture (MDA) [4] as a reference guide has paved the way for several methods, some of which achieve automatic software code generation. Following a bottom-up effort, the OO-method, an object-oriented MDD framework with code generation capabilities [5], has been recently extended with Communication Analysis (CA), a communication-oriented business process modeling and RE method [6]. However, MDD approaches are still incomplete in terms of providing high-level enterprise views [7].

A recent proposal for linking EM and MDD follows a top-down approach to provide a unifying meta-model that offers an integrated view of the enterprise and its supporting IS [8]. The unifying meta-model covers EM views and integrates them with software architecture and implementation platform views. However, it has no model transformation or code generation strategies. Therefore, it is an open challenge to exploit the potential of the unifying meta-model as an MDD artifact.

In this paper, we explore the alignment of the unifying EM meta-model with CA to (1) demonstrate the operationalization of the unifying meta-model and (2) provide a business context for CA models. This proposal sets the ground towards concrete integration of EM and MDD and is a first step combining the unifying meta-model and CA in single approach. The contributions of the paper are:

- An alignment of the unifying meta-model and CA meta-model, so as to identify similar modeling constructs
- Guidelines for transforming models of the unifying meta-model to CA models
- Extending CA with business context capabilities of the unifying meta-model
- A demonstration of the proposed alignment and guidelines using a lab demo

The paper is structured as follows. Sections 2 and 3 give backgrounds to the unifying EM meta-model and CA, respectively. Section 4 presents the alignment of the two frameworks, including the transformation guidelines. Section 5 illustrates the alignment with a lab demo. Section 6 discusses related works and Sect. 7 presents concluding remarks and issues for future work.

2 A Unifying Meta-model for Enterprise Modeling

EM typically focuses on organizational design and formulates high-level IS requirements. MDD is primarily concerned with IS design and development. In order to provide an integrated view of EM and MDD, a unifying meta-model has been proposed [8] to establish a formal connection between the two activities and enable the development of an IS that better fits organizational needs.

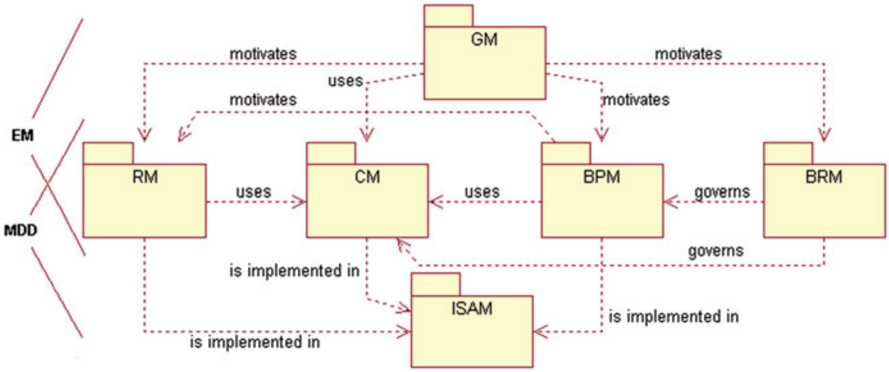


Fig. 1 The complimentary views of the unifying meta-model

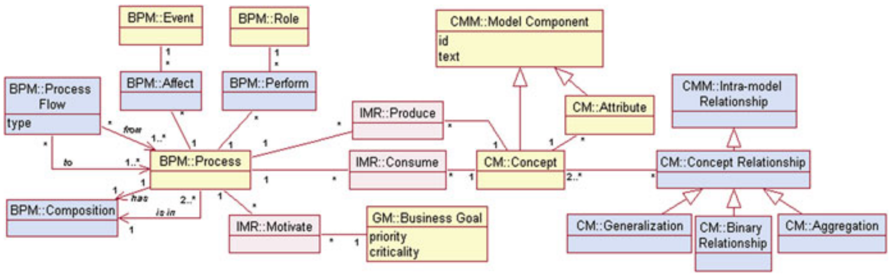


Fig. 2 View from the unifying meta-model showing CM and BPM

The unifying meta-model is comprised of multiple complimentary views (Fig. 1). It is based on Enterprise Knowledge Development (EKD) [1] and consists of four enterprise-level models—the *Goal Model (GM)*, *Concepts Model (CM)*, *Business Process Model (BPM)*, and *Business Rules Model (BRM)*, as well as two system-level models—the *Requirements Model (RM)* and *IS Architecture Model (ISAM)*. Model components are linked across different views using fine-grained *Inter-Model Relationships (IMR)*, which depict associations between concepts of different models. IMRs ensure the holistic view of the organization and the IS.

The focus in this paper is on aligning parts of the unifying meta-model with CA. Namely, concepts from the CM and BPM are considered (Fig 2). This is a first step towards an integrated EM-driven MDD approach. The CM captures concepts which represent the static aspects of organizations and its IS. Business processes that describe the necessary steps for the organization to realize its goals are captured in the BPM. Concepts are connected to processes using the IMRs *consume* and *produce*, denoting the inputs guiding process execution and outputs of the execution, respectively. Business goals are connected to the processes which they affect using the IMR *motivate*. A description of all the views of the unifying meta-model is found in [8].

3 Communication Analysis

Communication Analysis (CA) is a business process modeling and RE method that proposes undertaking IS analysis from a communicational perspective [6]. It offers a requirements structure and elicitation and modeling techniques. CA defines the *Communicative Event Diagram (CED)* which describes information-related actions, identified by the unity criteria and carried out in a complete way following an external stimulus. The *Event Specification Template (EST)* allows structuring the requirements associated to a communicative event. Among other requirements, EST contains a description of new and meaningful information that is conveyed to the IS in the event, specified using *Message Structures (MS)*. Figure 3 shows a meta-model that supports many of the constructs defined by the method specification. The models defined in CA are organized in five levels of abstraction, described briefly in the next sections. A full description of CA is found at [9].

A model transformation strategy has been defined to derive OO-method conceptual models from CA requirements models [10]. The transformations derive an initial model that is capable of being compiled to automatically generate code. The bottom-up integration of CA and the OO-method covers the software development lifecycle from RE to code generation. However, this MDD approach does not yet cover all the relevant views acknowledged to be relevant for EM. Integrating CA with the unifying EM meta-model will provide the necessary business context.

4 Aligning CA with the Unifying Meta-model

The unifying meta-model serves as an underlying foundation for creating modeling environments that are used by organizations to develop supporting IS. On the other hand, CA provides a concrete method for developing MDD-oriented models. Aligning concepts from EM and CA will help demonstrate the viability of the unifying meta-model using a tested and proven approach. Concurrently, the business context surrounding CA models will be captured explicitly using the views offered by the unifying meta-model. An overview of the alignment is shown in Fig. 4. As a first step for integration, comparable concepts from the CM, BPM, and RM of the unifying meta-model and from L2 and L3 in CA are identified, analyzed, and aligned. Alignment between other parts of both approaches is visited briefly.

4.1 CA Process Level (L2)

The CED in L2 can be roughly compared to BPM. They both describe activities that transfer and transform information between actors. An overview of the alignment on this level is shown in Table 1. A communicative event in CED can be aligned

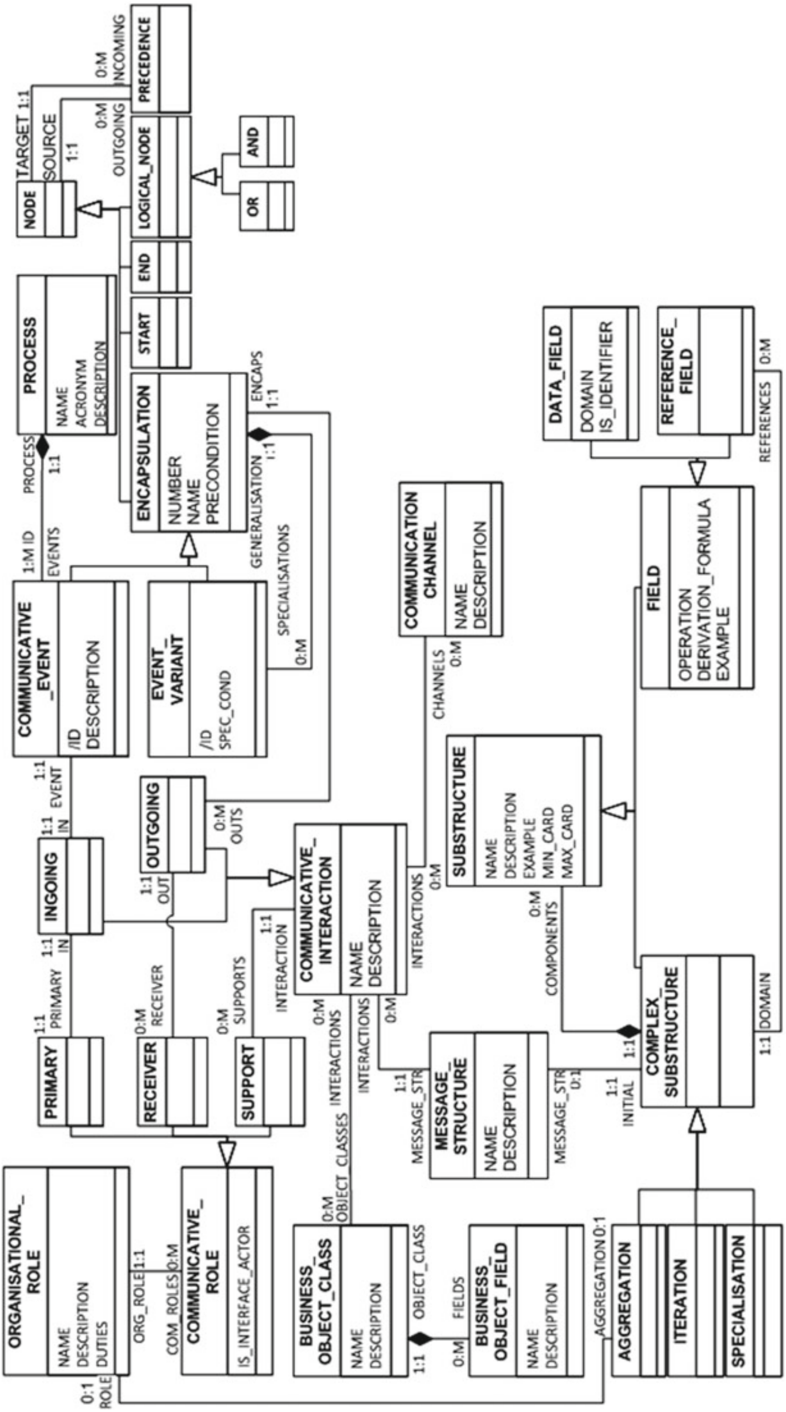


Fig. 3 View of the CA meta-model

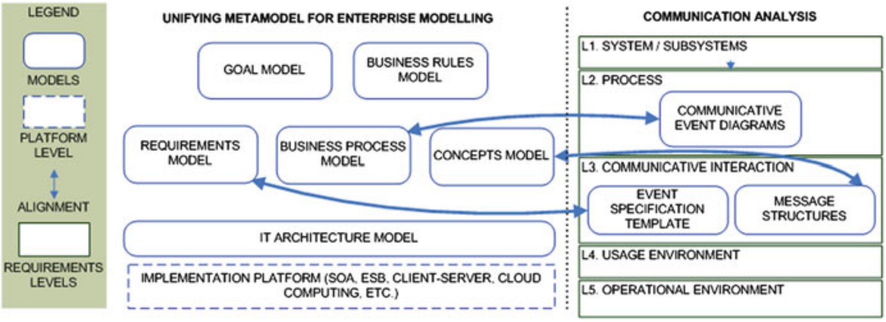


Fig. 4 Alignment between the unifying EM meta-model and CA

Table 1 Alignment between L2 of CA and the unifying EM meta-model

Communication analysis—L2	The unifying meta-model
Communicative event (Communicative_Event)	Process which fulfills CA's unity criteria (BPM::Process)
Refinement possibility	Process decomposition (BPM::Composition)
Name of communicative event (Encapsulation.Name)	Text of process (CMM::Model Component.text)
Ingoing communicative interaction (Ingoing)	"Produce" and "perform" relationships, sharing the same process (IMR::Produce + BPM::Perform)
Outgoing communicative interaction (Outgoing)	"Consume" and "perform" relationships, sharing the same process (IMR::Consume + BPM::Perform)
Precedence (Precedence)	The order in which processes produce and consume concepts (IMR::Produce/Consume + CM::Concept)
Event variants as alternatives for communicative events (Communicative_Event + Event_Variant + Encapsulation)	Process flow, of types OR and XOR (BPM::Process Flow)
Communicative role (Communicative_Role)	Role performing a process (BPM::Role + BPM::Perform)
Primary role (Primary)	Role that is an information provider to the process (BPM::Role)
Receiver role (Receiver)	Role that consults the information produced by the process (BPM::Role)

with a BPM process only if the process fulfills the unity criteria. BPM processes can be decomposed into smaller processes, but the unity criteria in CA prevent splitting communicative events into smaller events. However, CA includes the possibility of refining communicative events into activities.

Communicative interactions in CA represent the information exchange between communicative roles. Roles that provide information have ingoing interactions to events, and this is represented in BPM by combining roles which perform a process and concepts produced by the process. Similarly, CA roles that consult information have outgoing interactions from events, and this is captured in BPM by combining roles which perform a process and concepts consumed by the process.

Table 2 Alignment between L3 of CA and the unifying EM meta-model

Communication analysis—L3	The unifying meta-model
Reference field in a message structure (Reference_Field + Message_Structure)	Concept, consumed and produced by processes (CM::Concept)
Data field (Data_Field)	Attribute (CM::Attribute)
Aggregation (Aggregation + Reference_Field)	Aggregation (CM::Aggregation + CM::Concept)
Iteration (Iteration + Reference_Field)	Aggregation (CM::Aggregation + CM::Concept)
Specialization (Specialization)	Generalization (CM::Generalization)
–	Binary relationship (CM::Binary Relationship)
Message requirement (Message_Requirement)	Requirement referring to Concepts (RM::Requirement + IMR::Refer to + CM::Concept)
Contact requirement (Contact_Requirement)	Requirement referring to Processes (RM::Requirement + IMR::Refer to + BPM::Process)
Reaction requirement (Reaction_Requirement)	Requirement referring to Processes (RM::Requirement + IMR::Refer to + BPM::Process)

Precedence in CED dictates the execution order of communicative events. This is realized in BPM as the path created by processes producing and consuming concepts. Furthermore, the flow of execution can also be described in BPM using process flow. This is captured in CA using event variants that specialize communicative events, hiding the details of the alternatives and simplifying the CED.

The unifying meta-model defines one type of role—the entity responsible for business goals (through "define" relationship) and the execution of processes (through "perform" relationship). CA defines more specific roles. A communicative role provides/receives information to/from other communicative roles through communicative events and can be a primary, receiver, or support role. Communicative roles in BPM manifest as "perform" IMR between roles and processes.

4.2 CA Communicative Interaction Level (L3)

After specifying the communicative events in L2, CA defines the details of the events. Table 2 gives an overview of the alignment on this level. The information exchanged in communicative interactions is captured using MS, which can be seen in the unifying meta-model as concepts being consumed and produced by processes. Considering the unity criteria, care must be taken when relating concepts to MS since not all processes can be substituted with communicative events.

Data fields in MS carry information of simple types (numbers, strings, etc.) and exist in CM as attributes. MS also include reference fields that point to complex structures, which correspond to CM concepts. Aggregation and iteration in MS allow grouping of different structures and repetition of the same structure, respectively. Both are captured in CM as aggregation. The specialization structure in MS is equivalent to generalization in CM, albeit in the opposite direction. Finally, binary relationships in CM are used to relate concepts arbitrarily. However, MS is not intended to design data models; it provides dynamic views over exchanged information. Hence, binary relationships have no counterparts in MS.

L3 defines, in addition to MS, detailed textual specifications of communicative events as ESTs, including event metadata and requirements associated with event realization. Requirements defined in RM are generic, unlike EST requirements which are specialized based on its nature, context, and type of specified constraint. Message requirements in EST that control the structure of messages are captured in RM using requirements with references to concepts. Moreover, contact and reaction requirements in EST, describing interactions and implications of communicative events, are implemented in RM using requirements that refer to processes.

4.3 Other CA Levels and Unifying Meta-model Views

Communicative events are analyzed in L4 in terms of information-related actions, which are realized as user interfaces. The unifying meta-model does not address user interfaces, and extending it with the capabilities of L4 is another step towards a complete EM-driven MDD approach. This effort is part of the future plans for our research. As for L5 and ISAM, the benefit of aligning them is limited by their changing and platform-specific nature, which arises from the variety of implementation platforms and architectures. Work is currently ongoing for integrating L5 and ISAM rather than aligning them, thus offering a single customizable modeling base that abstracts from all possible implementation platforms.

4.4 Transformation Guidelines

Based on the considerations above, the following transformation guidelines have been compiled to transform BPM and CM to CA models, where it can be further processed to produce IS [11].

Guideline 1. For each BPM::Process (P), if P fulfills the unity criteria, then create a Communicative_Event. When P does not fulfill the unity criteria, the analyst must consider combining P with other process to create a single communicative event or splitting P into multiple communicative events.

Guideline 2. The ID and Text of a BPM::Process suggest the ID and Name of its Communicative_Event counterpart. ID and Name can differ from ID and Text since not all BPM::Processes are transformed into Communicative_Events.

Guideline 3. For each BPM::Process (P) that has part BPM::Processes (using BPM::Composition), if the part BPM::Processes have been identified as Communicative_Events, then create a Process in CED for P.

Guideline 4. For each BPM::Process (P) that has part BPM::Processes (using BPM::Composition), if P qualifies as a Communicative_Event (C), then create an activity in C for every part of P following the refinement possibility in CA.[1]

Guideline 5. For each BPM::Role (R) identified by the analyst as a role that provides the information consumed by a BPM::Process (P), create a Primary (Pr) role in CED. Consequently, an Ingoing relationship is created between the Communicative_Event for P and Pr.

Guideline 6. For each BPM::Role (R) identified by the analyst as a role that consults the information produced by a BPM::Process (P), create a Receiver (Re) role in CED. Consequently, an Outgoing relationship is created between the Communicative_Event for P and Re.

Guideline 7. For each two BPM::Processes (P1 and P2), if P1 has an IMR::Produce relationship with a CM::Concept (C) and P2 has a IMR::Consume relationship with C, then create a Precedence relationship between the corresponding Communicative_Events (E1 and E2, respectively) where E1 precedes E2.

Guideline 8. For each BPM::Process Flow (F) of types OR or XOR, create one Communicative_Event (E), then create an Event_Variant that specializes E for every BPM::Process on the outgoing end of F.

Guideline 9. For each CM::Concept (C) that has an IMR::Produce relationship from a BPM::Process (P), create a Reference_Field in a Message_Structure (M) and associate M with the Ingoing relationship of the Communicative_Event for P.

Guideline 10. For each CM::Concept (C) that has an IMR::Consume relationship to a BPM::Process (P), create a Reference_Field in a Message_Structure (M) and associate M with the Outgoing relationship of the Communicative_Event for P.

Guideline 11. For each CM::Attribute of a CM::Concept that has been identified as part of a Message_Structure (M), create a Data_Field as part of M.

Guideline 12. For every CM::Aggregation (A) relationship, if A aggregates different CM::Concepts (C), then create an Aggregation structure as part of the Message_Structure that includes the Reference_Fields corresponding to C.

Guideline 13. For each CM::Aggregation (A) relationship, if A aggregates several copies of the same CM::Concept (C), then create an Iteration structure as part of the Message_Structure that includes the Reference_Field corresponding to C.

Guideline 14. For every CM::Generalization (G) relationship involving a CM::Concept that has been identified as part of a Message_Structure (M), create a Specialization structure that involves M in the opposite direction of G. (If necessary, create a Message_Structure for the other CM::Concept in G).

[1] The current CA meta-model does not support communicative event refinement; this guideline is included for future extensibility.

5 Illustrative Lab Demonstration

The proposed transformation guidelines are applied in a lab demo to illustrate their use and the feasibility of the alignment proposal. The case describes the business process model of Photography Agency Inc., which manages illustrated reports and distributes them to publishing houses. The scope covers the management of photographers, publishing houses, and regular and exclusive reports. Delivery of invoice reports to publishing houses and payment reports to photographers is also within the scope. Only a part of Photography Agency Inc. is presented. Figure 5 shows the Photography Agency Inc. case specified using BPM, while Fig. 6 shows the CED resulting from applying the transformation guidelines.

Process P1 is found not to fulfill the unity criteria because no new information is provided to the IS. Following *Guideline 1*, process P2 is analyzed and communicative event PHO1 is created to represent both P1 and P2. Other processes that fulfill the unity criteria lead directly to the creation of communicative events with the same name. For example, processes P3 and P6 lead to communicative events PHO2 and PHO4, respectively. The IDs and names of communicative events can

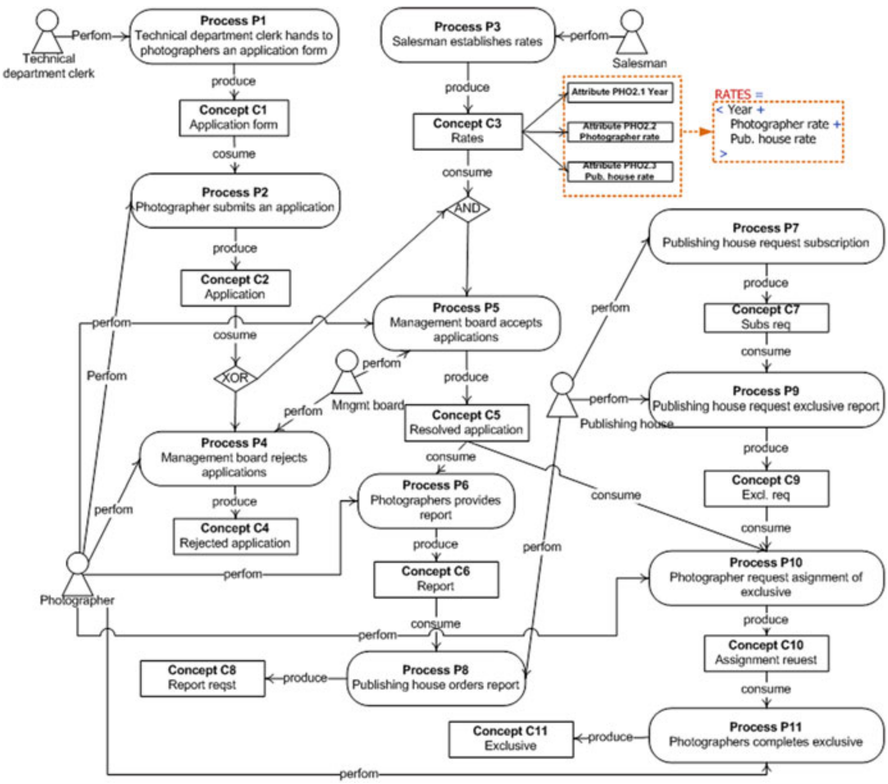


Fig. 5 BPM representing part of Photography Agency Inc.

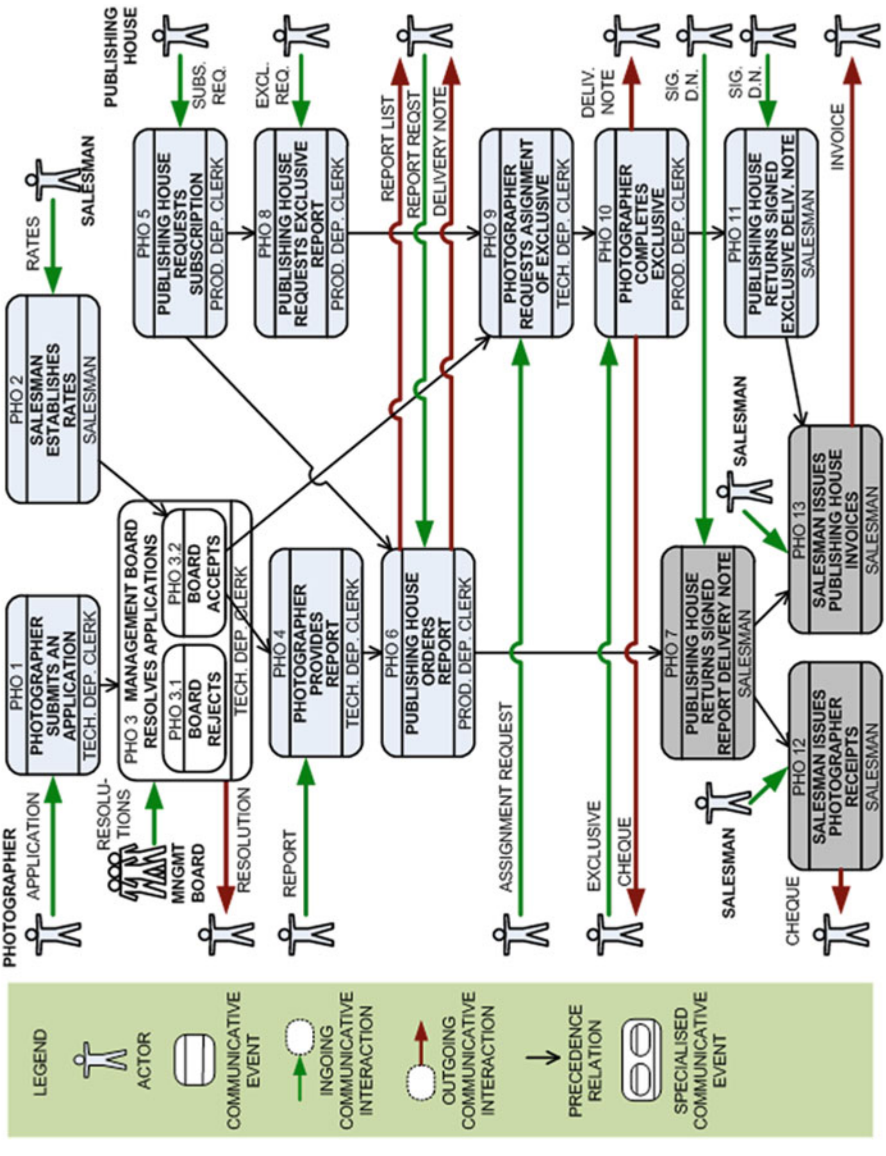


Fig. 6 CED representing part of Photography Agency Inc.

correspond with their process counterparts, but correspondence is not mandatory because not all processes are transformed into communicative events (*Guideline 2*).

Guidelines *5* and *6* advise the identification of communicative roles. For instance, "photographer" is identified as an information provider to PHO1 and hence as a primary role. Consequently, an ingoing interaction is created towards PHO1. Similarly, other roles are created along with their corresponding interactions.

Precedence relationships among communicative events are derived following *Guideline 7*. The IMR::Produce relationship between process P5 and concept C5 and the IMR::Consume relationship between concept C5 and process P6 lead to a precedence relationship between the communicative events PHO3.2 and PHO4.

The MS in communicative interactions contain the reference fields that describe concepts used in BPM, as defined in *Guidelines 9* and *10*. For instance, the concept C3 corresponds with the MS "rates" which is associated to the ingoing interaction of the PHO2. According to *Guideline 11*, data fields of MS are derived from the concept attributes in CM. Figure 5 (in red) shows the derivation of the data fields "year," "photographer rate," and "pub house rate" corresponding to the MS "rates."

6 Related Work

The inability of UML to capture organizational goals is highlighted in [12, 13]. As a solution, Santander and Castro [13] propose a set of guidelines for deriving use case models from i* organizational requirements [14]. Use case actors are identified from i* organizational actors, and use cases are derived from the dependencies between actors and other components in the i* Strategic Dependency model. Use case scenarios are discovered by analyzing actor relationships in the i* Strategic Rationale model. Using the unifying meta-model offers added benefits over i*, specifically regarding the providing a holistic view enabled by IMRs. On the other hand, CA has a more detailed description of processes compared to use cases.

Another emerging line of research relies on value models to capture the intentional aspects of process modeling. For instance, Weigand et al. [15] propose deriving BPMN [16] process models from e3-value models [17] that describe the value objects exchanged between actors. Value modeling captures motivation in terms of values, but it lacks the holistic view offered by the unifying meta-model.

Goal-Oriented Requirements Engineering (GORE) is an approach whereby requirements are elicited by analyzing organizational goals [18]. For instance, Koliadis and Ghose [19] developed techniques for creating traceability and satisfaction links between KAOS models [12] and BPMN models. Traceability links assert the intentional relationship between goals and process activities. Satisfaction links are realized using annotations in the BPMN model. The survey in [7] shows how traceability-based integration between requirement models and other MDD models, among other integration approaches, does not constitute a holistic MDD approach.

7 Conclusions and Future Work

In this paper, guidelines for aligning CA with the unifying meta-model were proposed. The unifying meta-model provides the required organizational context for CA, while CA offers a concrete instantiation of BPM and CM. Differences exist between the approaches, both conceptually and practically, complicating the application of the transformation guidelines and calling for analyst intervention.

Application of the unity criteria in CA is a decision made by the modeler, considering the overall system design and the way that functionality is distributed in processes. This implies that the alignment between BPM and CED is not isomorphic and requires further analysis when utilizing the guidelines. The implications of this reasoning are evident in many of the proposed guidelines.

The solution presented herein is a first step towards an integrated EM-driven MDD approach that provides a concrete holistic view and a methodology for developing IS for enterprises. An integrated process model that builds on CA is being developed as part of the unifying meta-model to eliminate the need for transformations. The proposal can be extended with patterns to capitalize on best practices and available solutions. Furthermore, CA includes guidelines for creating the modeling concepts and gathering related information. The guidelines form a method which the unifying meta-model lacks but is necessary for a complete MDD approach [7]. Explicit modeling guidelines for the unifying meta-model are being investigated. An action research case study is planned as part of our future work to test the integration solution in a realistic setting. A lab demo demonstrates the usefulness of the proposed guidelines, but a case study will identify shortcomings and help rectify them and improve the solution.

References

1. Bubenko JA Jr, Persson A, Stirna J (2001) D3: user guide of the knowledge management approach using enterprise knowledge patterns. Royal Institute of Technology (KTH) and Stockholm University, Stockholm, Sweden
2. Selic B (2003) The pragmatics of model-driven development. IEEE Soft 20:19–25
3. Object Management Group (OMG) Unified Modeling Language. http://www.uml.org/
4. Object Management Group (OMG) Model Driven Architecture. http://www.omg.org/mda/
5. Pastor O, Molina JC (2007) Model-driven architecture in practice: a software production environment based on conceptual modeling. Springer-Verlag, Berlin
6. España S, González A, Pastor O (2009) Communication analysis: a requirements engineering method for information systems. In: CAiSE 2009, LNCS 5565, 2009. Springer Berlin/ Heidelberg, pp 530–545
7. Zikra I, Stirna J, Zdravkovic J (2011) Analyzing the integration between requirements and models in model driven development. In: EMMSAD 2011, LNBIP 81, 2011. Springer Berlin Heidelberg, pp. 342–356
8. Zikra I, Stirna J, Zdravkovic J (2011) Bringing enterprise modeling closer to model-driven development. In: PoEM 2011, LNBIP 92, 2011. Springer, pp 268–282

9. España S (2011) Methodological integration of communication analysis into a model-driven software development framework, PhD Thesis, Departamento de Sistemas Informáticos y Computación (DSIC), Universitat Politècnica de València, Spain, 2011
10. González A, España S, Ruiz M, Pastor O (2011) Systematic derivation of class diagrams from communication-oriented business process models. In: BPMDS 2011, LNBIP 81. Springer Berlin Heidelberg, pp 246–260
11. España S, González A, Pastor O, Ruiz M (2011) Integration of communication analysis and the OO-method: manual derivation of the conceptual model. The SuperStationery Co. Lab Demo. CoRR 1101
12. Castro JF, Alencar FMR, Filhol GAC, Mylopoulos J (2001) Integrating organizational requirements and object oriented modeling. In: RE 2001, IEEE, pp 146–153
13. Santander VFA, Castro JFB (2002) Deriving use cases from organizational modeling. In: RE 2002. IEEE, pp 32–39
14. Yu ESK (1995) Modelling strategic relationships for process reengineering, PhD Thesis, University of Toronto, Toronto, Canada
15. Weigand H, Johannesson P, Andersson B, Bergholtz B, Edirisuriya A, Ilayperuma T (2007) Value object analysis and the transformation from value model to process model. Enterprise interoperability 2007, part II. Springer, London, pp 55–65
16. Object Management Group (OMG) Business Process Modelling Notation. http://www.bpmn.org
17. Gordijn J, Akkermans JM, van Vliet JC (2000) Value based requirements creation for electronic commerce applications. In: Proceedings of the 33rd Hawaii international conference on system sciences (HICSS-33). IEEE, 4–7 Jan 2000
18. Dardenne A, van Lamsweerde A, Fickas S (1993) Goal directed requirements acquisition. Sci Comput Program 20:3–50. North Holland
19. Koliadis G, Ghose A (2006) Relating business process models to goal-oriented requirements models in KAOS. In: PKAW 2006, LNAI 4303. Springer, pp 25–39

Index

A
AAA Bank, 487–488
ACBI. *See* Australian Centre for Broadband Innovation (ACBI)
Activity theoretic systems architecture (ATSA)
agnostic implementation, 78
analysis, 79–81
coffee machine problem, 79, 82, 83
construction, 79–80
description, 77–78
initial design, 78
and OO (*see* Object orientation (OO))
redesign, 81
Affordances
crowdsourcing, 512
large-scale ecological monitoring, 514
VWs
communication, 549–550
interaction, 551, 552
novice users, SL, 549
PWLD, 553, 554
rendering, 550–551
Agile software development (ASD)
adaptation, 426–427
autonomous agents, 425
beyond budgeting (*see* Beyond budgeting)
CAS characteristics, 422, 424–426
coevolution, 427
complex adaptive situations, 423–424
customer proxy, 304
edge of chaos, 428
energy exchange, 427
environment, 304
evolution, budgeting, 301–302
feedback processes, 426
interactions, 425–426
ISD projects, 422
legacy processes, 308
nonlinear dynamics and emergent behavior, 426
principles, 423
self-organization, 427
traditional software lifecycle, 423
Agile systems development
analytical lens, 573, 574
data sources, 573, 574
decision-making
and IT, 464
parameters, 460
efficiency, recommendations, 579–580
factors, claims and rebuttals, 577, 579
identification, claims and qualifications, 577, 578
impacts
decision quality, 465
excessive polychronicity, 465
time pressure, 464
impression, balance and harmony, 577
in-depth exploration, user requirements, 573
integration, 463
ISD, 571
non-modifiable factors, 580
product, process and experiential perspectives, 572
situational factors and claims, 578–580
and software, 572–573
synchronisation, polychronicity and timeliness, 462–463
temporal structuring, 460
time pressure, 461–462
Toulmin's model depiction, 580

H. Linger et al. (eds.), *Building Sustainable Information Systems: Proceedings of the 2012 International Conference on Information Systems Development*,
DOI 10.1007/978-1-4614-7540-8, © Springer Science+Business Media New York 2013

Agile systems development (*cont.*)
  and UCD, 573
  usability literature, 572
  user-centered design, 572
  work execution, 575–576
  work organization, 576–577
AMCIS. *See* Americas Conference on Information Systems (AMCIS)
Americas Conference on Information Systems (AMCIS), 587
AQUA-Web-Services (AQUA-WS), 192
Architecture
  ATSA (*see* Activity theoretic systems architecture (ATSA))
  CIMOSA, 375
  collaborative (*see* Collaborative architecture)
  enterprise (*see* Enterprise architecture (EA))
  evolving system of systems
    community group and services, 198–199
    layers, 197–198
    relationships, 198
  framework, C4ISR, 374
  product line, 212, 216
ASD. *See* Agile software development (ASD)
ATSA. *See* Activity theoretic systems architecture (ATSA)
Automotive industry
  application engineering
    architectural transformations, 216
    calculation, transformation process, 215–216
    fault tree analysis (*see* Fault tree analysis, automotive industry)
    key node safety metric calculation, 216–217
    product line architecture, 212, 216
    QVT-relations, 216
    relationships, 217–218
  domain engineering phase
    homogenous redundancy pattern, 214
    product architecture derivation process, 214
    relationships, 215
    TMR, 214
    trade-off analysis, 215
  features, 213–214
  inner structure, product line architecture, 213, 214
  VCS, 213

B
Banking and financial services, 67
Beyond budgeting
  agile methods, 302–303
  ASD environment, 303
  description, 302
  modern enterprise, 301
  systems development, 301
  traditional command-and-control management model, 302
BPEL. *See* Business process execution language (BPEL)
BP-ISP integration. *See* Business planning and IS planning (BP-ISP)
BPM. *See* Business process model (BPM)
BPMN. *See* Business process modelling notation (BPMN)
BPMS. *See* Business process management system (BPMS)
Business and information system alignment
  balancing alignment., 48
  definition, 42
  description, 41, 48
  exploration and exploitation, 45, 47
  framework, 46, 47
  knowledge creation, 45
  levels, 44
  management literature, 45
  mechanisms and enablers, 43–44
  performance, 47
  planning, 45
  strategy and structure mode, 42–43
Business and IT alignment (BIA). *See* Higher education, BIA
Business planning and IS planning (BP-ISP), 52, 59–60
Business process (BP) enactment plans
  design and analysis phase, 432
  generation, 436, 441–442
  management, 431–432
  quality, 432
  recommendations, 441
  simulation, 440–441
  time prediction, 441
Business process execution language (BPEL), 70
Business process management system (BPMS), 288
Business process model (BPM)
  CED, L2, 600
  Photography Agency Inc., 606
  production and consumption, 603

transformation guidelines, 604–605
unifying meta-model, 599
Business process modelling notation (BPMN), 255, 258

C

CA. *See* Communication analysis (CA); Competitive advantage (CA)
CAIS. *See* Communications of the Association for Information Systems (CAIS)
CAS. *See* Community Children Academy Spaces (CAS)
CIMOSA framework. *See* Computer Integrated Manufacturing Open System Architecture (CIMOSA) framework
C4ISR architecture, 374
Class life cycle model
business processes, 446, 447
conceptual system, 445–446
description, 447–448
hierarchical abstraction, 467
impacts, EA development, 445
motivation, 446–447
structure, 447
system *vs.* process
additional valid life cycle, 452–453
Cartesian product, 453
class customer, 448–449
conditions, 449
conflicting phase, class customer, 451, 453
generic class customer, 450, 451
hierarchical classification, 448
methodological construction, 454
object-oriented approach, 450
solution, conflict phases, 453–454
specific—active—type generic customer, 450–451
subtypes, 450
CM. *See* Concepts model (CM)
CNs. *See* Collaborative networks (CNs)
CobiT. *See* Control objectives for information and related technology (CobiT)
Cobweb graph, 393
Cognitive load theory (CLT)
definition, 131
ECL, 132
Germane load (GL), 133
and HCI, 135
ICL, 132
instructional design, 133–134
interaction and limitations, 131
knowledge reconstruction, 131–132
LTM and WM, 131
Collaborative architecture
communication system, 199, 200
consistency, design subsystems, 200
environment, 200, 202–203
individual systems, 199
information subsystems, 203
representation, business networking, 200, 201
sustainable collaboration, 200
system of system, 199–200
Collaborative networks (CNs)
disaster management, 357–358
global business environment, 355
COMET system, 437
Commercial off-the-shelf (COTS) software, 250, 253
Commonwealth Scientific and Industrial Research Organisation (CSIRO), 104
Communication analysis (CA)
alignment, unifying meta-model (*see* Unifying meta-model)
CA meta-model, 600, 601
description, 600
OO-method conceptual models, 600
Communications of the Association for Information Systems (CAIS), 587
Community Children Academy Spaces (CAS), 348–349
CompCul
agile principles, 8
core cultures, 8, 9
establishment, 7–8
project and management, 8
structure, 8
Competitive advantage (CA)
definitions, 264
description, ERP system, 268–271
end-user organizations, 265, 266
ERP and stakeholders, 262–264
and ICTs, 261
organization, 265
propositions, 266–268
resource-based view claims, 265
software applications, 264
vendors and resellers, 265, 266
Complex adaptive systems (CAS)
autonomous agents, 420–421
characteristics
ASD methods principles, 424
description, 421–422
ISD projects, 422
mapping, 424–425

Complex adaptive systems (CAS) (*cont.*)
  complexity, 420
  emergent organizational behavior, 421
  formal theory, 421
Computer Integrated Manufacturing Open System Architecture (CIMOSA) framework, 375
Concepts model (CM)
  binary relationships, 604
  Photography Agency Inc., 606
  unifying meta-model, 599
ConDec-R process model
  approaches, 434, 435
  BP enactment plans, 436
  COMET system, 437
  CSP solution, 437
  filtering rule, 436–437
  Gantt chart, 437
  generic ConDec models, 437, 438
  NP complexity, 437
  OptBPPlanner tool, 434, 436
  scheduling and repeated activity, 436
  specification, 436
  templates, 433
  tuples, 434
Constraint optimization problems (COPs), 434, 437
Constraint programming (CP), 433, 434
Constraint satisfaction problem (CSP)
  and C_{CSP}, 433, 437
  and COPs, 434
  solution, 434, 435
Control objectives for information and related technology (CobiT), 279–281
CoolStreaming/DONet, 314
CoopCul, 7
COPs. *See* Constraint optimization problems (COPs)
Corporate sustainability, 376–377
COTS software. *See* Commercial off-the-shelf (COTS) software
CP. *See* Constraint programming (CP)
Crowdsourcing
  collective intelligence, 513
  information technology, 512–513
  micro-tasks, 513
  and motivational affordances, 512
  observations, 513
CSIRO. *See* Commonwealth Scientific and Industrial Research Organisation (CSIRO)

D
DaemonX, 404, 417
Data and information quality (DQ/IQ) research
  backward and forward (Bw/Fw) searches, 588
  challenges, 587, 593
  connections, identified challenges and level, 589, 590, 593, 594
  description, 585
  framework
    AM, 592
    assessment, 587
    tools, 592
  implications, 594
  iterative nature, concurrent engineering, 589
  lack of integration
    between processes, 591
    between systems, 591
    timely information, 592
  literature reviews and databases, 587, 589
  outcomes, 588, 589
  publication outlets, 587
  researchers, 587
  search terms, IT/IS outlets, 588
  technical and nontechnical issues, 586
  uniqueness, engineering data, 589, 591
Database Administrators (DBAs), 498–500
Database intrusion detection systems (DIDS)
  actions, 495–496
  alert correlation techniques, 501
  analysis, 499
  architecture, 499, 500
  business knowledge and decision making, 496
  commercial DBMS, 497–498
  contributions, 496
  DBA, 500
  detection
    anomaly, 505–506
    misuse, 504–505
  end user workload, 507
  false-positive (FP) and true positive (TP) rate, 508
  features and assumptions, 508–509
  functions, IRM, 500–501
  joint functionality, 496, 497
  monitoring, 499
  outcomes, generated alerts, 508
  probabilities assessment, 502–503
  risk exposure assessment, 497, 501–502
  security administrator rules, 503
  security *vs.* intrusions, 498–499

TPC-H benchmark, 506–507
web-based infrastructures, 496
Data warehouse conceptual models
approaches and development, 533–534
Cube design, 534
implementation, 543
information requirements, 534
literature, 535
metadata, evolution support
framework, 537, 538
logical, 538
mapping, 538
reporting and management tool, 538–539
repository, 537, 538
semantic, 538
tools, 537
metadata management solutions, 535
motivating (*see* Motivation, data warehouse schema)
query-driven method (*see* Query-driven method, data warehouse schema)
query usage study, 539–540
Declarative process models, 432, 434
Design science research (DSR), 250, 258
Design science research methodology (DSRM), 156–157
Developing countries
development, Indonesia, 558
IS
local government, 557
sustainability, 559
Digital Doorways, 345
Digital literacy
broader life outcomes measures, 111
cross-sectional benchmark survey, 110–111
definition, 109
internet-related behaviour, 109
measures, 109
senior Australians
attitudes, 97–98
social media, 91–92
telephone interviews, 109–110
validation, 110
Disaster management breeding environment (DMBE), 361–362
Disaster management organisations (DMOs)
business survival and success, 353
centralisation/federalisation, 354
CNs, 357–358
commercial enterprises, 355
emergency services, 354
environmental sustainability, 353
interoperability, IS (*see* Information systems (IS))
life cycle and EA, 358–361
organisational diversity, 354
virtual enterprise and commercial breeding environment, 363
Disaster management project (DMP), 360, 362
Disaster management qualification and training organisation (DMQT), 362
Disaster management task forces (DMTFs), 361–362
DMBE. *See* Disaster management breeding environment (DMBE)
DMOs. *See* Disaster management organisations (DMOs)
DMP. *See* Disaster management project (DMP)
DMQT. *See* Disaster management qualification and training organisation (DMQT)
DMTFs. *See* Disaster management task forces (DMTFs)
DQ/IQ. *See* Data and information quality (DQ/IQ) research
DSR. *See* Design science research (DSR)
DSRM. *See* Design science research methodology (DSRM)
DYA model. *See* Dynamic architecture (DYA) model
Dynamic architecture (DYA) model, 375

E

EA. *See* Enterprise architecture (EA)
ECIS. *See* European Conference on Information Systems (ECIS)
ECL. *See* Extraneous cognitive load (ECL)
Education
EduMON (*see* Education d/o information monitoring system (EduMON))
EFA (*see* Education for all (EFA))
ICT, 130–131
information
challenges, 470, 479
EduMON, 472–473
institution, 470
monitoring (*see* Monitoring system, education information)
Web crawler, 480
software, 135
and training, SCC model, 124

Education d/o information monitoring system (EduMON)
  architecture, 472
  challenges, 479
  degrees, service automation, 476–478
  design classes, 473
  dynamic and systemic framework, 472
  functionality and implementation, 473
  management changes and notification services, 478
  responsible user, 479
  revisited architecture, 472
  SOA approach, 472
  source and processing information, 472–473
  textual representation and diversity of information, 474–476
  World Wide Web, 473
Education for all (EFA)
  barriers, 347
  CAS, 348, 349
  home schools, 349
  The Millennium Development Goals (MDGs), 346
  social inclusion, 346–347
  systems approach, 347
EFA. *See* Education for all (EFA)
Ego network, 328, 331
E-health, Australia
  applications, 92–93
  developments, 92
  government report, 92
  qualitative pilot project, 93–94
  social media collaborations, 93
E-learning systems
  CLT (*see* Cognitive load theory (CLT))
  description, 129–130
  designing, 129, 138
  development, 130
  and HCI (*see* Human–computer interaction (HCI))
  ICT, education, 130–131, 138
  learners, 138
  semantic memory, 130
ELISA. *See* Extensible layer for internet services (ELISA)
EM. *See* Enterprise modeling (EM)
Emergence concept. *See* Information systems (IS)
Enterprise architecture (EA)
  areas, 524
  coherent enterprise, 524–525
  communication tool, 371
  conceptual framework, IEEE 1471, 370, 371
  definition, 358, 522
  description, 521–522
  determination, 522–523
  DMP, 360
  DYA model, 375
  evolution and agility, 358
  FEAF, 373–374
  framework
    application behavior and layers, 526–528
    ArchiMate 2.0, 529–530
    ArchiSurance, 526
    CIMOSA, 375
    C4ISR, 374
    coherent description, 526, 528
    communication, 523
    criteria assessment, technology layer, 526, 527
    DoDAF 2.02, 529
    features, 523
    ISO/IEC/IEEE 42010 standard, 524
    language ArchiMate, 526
    planning changes, 523–524
    service viewpoint, 526, 527
    standard software intensive systems, 523
    TOGAF 9.0, 529
    Zachman v2.1, 529
  GERA MF, 358–360
  harmonization, 530
  human ecological support systems, 367
  ICT systems, 371–372
  impacts, globalization, 521
  integration, 522
  IS interoperability requirements, 358, 360
  and ITSM (*see* IT service management (ITSM) and EA)
  MODAF, 374–375
  outcomes, service-oriented models, 530
  principles, corporate sustainability, 376–377
  response phase, data and process IS, 361
  risk-monitoring mechanisms, 377
  selection, 524
  service orientation
    cohesion enterprise, 524–525
    frameworks, 525–526
    supporting, 522
  sustainability
    development, 368–369
    strategy, 370
  systematic activities, 370
  TEAF, 374
  TOGAF, 372–373
  and WSM, 529
  ZF, 373

Enterprise modeling (EM)
CM and BPM views, 599
complimentary views, 599
description, 597
high-level IS requirements, 598
MDD, IS development approach, 598
UML and MDD, 598
unifying meta-model, 598
Enterprise resource planning (ERP)
BPMN and UML, 258
"business blueprint and task definition", 257
CA (*see* Competitive advantage (CA))
COTS, 250
current and planned part, 256
current scenarios, 254–255
customer implementation, 270
data collection, users, 253–254
development
end-user organizations, 267–268
scenarios, 255
stakeholders, 264
win-win situation, 268
DSR, 250, 258
ISD, 249
requirements
BPMN and UML, 255
current scenarios, 254–255
data collection, users, 253–254
developed scenarios, 255
documentation, 257
identification, 251–253
software vendors, 254
specification, 255–256
resource-based view, 264–266
value chain and stakeholders, 262–264
vendors (*see* Vendors)
win/lose relationship, 268
win-win situation, 269
eProcurement
adoption, Indonesian government, 558
collective awareness and cultivate legitimacy, 566
community of practices, 565
contributions, 566–567
government II, legal factors, 564
information infrastructure, 559–561
intensive links, dimensions of II and factors, 561–564
interviews, 560
IS, 557
JAD workshop, 566
legal and managerial factors, 561
"legalized", 565
in LKPP, 565–566
National Public Procurement Agency, 558
problems, 557
public procurement expenditure, 558–559
socio-technical artifact, 558, 560
subjects and data, 560
sustainability, ISs, 559
system to combat corruption, 559
triple-helix model, government II, 564, 565
ERP. *See* Enterprise resource planning (ERP)
Escalation theories
accentuate erratic decisions, 381
3D project, 383
project, psychological, social and structural perspective, 381
risk management, IS, 380
European Conference on Information Systems (ECIS), 587
Evolving system of systems
architecture (*see* Architecture)
business networks, 195–196
cloud implementation, 204
description, 196
design, complex environments, 196–197
emergence, 203–204
Enterprise 2.0, 195
management, 205
mapping, 203, 205
software engineering requirements, 196
Extensible layer for internet services (ELISA)
analysis and design
API network, 315–316
architecture schema, 315, 316
client application, 316
concrete plug-in, 315
interaction, 315, 316
plug-in manager component serves, 315
P2P (*see* Peer-to-peer (P2P) network)
client-server architecture, 311
communication services, 312
computer network, 312
construction, P2P network, 314
CoolStreaming/DONet, 314
manycast
ambiguous receivers, 319
distinct receivers, 319–320
implementation, 318
message integrity, 322
multiple architectural proposals, 312
NEBLO, 314
performance comparison, time to delivery, 320–321
pseudo-random Pastry key, 313
SAAR, 313
security comparison, 320, 321
SplitStream, 313
sybil and *eclipse* attack, 315

Extensible Markup Language (XML). *See* XML query adaptation
Extraneous cognitive load (ECL), 132

F
Fault tree analysis, automotive industry
  tolerance/latency, 216, 217
  with and without TMR pattern, 216, 218
FEAF. *See* Federal Enterprise Architecture Framework (FEAF)
Federal Enterprise Architecture Framework (FEAF), 373–374

G
Generalised Enterprise Architecture and Methodology (GERAM), 358
Geocoded national address file (G-NAF), 110
Geographic information system (GIS), 110
GERA. *See* GERAM's reference architecture (GERA)
GERAM. *See* Generalised enterprise architecture and methodology (GERAM)
GERAM's reference architecture (GERA), 358–360
Germane load (GL), 133
GIS. *See* Geographic information system (GIS)
GL. *See* Germane load (GL)
Global development
  active learning, 343
  concept, 341
  developing countries, 349
  distinctive culture, 341
  economic, environmental and social development, 340
  EFA, 346–349
  ICT4D and active learning, 344–346
  organic way, living, 342
  service-dominant logic approach, 340
  social changes, 339
  strong identity, 341–342
  "support oneself", 342
  sustainable development and information systems, 343–344
G-NAF. *See* Geocoded national address file (G-NAF)

H
HBI. *See* High-bandwidth internet (HBI), Australia
HCI. *See* Human–computer interaction (HCI)
Heriot-Watt University information systems
  business schools, 222
  core and choice course, 231
  curriculum
    actual course structure 2008, 223–224
    BISD 2007, 223
    discipline, 224–225
    IS 2010 and IS 2002
    IS 2010 and IS 2002: courses and changes, 230
    IS 2010 and IS 2002: NSS, 229
    IS 2010 and IS 2002: personal productivity applications, 230
    IS 2010 and IS 2002: programmes, 229
    IS 2010 and IS 2002: skill and competency, students, 230
    IS 2002 and report, 223, 225
    professionals, 223
    technological and sociotechnical, 225
  employability, 225–226
  functional areas and management skills, 222–223
  methodologies, 222
  monitoring, 222
  past student satisfaction, SNS
    Facebook, 226–227
    guidelines, IS 2010, 227
    knowledge-gathering platform, graduates, 226
    qualitative data, 226
  perspectives, 222
  potential students, 228
  programmes, 222
  proposal, 221–222
  teaching quality, 231
  technical and sociotechnical subjects, 223
High-bandwidth internet (HBI), Australia
  consumer adoption and diffusion, technology innovations, 106
  control, 111
  and CSIRO, 104
  digital literacy and determinants (*see* Digital literacy)
  establishment, TPB, 106
  evolutionary framework, 112
  impacts, NBN, 112
  integration, human behaviour, 106–107
  intervention, 105
  measurements, broadband connectivity
    download and upload speeds, 108
    high-bandwidth, 108
    transfer rate, 108
    validation, 108–109
  national broadband network, 104

NONIE guidelines, 104
outcomes value, stakeholders, 105–106
qualitative and quantitative empirical research, 107
Higher education, BIA
BP-ISP integration, 52
common conceptual model, 54
contribution, 53
course validation processes, 52
IS and services, 52, 61
JISC S-ICT Toolkit, 54
limitations, 61
non-profit organisations, 61
on-line, 51–52
perception, 54–55
public funding, 51
research activities, 61
SICT Toolkit and components, 55–56
in UK, 52
validation
academic, 56
architecture integration and BP-ISP, 59–60
business processes, 56
ICT governance, 58–59
institutional maturity, 58
methodology, 57–58
reliability and correlation, 59
SICT, 58, 60–61
Human capital, 70
Human–computer interaction (HCI)
and CLT, 135
definition, 134
design, 134–135
element interactivity, 136–137
evaluation, educational applications, 135
integration, 138
interaction devices, 135
learning process, 135
mid-1980s, 135
redundancy effect, 136
split-attention, 135–136
user-centered development, 135

I
ICIQ. *See* International Conference on Information Quality (ICIQ)
ICIS. *See* International Conference on Information Systems (ICIS)
ICL. *See* Intrinsic cognitive load (ICL)
ICT4D. *See* Information and communication technologies for development (ICT4D)
ICTs. *See* Information and communication technologies (ICTs)
Identity management systems (IdMS)
artifact, 32–33, 36
classification and function, 28
definition, 27
services, 27–28
and technological artifacts, 28–29
IdMS. *See* Identity management systems (IdMS)
Indonesian Public Sector. *See* eProcurement
Information and communication technologies (ICTs)
EA frameworks, 376
education, 130–131, 138
ICT4D (*see* Information and communication technologies for development (ICT4D))
sociocultural diversity, 369
strategy, 370–371
sustainability, 369
Information and communication technologies for development (ICT4D)
"active learning", 344
computer games, 345
Digital Doorways, 345
global development, 340
learning processes, 346
mobile phone laboratory, 345
multimedia studios, children, 345
"smart home" technology, 345
Information infrastructure, eProcurement, 559–561
Information systems (IS)
interoperability
acceptable and desirable, 355
clarification, 355
"co-habitation", organisations, 357
data and process areas, 356
disaster management (*see* Disaster management)
DMBE, 362
DM projects, 361
DMQT, 362
integration and federalisation, 356
knowledge and ISO14258, 356
organisational structure diversity, 356–357
population, 362
pragmatic aspect, 356
requirements, 361, 362
task force, 356
risk management
accentuate erratic decisions, 381
communication dimension, 384–385
consequences and causes, 380–381
deaf effect, 385
3D project escalation, 383

Information systems (IS) (*cont.*)
emotional attachment, 385, 386
framing issues, 385
investigating, 386
mum effect, 386
perspectives, escalation, 385
potential risk, 380
problems, communication, 381
project success rate, 379
psychological dimension, 383–384
research methodology, 382
user expectations and delivery, 380
Information systems development (ISD)
agile methods, 420, 428–429
and ASD methods (*see* Agile software development (ASD))
autonomy, 305
beyond budgeting (*see* Beyond budgeting)
and CAS (*see* Complex adaptive systems (CAS))
CHAOS Report, 299–300
complexity and management perspectives, 420
control, 307
coordination, 308
customers, 304
evolution, budgeting, 301–303
Geneca's survey, 419–420
goals, 306
language, 458
organization, 304–305, 420
planning, 307
practitioners, 420
quality, human life, 457–458
research methodology
case study database creation, 303
chain of evidence, 303
confidentiality agreement, 303
data analysis, 304
multiple sources, evidence, 303
resources, 307
responsibility, 305
rewards, 306–307
theoretical justification, 428
time-consuming, 458–459
traditional budgeting process, 300
transparency, 306
turbulent business environments, 419
values, 306
Western culture, 458
Information technology (IT)
governance implementation, Thailand
AAA Bank, 487–488
alignment strategy, 489–490
business growth, 484
definition, 484
effectiveness, 491
frameworks, 488
investment, 488–489
management, business, 484–485
organization and management, 483–484
research, 486–487
senior IT strategy, 4888
strategic alignment, 485
value delivery, 485–486, 490–491
ITSM and EA (*see* IT service management (ITSM) and EA)
management
categorisation, 275–276
communication and interaction, public, 284
environmental problem, 274
frameworks, governance, 284
individual characteristics, 282
integrative approach, 285
organisational structures, 284
shareholder value and profit gain, 273
sustainability (*see* Sustainability, IT management)
TBL, 274
western countries, 274
outsourcing industry, 63–64
Information technology infrastructure library (ITIL), 281–282
Instantiating design principles
autonomy and self-identity, 515–516
competence, 516–517
mobile phone characteristics and human needs, 514–515
relatedness, 517
Intellectual capital (IC), 66
Interaction design, agile contexts, 580
International Conference on Information Quality (ICIQ), 587
International Conference on Information Systems (ICIS), 587
Intrinsic cognitive load (ICL), 132
Intrusion Response Manager (IRM)
DIDS-DW, 500–501
probability and impact rules, 505
IS. *See* Information systems (IS)
ISD. *See* Information systems development (ISD)
ITIL. *See* Information technology infrastructure library (ITIL)
IT service management (ITSM) and EA
activity identification, 162
aggregation, 161
alignment approach

and EA, 157
ITIL, 157–158
relationship, 159
SOA, 158
artifacts and components, 160
demonstration, 163–164
and DSRM, 156–157
evaluation, 164–165
frameworks, 156, 162–163
governance, 155–156, 165
graphical representation, 161
implementation, 162
infrastructure, 155
initiatives, 160
integration, 163, 165
meta-model, 163, 164
modulation standards, 165
organizations, 155
parlance, 160
product/service, 162
service-level agreement (SLA), 162
IT service management framework (ITIL), 157–158
ITSM. *See* IT service management (ITSM)

J
JAD. *See* Joint-application development (JAD)
Joint-application development (JAD), 566

K
K-networks and innovation-driven collaboration
centrality, 331
content, 326–327
data collection, 329–330
definition, 329
density, 331
description, 325–326
ego network, BF94, 328
exploring, interactions
configuration, 334–335
dense networks, 334
heterogeneous knowledge, dense networks, 332
innovation network, 332
statistical tests, 332, 333
structural exploration process, 332
firms
exploration and exploitation process, 330–331
knowledge heterogeneity, 330
IDC manufacturing insights, 326
intra-organisational aspects, 326, 336
possible structural configurations, 331–332
process, 327
quadrant framework, 335
structure, 327–328
Victorian biotech industry, 329

L
Large-scale ecological monitoring
awareness, 518
“clean marine”, 512
crowdsourcing (*see* Crowdsourcing)
impacts, humans, 511–512
instantiating design principles (*see* Instantiating design principles)
motivation
affordances, 514
participation, 513–514
projects, 515
Long-term memory (LTM), 130–132
LTM. *See* Long-term memory (LTM)

M
MDD. *See* Model-driven development (MDD)
MDETest, 192
MDSD. *See* Model-driven software development (MDSD)
Meeting scheduler system (MSS), 147–148
MeTAGeM-Trace, 239–240, 242, 244
Methodology for design, 287
Mindfulness, ISD projects
coarser plan, 20
collective, 18
component, 20
individuals and interaction, 17, 19
openness, 17
pair programming, 19
preoccupation, 18
procedure perceive changes, 17–18
self-organisation, 16
turbulent environments, 19
Ministry of Defense Architectural Framework (MODAF), 374–375
MODAF. *See* Ministry of Defense Architectural Framework (MODAF)
Model-driven development (MDD)
CA methods, 600
coding, 237
Eclipse and EMF, 239
integrated EM-driven, 609
inter-modeling specifications, 239
IS development approach, 598
mapping, Families2Persons case, 242–244
MeTAGeM-Trace, 242

Model-driven development (MDD) (*cont.*)
metamodels, 242
NDT-quality, 238
pattern-matching parts, 244
purely imperative/graph-based, 245
relationships, metaclasses, 242, 243
software
artifacts, 235
development, 236
traceability-based integration, 608
unifying meta-model, CM and BPM, 598, 599
Model-driven engineering (MDE), 170, 178
Model-driven software development (MDSD), 236, 244
Model transformations
ATL, 238, 243
traces generation
lower-levels, 237
MeTAGeM-Trace, 239, 240
methodological proposal, 239–241
supporting, 236, 239–241
technical support, 241
Monitoring system, education information
architectures, 471
challenges, 479
components, 471
design, 471
educational institutions, 471
EduMON, 472–473
functionality, 469
learning outcomes and job advertisements, 478
Motivation
class life cycle model, 446–447
crowdsourcing, 512
data warehouse schema
extraction, transformation and loading processes, tables, 536–537
potential dimension, 543
source elements, 535–536
students (*see* Student motivation)
large-scale ecological monitoring, 514
MSS. *See* Meeting scheduler system (MSS)

N

National Public Procurement Agency, 558
Navigational development techniques (NDT)
AQUA-WS project, 192
concrete syntaxes
functional requirements, 188, 189
test artifacts, 188, 190–191
definition, 182
description, 181
functional requirements, 182
implementation, 191
MDETest, 192
metamodels
artifacts, 186
elements, 185, 186
functional requirement, 185
roundstrip technique, 185, 186
test values, 185, 186
tracing relationships, 186–187
and transformations, 191
metrics, QVT-operational code, 191
process, 182
quality, 238
round-strip strategy, 187
software projects, 191
T2 and T3 relation, 188
test cases generation
functional requirements, 183
techniques, 184
transformations, 187, 188
web development requirements, 181
NDT. *See* Navigational development techniques (NDT)
NDTQ-framework
definition, 175
guarantees, 178
interrelationship, 179
levels and typologies, 174
participants, 178
quick and direct access, 178
requirements engineering, 176–177
research, 178
software development processes, 175–176
NEarly BLind Overlay (NEBLO), 314
NEBLO. *See* NEarly BLind Overlay (NEBLO)

O

Object orientation (OO)
adaption, 87
classes identification, attribution and methods, 85
clustering the activity and rules, 87, 88
coffeemaker, Model-0 Class image, 85–86
description, 78
elements, 82
heuristics
application, 82
commonality process, 84–85
proto-classes, 84
“sink” activities, input instruments, 84
“source” activities, output instruments, 84

unclustered activities, associated instruments, 85
variations, 83, 84
redesigned activities and transactions, 82
techniques, 87
OMS. *See* Operations management system (OMS)
Online social networks (OSN)
Actors' profiles, 391
information retrieval methods and graph processing, 392
profile matching, 395–399
PSN, 389–390
User's profiles matching, 390
web applications, 389
world social structures, 390
OO. *See* Object orientation (OO)
Operations management system (OMS), 21, 22
OptBPPlanner
activity durations and resource availabilities, 440
business process (BP) (*see* Business process (BP) enactment plans)
and ConDec-R process model (*see* ConDec-R process model)
constraint-based process model, 433, 440
constraint programming, 433
and COPs, 434
and CSP (*see* Constraint satisfaction problem (CSP))
declarative model, 432
description, 442
empirical evaluation
data analysis, 439
experimental design and execution, 439
independent variables, 438
objects, 438
optimal solutions, 440
purpose, 437
response variables, 439
and LTL, 442
organization, 432
planning and scheduling, 432
traditional BPM life cycle, 432
OSN. *See* Online social networks (OSN)
Outsourcing
assumptions, 73
Australia company, 66–67
backsourcing, 74
banking and financial services, 67
businesses, 64
data collection and analysis
factors ranking across two companies, 68–69
participant distribution, 68
definition, post-outsourcing, 64
differences, 64
IC, 66
initiation, 73
IT industry, 63–64
post-outsourcing (*see* Post-outsourcing)
ranking, 72
relationship, 74
research method, 67–68
termination, contract, 65

P
Pacific Asia Conference on Information Systems (PACIS), 588
PACIS. *See* Pacific Asia Conference on Information Systems (PACIS)
Peer-to-peer (P2P) network
client-server systems, 312
distinct receivers, 317
frameworks, 312
key-based routing, 314
N-anycasts usage, 317
secure connections, 317–318
tree distribution usage, 317
People with lifelong disability (PWLD)
description, 546
mobility, 553
technology capabilities and affordances, 551, 552, 554
Photography Agency Inc.
BPM, 606
CED, 606, 607
Platform-specific model (PSM)
definition, 405
sample, 406
XPath query, 407
XSEM, 409, 417
Post-outsourcing
decision-making, 73, 74
human capital, 70
organisational capital, 71
physical capital, 71–72
social capital, 70–71
vendor and managers, 74
PP. *See* Professional profiles (PP)
P2P network. *See* Peer-to-peer (P2P) network
PRINCE2 processes, 282, 283
Process modelling
design process, 294
e-learning projects, 293, 294, 296
multimedia objects, instructional designers, 294, 295
plan and script author's requirements, 293

Process modelling (*cont.*)
  quality-oriented model, business processes, 294, 296
  surveys, 296
  verification, 293
Professional profiles (PP)
  definition, 392
  matching, 395
  profile management, 392–393
Professional social networks (PSN)
  actors' profiles, 391
  definition, 391
  information retrieval methods and graph processing, 392
  ontologies
    ACM, 394
    corpus-based methods, 393–394
    knowledge-based approaches, 394
  profile management, 392–393
  profile matching, 395–399
  structured information, 390
Profile management, 392–393
Profile matching
  categories, 395
  compatibility, 397–399
  definition, 396
  D2 software engineering, ACM, 396
  notion of distance, 395
  programming languages, 397
  score, 397
  types, 397
Project failure, ISD, 299
Projects and management
  author's experience and interviews, 290
  business processes, 290, 291
  categories, 289
  e-learning course development
    integration, 289, 290
    requirements analysis, 291, 292
  ISD
    adaptive system, 3
    advantages and disadvantages, 4
    CAS principles, 3–4
    co-evolution and agile development, 3, 4
    complex activity, 2
    description, 11
    emergent system behaviour, 2
    exploitation and exploration, 15–16
    interaction, 2
    interconnected autonomous agents, 2–3
    mindfulness (*see* Mindfulness, ISD projects)
    OMS, 21, 22
    performance, 21
    processes, 2, 21–22
    recommendations, 4
    and self-organisation, 3, 13–15
    time pacing, 3, 12–13
  positive impact, 292
  verification/raising quality, 291
PSM. *See* Platform-specific model (PSM)
PSN. *See* Professional social networks (PSN)
Public sector. *See* eProcurement
PWLD. *See* Virtual worlds (VWs)

Q
Quality-oriented modelling approach
  BPMS, 288
  integrated modelling approach, 297
  organisations' reorienting, 288
  process modelling (*see* Process modelling)
  project processes (*see* Projects and management)
  risk management modelling, 288
  UML, 287, 288
Query-driven method, data warehouse schema
  analysis, data modification and selection statements, 541
  conflict resolution, 542
  framework, design, 534–535
  potential development, 542
  preprocessing, information sources, 540
QVT-operational language, 191

R
RE. *See* Requirements engineering (RE)
Regional Initiative for Social Innovation and Research (RISIR), 104
Requirement descriptions. *See* Enterprise resource planning (ERP)
Requirements engineering (RE)
  information system, 141, 142
  SCC model
    creative and cognitive process, 123
    perception, analysts, 122
    professional, 125
    right tools and techniques, 123
Requirements modelling
  BPMN and UML, 255
  ERP business, 251–252
Requirements sources, 252–254
Resource-based view, 264–266
RISIR. *See* Regional Initiative for Social Innovation and Research (RISIR)
Risk emergence

data collection, 382
documentation, 382
3D project escalation, 383
quantitative research method, 382
thematic data analysis method, 382

S
SAO. *See* Service-oriented architecture (SOA)
Scandinavian Journal of Information Systems (SJIS), 588
SCC. *See* Social-creative-cognitive (SCC) model
Scheduling
BP activities, 439, 442
ConDec-R process model, 434, 437
CSP problem, 436
temporal and resource constraints, 433
Second life (SL), 548
SEM. *See* Structural equation modeling (SEM)
Service-level agreement (SLA), 162
Service-oriented architecture (SOA), 158, 471, 472
Service system, 472, 476
Six Sigma DMAIC project, 278–279
SJIS. *See* Scandinavian Journal of Information Systems (SJIS)
SL. *See* Second life (SL)
SLA. *See* Service-level agreement (SLA)
SOA. *See* Service-oriented architecture (SOA)
Social-creative-cognitive (SCC) model
commercial semi-object-oriented method, 118
confidential project, 118–119
consultant information, 118
custom-built systems, healthcare projects, 120–121
data analysis, 118
description, 115
evolutionary case research approach, 117
generic packaged software systems, stockbroking industry, 120
implications
agreement, 122
creative modelling skills, 121
education and training, 124
interaction, 122
relationships, 121–122
requirements engineering (*see* Requirements engineering (RE))
social process, 121
informal and formal, 121
object-oriented methods
actuarial and insurance systems, 120
booklets-based template, 119
fault management system, telecommunications organisation, 119–120
relationships, 125
social processes, 116
Social well-being, senior Australians
ageing population, 89
capabilities, experiences and needs, 94
co-evolving systems, 98–99
complex theory
description, 94
emergence processes and co-evolution, 98
guidance, 99
interventions, online communities, 99
description, 89–90
digital literacy (*see* Digital literacy)
e-health (*see* E-health, Australia)
emergent processes and co-evolution, 98
isolation and loneliness, 90–91
psychological adaptation, 99
research
online observations, 95–96
proposal, Aged Care Sector, 97
stages, 95
residential care, 89
traditional scientific method, 94
Software process management
BPEL, 170
BPM, 169–170
communication, 178–179
definition, 171
development, 170
engineering area, 172
enterprise architecture, 174
evolution, 170
guarantees, metamodel, 179
ISO/IEC 24744 and SPEM 2.0, 172
ISO standards, 171
life cycle, 171
MARVEL, 171
MDE, 170, 178
metamodel, 172–173
MODAL, 172
NDTQ-framework (*see* NDTQ-framework)
product development, 169
relationship, 179
SPADE and SPELL, 171
SPMLs, 171
standard guideline, 178
support processes, 171
tools, 179
UML-based approaches, 171

Software product line engineering (SPLE)
advantages, 218
approaches, 208
ATL model transformation process, 209
automatic derivation, product architectures, 208–209
characteristics, 208
definition, 218
description, 207–208
multimodel viewpoints
functional, 210
metamodel, 211
quality, 210–211
relationships, 211–212
variability, 210
quality-driven product architecture, 212–213
QVT transformation, 209
safety-critical embedded systems (*see* Automotive industry)
trade-off and optimization techniques, 219
SPLE. *See* Software product line engineering (SPLE)
Structural equation modeling (SEM), 35
Student motivation
changing activity, 542–543
changing process and tables redesign, 537
information, 535
status evolution, 535, 536
Sustainability
enterprise architecture
development, 368–369
strategy, 370
eProcurement, 559
ICT, 369
IT management
COBIT, 279–281
frameworks, 277–278
ITIL, 281–282
lack of, 276
methodology, 277–278
PRINCE2 processes, 282, 283
quality and development, Six Sigma, 278–279
requirement, 282
Sustainable information systems development (ISD)
complexity, cultures
agile development, 5
characteristics, organisational core cultures, 5–6
collaboration, 6
CompCul, 7–9
control, 6
CoopCul, 7
cultivation, 6
definition, 5
framework, 5
identification, 9–10
innovative and open organisational culture, 5
organisational culture theory, 1
projects and management (*see* Projects and management, ISD)
public sector organisation, German, 21

T

TBL. *See* Triple bottom line (TBL)
TEAF. *See* Treasury Enterprise Architecture Framework (TEAF)
Technology capabilities, VWs
communication, 546–547
definition, 546
interaction, 547
lifelong disability, 547–548
rendering, 547
The Open Group Architecture Framework (TOGAF), 372–373
Theory of planned behaviour (TPB), 106, 109, 110
TOGAF. *See* The Open Group Architecture Framework (TOGAF)
Toulmin’s model depiction, 580
TPB. *See* Theory of planned behaviour (TPB)
Transformation guidelines, CA, 604–605
Transition framework, software requirements and architecture
application, 148
approaches, 142–143
atomic requirements, module 1, 143–144
data collection
external couplings, 150
FW-based values, 149
metric values, 149
modularity factor, individual subsystems, 149
student projects, 148–149
description, 141–142
elements, module 4, 145–146
flexibility, 143
formalization and expert-based approaches, 143
goal and proposition, 147
humans, 151
impacts, module 2, 144
information systems design, 147

integration, 143
interaction modules, 146
knowledge reuse, 151–152
limitations, 151
mapping, 143
MSS design, 147–148
process and process modules, 143, 144
RE, 141, 142
surveys, 146
systematic, 142
textual description, module 3, 145
validity, 150–151
Treasury enterprise architecture framework (TEAF), 374
Triple bottom line (TBL), 274, 276, 279
Triple-helix model, government II, 564, 565
Triple modular redundancy pattern (TMR), 214
Trust
artifact adoption, 26–27
and behavioral intentions, 34, 37
construction, 26, 31–32
definition, 29–30
description, 25–26
digital identity, 25
empirical evidence, 26
IdMS (*see* Identity management systems (IdMS))
integration, 27
internet, 31, 36–37
literature, 26, 27
online survey, 34–35
perceptions, 35
perspective, 26
quantitative examination, 35
research model, 31
SEM technique, 35
services and technologies, 27
social responses, computers, 30–31
technological artifacts and IdMS, 28–29

U
UCD. *See* User-centered design (UCD)
UML. *See* Unified Modeling Language (UML)
Unified Modeling Language (UML), 255, 258, 287–289, 598, 608
Unifying meta-model
alignment, unifying EM meta-model and CA, 600, 602
CA communicative interaction level (L3), 603–604
CA process level (L2), 600, 602–603
EM (*see* Enterprise modeling (EM))
illustrative lab demonstration, 606–608
information-related actions, 604
MDD-oriented models, 600
transformation guidelines, 604–605
Usability work. *See* Agile systems development
User-centered design (UCD)
and agile methods, 573
work execution, 575–576
work organization, 576–577
User experience design. *See* Agile systems development

V
Value-chain. *See* Enterprise resource planning (ERP)
Vendors
ERP, 269, 270
and resellers, 265, 266
software, 254
Victorian biotech industry, 329
Virtual worlds (VWs)
adult learning centers, 549
affordances (*see* Affordances, VWs)
challenges and affordances, 546, 549
communication channels, 551–553
earlier research educators, 552
metaverses, 546
novice participants, 553
participants, experienced users, 549
participants list, 548–549
personalization, avatars, 553
and PWLD (*see* People with lifelong disability (PWLD))
research, SL, 548
socialization and inclusion, 545
technology capabilities (*see* Technology capabilities, VWs)
United Nations (UN) Convention, 546
VWs. *See* Virtual worlds (VWs)

W
Weighted sum model (WSM), 529
WM. *See* Working memory (WM)
Working memory (WM), 130–132

X
XML query adaptation
adding, 410–411
DaemonX, 404

data inconsistency, 403
evolution and adaptability, 403
evolution architecture, 404–405
natural extension, 418
non-existing element, 404
operations, 408, 409
queries, 417
reconnection, 415–417
removing, 411–413
renaming, 413–414
reordering, 414
self-explanatory names, 409
and XPath (*see* XPath)

XPath
axes and nodes, 407
definition, 405
expression node, 407, 408
mapping, 407, 408
positive core, 406
PSM schema, 405, 406
XSEM, 405

Z

Zachman framework (ZF), 373
ZF. *See* Zachman Framework (ZF)