Progress in IS

Rudy Hirschheim
Armin Heinzl
Jens Dibbern *Editors*

Information Systems Outsourcing

The Era of Digital Transformation

Fifth Edition

Springer

Progress in IS

"PROGRESS in IS" encompasses the various areas of Information Systems in theory and practice, presenting cutting-edge advances in the field. It is aimed especially at researchers, doctoral students, and advanced practitioners. The series features both research monographs that make substantial contributions to our state of knowledge and handbooks and other edited volumes, in which a team of experts is organized by one or more leading authorities to write individual chapters on various aspects of the topic. "PROGRESS in IS" is edited by a global team of leading IS experts. The editorial board expressly welcomes new members to this group. Individual volumes in this series are supported by a minimum of two members of the editorial board, and a code of conduct mandatory for all members of the board ensures the quality and cutting-edge nature of the titles published under this series.

More information about this series at http://www.springer.com/series/10440

Rudy Hirschheim · Armin Heinzl ·
Jens Dibbern
Editors

Information Systems Outsourcing

The Era of Digital Transformation

Fifth Edition

Springer

Editors
Rudy Hirschheim
Ourso College of Business
Louisiana State University
Baton Rouge, LA, USA

Armin Heinzl
School of Business
University Mannheim
Mannheim, Germany

Jens Dibbern
Institute for Information Systems
University of Bern
Bern, Switzerland

ISSN 2196-8705 ISSN 2196-8713 (electronic)
Progress in IS
ISBN 978-3-030-45818-8 ISBN 978-3-030-45819-5 (eBook)
https://doi.org/10.1007/978-3-030-45819-5

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

Preface of the Fifth Edition

More than 30 years after the Kodak Corporation handed over its information technology functions (IT) to IBM, DEC and Business Land, Information Systems Outsourcing has continued to thrive. Although some politicians have exerted pressure on firms to refrain from moving jobs outside of their home nations, recent surveys report outsourcing and offshoring as continuing to grow significantly.

In addition, outsourcing is on the verge of experiencing a significant impact from IT-induced automation. Advances in artificial intelligence and machine learning permit software to take over routines that have been previously conducted by humans. If these software robots (or bots) are provisioned via the cloud, they represent outside resources. Thus, IT and information systems (IS) facilitate the combination of two previously distinct concepts: outsourcing and automation. Bots can be easily provisioned from external service providers via the cloud and seamlessly integrated into internal business processes. This combination of software automation and outsourcing is a key phenomenon in the era of digital transformation. IT becomes ubiquitous, providing the capability to process vast amounts of digitized data through the wide availability of sensory devices, which are connected into the Internet of Things via the cloud. Thus, the era of digital transformation already signals a deep impact on the domain of IS outsourcing.

For this reason, we decided to name the fifth edition of our book *Information Systems Outsourcing: The Era of Digital Transformation*. As a basis for our considerations and discussion, we invited a multitude of renowned international scholars and key practitioners to the 5th International Conference on Outsourcing of Information Services (ICOIS) to Mannheim, Germany, again. The idea is to present the state of the art in IS outsourcing research as well as to discuss its implications for theory, practice, and society. The papers presented at Mannheim provide the foundation of the fifth edition of our book. It succeeds the fourth edition which looked at sustainable business value of IS outsourcing.

At the beautiful new Study and Conference Center of the Mannheim Business School, it became clear that most of the contributions to the conference focused on cloud computing, platforms, and robot process automation. Hence, the era of digital transformation has already offered a clearly visible footprint. After carefully discussing and editing the submitted papers, we decided to structure the content of this book into eight sections: (1) Emergent Sourcing Challenges, (2) Mastering Innovation Through Outsourcing, (3) Leveraging the Value of Offshoring, (4) Adopting and Innovating Cloud Services, (5) Balancing Risks and Opportunities in Cloud-Based Outsourcing, (6) Benefitting from Service Workforce Platforms, (7) Replacing Humans by Bots, and (8) Final Synthesis. Our book serves as a basis for further interactions and discussions in the rich and dynamic field of IS outsourcing. It should be of interest to academics and graduate students in the field of Information Systems, Innovation Management and Digital Transformation as well as to corporate executives and professionals who seek a better understanding of the underlying concepts and idiosyncrasies of IS outsourcing in the era of digital transformation.

We would like to offer our deepest gratitude to our respected colleagues from around the world who provided the foundations of this fifth edition. Such a book would not be publishable without the enormous efforts of all researchers involved. Thus, we would like to thank all contributing authors for their highly appreciated thoughts and professional cooperation. They have laid the foundation for this book! While we are very grateful to the authors for their contributions to this book, we take responsibility for the content and any errors.

In addition, we would like to convey our deepest gratitude to Nikolaus Kratzat and Matthias Hampel (Strategy&), Dr. Dennis Lips and Toan Nguyen (Anyon/e-shelter), as well as Dirk Schneider (Salesforce) for sharing their industry perspectives and complementing our academic views at the conference. Without their backing, ICOIS 2019 would not have been such a success.

Finally, we owe many thanks to those team members who supported us with respect to the vast organizational activities of the conference and this monograph. In particular, we would very much like to thank Anne Wesch from Mannheim for her relentless support in organizing ICOIS 2019 in Mannheim and Nicolas Mayr von Baldegg from Bern for designing the ICOIS website and providing help to Anne when the circumstance required to do so. Many thanks also to Nicolas, Anna Filippova, and Louis Felder for helping in the coordinative processes in creating this book.

We hope that you as our readers find the fifth edition as vivid and insightful as our last editions.

Baton Rouge, Louisiana — Rudy Hirschheim
Mannheim, Germany — Armin Heinzl
Bern, Switzerland — Jens Dibbern

Contents

Introduction: Riding the Waves of Outsourcing Change in the Era of Digital Transformation

Jens Dibbern and Rudy Hirschheim

Abstract The trend towards digital transformation has brought about a number of emerging challenges for information systems outsourcing. Organizations have to understand how to digitally innovate through IS outsourcing, how to govern outsourced digitalization projects, how to cope with complex multi vendor and micro-services arrangements, how to manage data sourcing and data partnerships, including issues of cybersecurity, and how to cope with the increasing demands of internationalization and new sourcing models, such as crowdsourcing, cloud sourcing and robotic process automation. This introductory chapter summarizes these challenges as three entangled or intermingled waves of change. It shows how recent research addresses these waves of change as a basis for organizations to learn how to successfully ride the waves.

1 Introduction

The notion of outsourcing—making arrangements with an external entity for the provision of goods or services to supplement or replace internal efforts—has been around for centuries. Kakabadse and Kakabadse (2002) track one of the earliest occurrences of outsourcing to the ancient Roman Empire, where tax collection was outsourced. In the early years of American history, the production of wagon covers was outsourced to Scotland, where they used raw material imported from India in the production process (Kelly 2002). Outsourcing remained popular in the manufacturing sector, with part of the assembling in many industries being sub-contracted to other organizations and locations where the work could be done more efficiently and cheaply. Commenting on this unstoppable trend, Pastin and Harrison (1974)

J. Dibbern (✉)
Institute of Information Systems (IWI), Information Engineering, University of Bern, 3012 Bern, Switzerland
e-mail: jens.dibbern@iwi.unibe.ch

R. Hirschheim
E. J. Ourso College of Business, Louisiana State University, Baton Rouge, LA 70803, USA
e-mail: rudy@lsu.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_1

wrote that such outsourcing of manufacturing functions was creating a new form of organization which they termed the “hollow corporation” (i.e. an organization that designs and distributes, but does not produce anything). They note that such an organizational form would require considerable changes in the way organizations were managed. While they limited their research to the role of management in the hollow corporation, they comment on the substantial (and unpleasant) social and economic changes that the outsourcing of manufacturing was causing.

It was not long before the idea of outsourcing was applied to the procurement of information technology (IT) services also. While the origins of IT outsourcing can be traced back to EDS’ deal with Blue Cross in the early sixties, it was the landmark Kodak deal in 1989 that won acceptance for IT outsourcing as a strategic tool. Many large and small outsourcing deals were inked in the years that followed.

From its beginnings as a cost-cutting tool, IT outsourcing has evolved into an integral component of a firm’s overall information systems strategy (Linder 2004). Still, reducing costs is an idea that never loses its appeal, and the opportunity to meet the IT demands of the organization with a less-expensive but well-trained labor pool has led organizations to look past the national borders, at locations both far and near, for such resources. There is little doubt about the continued acceptance and popularity of IT outsourcing as well as the trend towards outsourcing to different global locations. According to a recent Gartner study the global IT outsourcing market continues to grow having reached $283.5 billion in 2016 (Gartner 2017). IT outsourcing has also evolved from sole-sourcing and total sourcing arrangements of yester-years where one vendor provides all IT services to its client to complex arrangements involving multiple vendors and multiple clients (Gallivan and Oh 1999; Oshri et al. 2019). According to Mears and Bednarz (2005) companies are also outsourcing on a much more selective basis than ever before (See also Overby 2018). The tools and resources available today make it easier for IT executives to manage their IT portfolio and achieve the economies they need without outsourcing everything. (Of course, a key challenge is determining what pieces of the IT portfolio to outsource and what to keep internal.) Outsourcing also now embraces significant partnerships and alliances, referred to as co-sourcing arrangements, where client and vendor share risk and reward. These co-sourcing arrangements build on the competencies of the client and vendor to meet the client’s IT needs. Moreover, outsourcing has grown beyond the domain of IT embodying decisions such as where and how to source IT to a much wider set of business functions.

This inexorable trend towards IT outsourcing as being interwoven with business strategy has been fueled by the ongoing trend towards digital transformation. A market report by International Data Corporation (IDC 2019) forecasts the worldwide spending on the technologies and services that enable digital transformation of business practices, products, and organizations to reach $2.3 trillion in 2023 with a five-year compound annual growth rate of 17.1%. As organizations become increasingly aware of IT as an enabler of change, the role of IT outsourcing has begun to change as well. The notion of digital transformation is closely linked with new IT-enabled ways of providing IT services in the form of cloud computing and new IT-enabled business models, such as the emergent sharing economy (Apte and Davis 2019; Venters and

Whitley 2012; Weinhardt et al. 2009). Moreover, the potential of IT as a driver of change is not limited to supporting humans and business functions in new ways, but also increasingly includes the potential of replacing humans by automating business processes, i.e. through robotic process automation and the trend of replacing humans by bots (Lacity and Willcocks 2016; van der Aalst et al. 2018), also referred to as the 'second machine age' (Brynjolfsson and McAfee 2016). Cloud computing has considerably transformed the IT services industry in recent years. The provision of IT-as-a-service via the internet has enabled simpler access to IT resources and services and an increasing disaggregation of IT services into smaller components that are provided by an ever-larger pool of new players in the market. In parallel, large platform providers have emerged that seek to orchestrate the myriad of players in their platform ecosystems (Gawer and Cusumano 2002; Ghazawneh and Henfridsson 2013; Huber et al. 2017; Tiwana 2014). This trend towards platforms and ecosystems of IT services has important implications for IT outsourcing. It brings with it new challenges, such as the challenge of orchestrating IT services from a network of providers, to manage the scattered data resources and assets across service providers and to cope with heightened cybersecurity risks due to distributed services and the reliance on cloud infrastructures. Moreover, with the increasing digitalization of business processes, the availability of increasing amounts of data, and advancements in the application of automation techniques and technologies (including advanced machine learning algorithms),[1] the notion of robotics that revolutionized industrial engineering has entered the era of business process engineering in the form of business process automation and various types of bots that take over work previously performed by humans, e.g. chatbots replacing call center agents (Brynjolfsson and Mitchell 2017; Rutschi and Dibbern 2020). The reliance on bots can be viewed as a special type of outsourcing, i.e. outsourcing to machines, which provides new challenges, such as which processes to automate, by whom to automate (i.e. internally or externally), and how to manage human-machine interaction (with internally versus externally provided machines).[2]

Organizations have to walk a tightrope between the many opportunities that the evolving and emerging trends in outsourcing, offshoring, cloud computing, and robotics could provide and the risks and challenges they bring with them. Moreover, multiple partly opposing objectives have to be reconciled which has been framed under the notion of ambidexterity (Gregory et al. 2015). The objectives of increasing efficiencies and achieving cost savings through outsourcing have to be reconciled with the potential of leveraging innovation and transforming the business through outsourcing. Moreover, the adverse reactions from a society increasingly disenchanted by the job displacement and loss that outsourcing causes, have to be considered,

[1] "Machine learning aims to provide increasing levels of automation in the knowledge engineering process, replacing much time-consuming human activity with automatic techniques that improve accuracy or efficiency by discovering and exploiting regularities in training data." (Langley and Simon 1995, p. 55).

[2] According to Wikipedia drawing on the manufacturing context (based on Groover 2014), "Automation is the technology by which a process or procedure is performed with minimal human assistance.".

which also brings about new opportunities for clients and venders to take on societal and economic responsibility in their IT outsourcing endeavors and relationships (Carmel et al. 2014).

2 Three Waves of IT Outsourcing

Whilst outsourcing has evolved considerably since the late 1980s where large IT outsourcing vendors signed multibillion-dollar deals with clients involving the wholesale transferal of corporate IT to these vendors, we see several emerging trends or waves that are associated with the rise of digital transformation.

According to Westerman et al. (2014) digital transformation marks a radical rethinking of how an organization uses technology, people and processes to fundamentally change business performance. While the notion has been around for some time, the role of outsourcing within this transformation has only relatively recently been considered. It appears that the emphasis has shifted from outsourcing legacy and/or traditional services to outsourcing for digital transformation. In fact, according to a recent market study by IDC, worldwide spending on digital transformation will reach $2.3 trillion in 2023, more than half of all ICT spending (IDC 2019). Organizations are looking to vendors, consultants, and researchers who can assist them in this transformation. Academic researchers are now exploring emerging sourcing topics such as crowdsourcing (Blohm et al. 2013; Geiger and Schader 2014), platform ecosystems (Constantinides et al. 2018; Foerderer et al. 2018; Ghazawneh and Henfridsson 2013; Huber et al. 2017; Schmeiss et al. 2019; Tiwana 2002), cloud computing (Schneider and Sunyaev 2016; Yinghui et al. 2018), service innovation (Barrett et al. 2015; Lusch and Nambisan 2015), service automation (robotic process automation—RPA) (Lacity and Willcocks 2016; Rutschi and Dibbern 2020; Willcocks and Lacity 2016), artificial intelligence/machine learning (Davenport and Ronanki 2018), process mining/analytics (Fogarty and Bell 2014), internet-of-things (Dijkman et al. 2015), and blockchain (cf. Lacity and Willcocks 2018).

The rapid growth of digital transformation has led to a changing IT outsourcing landscape that over the past decade has involved into what might be considered three entangled or intermingled 'waves of change'. They are:

A. The evolving traditional outsourcing of IT services, which refers to the outsourcing of IT functions, i.e. IT tasks, such as software development or data center operations, that are performed by external IT work forces. In such labor-intensive traditional outsourcing of IT services, enduring trends include offshoring and multisourcing, which have been around for some time. But also new sourcing arrangements that are characterized by novel value propositions, such as striving for innovation through outsourcing rather than simply cost savings or getting access to scarce resources. The notion of impact sourcing also relates to a new view on value. It means that clients and vendors consider how their outsourcing arrangements contribute to creating social and society-wide (rather than purely firm) economic value.

B. The emergence of cloud computing and platform ecosystems, which refers to a new way of service development and delivery by the IT industry, i.e. where IT services are developed in large platform ecosystems and provided via platforms with new pricing models (i.e. renting readily available services) and providing services via the internet (i.e. cloud) as Software as a service (SaaS), Infrastructure as a service (IaaS), or Platform as a service (PaaS) (Weinhardt et al. 2009). The move towards platforms also includes the provision of labor as a service and has led to entirely new business models that disrupt traditional industries. This includes crowdsourcing, i.e. engaging the crowd in a new service delivery model, and embracing the use of digital platforms to expose untapped supply and demand of services that are based on the sharing of individually owned resources and assets as exemplified by Airbnb and Uber.

C. The development of robotic process automation and 'outsourcing' to software bots, where entire tasks or business functions are taken over by IT, i.e. a machine, such as a chatbot. Thus, the goal is not to support humans with ever cheaper or better IT services that may stem from external providers, but rather to replace humans by IT.

Although these represent three distinct waves of change, they are in fact intermingled. As visualized in Fig. 1, there are logical interfaces between the waves (i.e. circles). For example, a software bot may be developed by an external service provider using a traditional outsourcing arrangement with an external vendor and the bot may take over work from former in-house personnel of the client and hence the work is outsourced to the bot (Interface between A and C). The bot may then also be provided as a service via cloud computing (Interface between B and C).

The case of Volvo's digital transformation journey (Svahn et al. 2017) is a good example of how digitalization is enabled by all three waves of outsourcing. In order

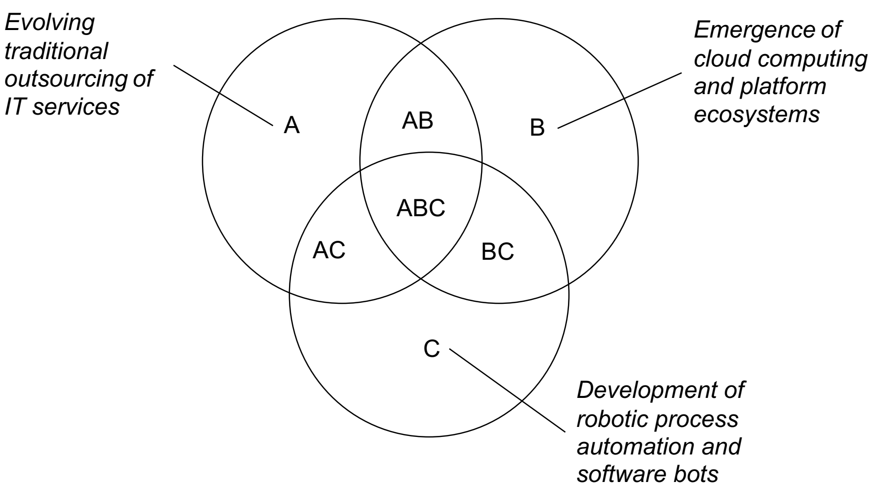

Fig. 1 Crossing waves of change in IT outsourcing

to enter the digital world, Volvo relied on new sourcing models for innovation, i.e. it engaged the crowd to gain new ideas on how the automobile will be transformed into a digital vehicle and what kind of complementary digital services they have to offer to their customers. In implementing these novel digital services, they partly relied on traditional IT outsourcing models by engaging external vendors, but the new digital services were also heavily based on cloud services as the dominant delivery model. In fact, Volvo initially relied on an external cloud infrastructure as a service, but eventually developed its own infrastructure to digitally connect to surrounding digital services, such as bringing Spotify into the car and connecting the car to the environment, i.e. the world of internet of things (Svahn et al. 2017). The latter, i.e. connecting the car to its environment, brought with it an increasing potential of making use of digital data for optimizing processes within the car, leading to add-on services including the additional potential for robotic process automation, e.g. in predictive maintenance.

Thus in the course of digital transformation, organizations increasingly gain the ability of riding all three waves of IT outsourcing.

3 Motivation for the Fifth Edition

This fifth edition of the edited book on *Information Systems Outsourcing* seeks to provide new insights on how organizations can cope with the challenges and opportunities of information systems (IS) outsourcing that arise in *The Era of Digital Transformation*. It offers a roadmap of the current IT outsourcing academic literature, highlighting new perspectives while also considering what has been learned so far and how the work fits together under a common umbrella.

When we produced the first edition of the book *Information Systems Outsourcing in the New Economy: Enduring Themes, Emergent Patterns and Future Directions* in 2002, the motivation rested on the need to take stock of a field, which had been around for about 10 years. Since then, we published a paper which offered a good overview of the field (Dibbern et al. 2004). However, because it was a paper, it could not do justice to the depth and breadth of the outsourcing landscape and with the dynamic developments in the field, such as IS offshoring. To that end, a second edition was developed in 2006. In that follow-up edition, we reproduced a number of what we considered more ‘classic’ papers in the field and supplemented them with a large number of new contributions, in particular on the topic IT offshoring. This new direction was reflected by the subtitle: *Enduring Themes, New Perspectives and Global Challenges.*

Following on from the second, came the third edition in 2009, which included a completely new collection of papers on the topic of information systems outsourcing. Similar to the first edition, the contributions of the third edition were based on an international conference that we held for the third time involving key researchers from around the world with a proven track record in the field of Information Systems Outsourcing. The third edition book was based on the research presented by the participants attending the 3rd International Conference on Outsourcing of Information Services (www.ICOIS.de) which was held in Heidelberg, Germany, May 29–30, 2007. This edition was reflected by the subtitle: *Enduring Themes, Global Challenges, and Process Opportunities.*

Subsequent to the third edition, we held our 4th International Conference on Outsourcing of Information Services (www.ICOIS.de) which was held in Mannheim, Germany, June 9–11, 2013. As in previous ICOIS events, this brought together key researchers in the field discussing their latest research and thinking about outsourcing. The papers presented at the conference formed the basis of the fourth edition of the book. At that event, the majority of the contributions to the conference shifted their focus towards new forms and mechanisms of outsourcing that aim at offering a more long-term and value-oriented perspective on IT outsourcing. Hence, the subtitle of the fourth edition was: *Towards Sustaining Business Value*.

As the field of Information Systems Outsourcing continued to mature, new technologies emerged and along with them, new forms of outsourcing. The advent of crowdsourcing where work is outsourced to the 'human cloud' (Kaganer et al. 2013) has provided new opportunities for how and where work is done and by whom. This along with the arrival of new technologies and tools such as social media, mobile applications, big data analytics, cloud computing (SMAC) and more recently the internet-of-things and robotic process automation, made it clear that another conference was needed. Thus, we held the 5th International Conference on Outsourcing of Information Services in Mannheim, Germany, June 16–18, 2019 to explore these new developments. As in the past, this event brought together key researchers in the field to offer their thoughts, reflections, and research results on these new topics. The papers presented at the conference form the basis of this 5th edition whose subtitle is termed *The Era of Digital Transformation*.

4 Book Structure and Outline

In seeking to be a resource for researchers and practitioners alike, the fifth edition of *Information Systems Outsourcing* subtitled *The Era of Digital Transformation*, is organized into eighth sections (i.e. Parts). The chapters in each section fall into one or more of the three outsourcing waves identified in the era of digital transformation (see Fig. 1). They either address issues concerning one specific wave of outsourcing or deal with issues at the intersection of waves.

4.1 Part I: Emergent Sourcing Challenges

Following this introductory Chapter (i.e. Chapter "Introduction: Riding the Waves of Outsourcing Change in the Era of Digital Transformation"), the first section examines and provides foundations for understanding *Emergent Sourcing Challenges*.

These emergent sourcing challenges arise at the interface between the evolving traditional wave of IT outsourcing and the two emergent waves of outsourcing, i.e. the wave of cloud computing and platform ecosystems and the wave of robotic process automation or 'outsourcing' to software bots (i.e. ABC in Fig. 1). In fact, the two emergent waves have implications for and can be viewed from the rich foundations of traditional outsourcing research that continues to evolve. Specifically, the important themes of making the sourcing decision, designing contractual structures, and managing the relationship need to be re-evaluated and revisited in light of the increasing digitalization of IT services both in terms of cloud computing and robotics.

Bozan, Lyytinen, and Rose (Chapter "Software Architecture and Outsourcing Governance: Raising Thoroughbreds Versus Cultivating Schools of Goldfish") identify a number of compelling questions that arise in light of the increasing availability of so called micro services based on which client organizations increasingly build their IS landscapes. These micro services are provided by ecosystems of cloud services (including SaaS, PaaS, and IaaS) providers, which are often organized along service stacks and platforms provided by dominant players in the industry that form their own platform ecosystems. Choosing the right portfolio of services and service stacks as well as orchestrating the services and service providers leads to new sourcing challenges that the authors present using the metaphor of a school of goldfish that has to be populated, fed, maintained and put into the right pond with the right environment.

Aubert and Rivard (Chapter "The Outsourcing of IT Governance") examine how IT governance needs to be rethought in light of a changing sourcing landscape characterized by an increasing reliance on outsourcing, offshoring, use of platform services, consumerization of IT and pressure to comply with regulations and standards, such as the Sarbanes–Oxley Act. They argue that these trends have led to an erosion of control over IT governance and suggest a reconceptualization of IT governance in terms of new decision-making structures, processes, and relational capabilities.

Jarvenpaa and Markus (Chapter "Data Sourcing and Data Partnerships: Opportunities for IS Sourcing Research") argue for a change in perspective towards putting a spotlight on data as the key object of sourcing. Access and preparation of data, but also the ability to make use of data has increasingly become a source of competitive advantage. Many new business models are based on the application of machine learning and artificial intelligence algorithms that are only as good as the data with which the algorithms are fed. While organizations produce large amounts of data by themselves, they also increasingly rely on external data, which brings along new challenges for the sourcing of data. The authors discuss the implications of different forms of data partnerships in light of different views of data, i.e. the commodity view, the process view, and the relational view. In doing so they draw a link to traditional perspectives on outsourcing regarding decision-making, contracting, and relationship management.

4.2 Part II: Mastering Innovation Through Outsourcing

In the second section, scholars provide insights into how to achieve innovation through outsourcing, which can be viewed as an evolving theme in traditional IS outsourcing (i.e. Wave A). The focus on innovation as an objective of IT outsourcing has been recognized early on in the outsourcing literature. For example, DiRomualdo and Gurbaxani (1998) have distinguished different strategic intents associated with outsourcing, such as IS improvement ("do IS better"), business impact ("use IT to/achieve better business results"), and commercial exploitation ("exploit IT assets externally). This notion of strategic innovation through outsourcing (Oshri et al. 2015), however has only recently gained momentum in light of today's developments towards digital transformation, which refers to the process of using digital technologies to create new—or enhanced—business processes, customer experiences, and business models (Capgemini 2011). Such digital transformation often occurs with the help of external vendors that develop new innovative IT solutions that can bring along either incremental or radical changes to the client organization. However, the focus on achieving innovation still provides a stark contrast to the more common objectives associated with outsourcing, such as focusing on core competencies and cost savings. In fact, contracting for IT services and achieving innovation have been viewed as a paradox that needs to be addressed sensitively (Aubert et al. 2015). Acknowledging the increasing importance of innovation as an outsourcing objective, the following two studies provide new insights into current innovation practices in the context of IT outsourcing. Specially, they provide insights into the general capabilities required for achieving innovation through outsourcing and the specific approach of making use of multiple vendors in a 'coopetitive' (cooperative and competitive) setting to achieve a desired digital innovation product.

Meiser and Beimborn (Chapter "Innovation in Outsourcing—An Empirical Analysis of Outsourcing Vendors' Innovation Approaches") take stock of current innovation practices in IT outsourcing. Based on data on innovation initiatives of a sample of 180 outsourcing providers they identify key innovation outcomes of these innovation initiatives and their enablers. Notably, the most often mentioned innovations by vendors referred to the development of new, innovative IT products for client organizations, specifically particular software solutions, such as speech and text analysis software or chatbots for employee self-service apps. Innovations are enabled by different factors that can be grouped along four dimensions, i.e. collaboration, processes, structures, and events. Two of them, collaboration with clients and the establishment of innovation labs (as a structural initiative) stood out as especially important for enabling innovation.

Hurni, Dibbern and Huber (Chapter "Emerging Innovation Ecosystems: The Critical Role of Distributed Innovation Agency) focus on the development of IT innovations as a product through outsourcing in a multi-sourcing context. Drawing on the concept of innovation ecosystems, they studied the emergent mechanisms that helped a large logistics provider to master the orchestration of multiple vendors in developing a multi-million dollar mobile solution for its thousands of logistics workers.

Specifically, their findings brought to light the critical role of distributed innovation agency, i.e. the critical role that particular vendors played in coping with emergent coordination challenges that had to be resolved in order to progress in the development of the innovative solution. When particular vendors took agency, this resulted in new procedures and structures that benefited the multi-sourcing arrangement as a whole. Moreover, such behavior mirrors both cooperative (i.e. helping others) and competitive behavior (i.e. improving its own position compared to others) as another constituent element of the emergence of an innovation ecosystem.

4.3 Part III: Leveraging the Value of Offshoring

The third section puts a spotlight on the practice of offshore outsourcing which reflects another continuously evolving theme in traditional outsourcing (i.e. Wave A). Offshoring refers to the outsourcing of IT tasks to an internal or external provider located in another country, also referred to as international or global outsourcing (Apte et al. 1997; Lacity and Willcocks 2001; Rottman and Lacity 2004; Sabherwal 1999; Sahay et al. 2003). While offshoring has been viewed as a major organizational innovation in the 90s and early 2000—mostly due to its significant potential for achieving cost savings—it is now widely recognized as a complementary approach to domestic outsourcing for the provision of IT services, not just in terms of lowering costs but also for achieving quality improvements (Gopal and Koka 2012). In order to achieving the expected benefits associated with offshore outsourcing, however, it is important to recognize both its unique opportunities and challenges (Winkler et al. 2008). In fact, rather than focusing on either the opportunities or the challenges/risks, it is important to find a balance between both (i.e. opportunities and challenges) in order to achieve sustainable value from outsourcing. For example, while offshoring promises lower "production" costs due to lower labor costs, it is also necessary to consider the extra costs associated with outsourcing, in terms of costs for monitoring, coordinating, transferring knowledge, and specifying requirements (Dibbern et al. 2008). Three chapters of this book are dedicated to a better understanding of the idiosyncratic aspects of offshore outsourcing, and how to cope with its inherent tensions in this balancing act.

Krancher and Dibbern (Chapter "Knowledge Transfer in Software Maintenance Outsourcing: The Key Roles of Software Knowledge and Guided Learning Tasks") focus on how to cope with the challenge of knowledge transfer in offshore outsourcing. When offshore vendors take over IT tasks from a client it is often not feasible to acquire previous in-house knowledge through people transfer (as often practiced in domestic outsourcing). Instead, vendor personnel have to gain the (often) client-specific knowledge necessary to perform the required tasks through a process of learning from those that previously performed the task. The authors unpack this knowledge transfer process through a longitudinal study by examining a number of knowledge transfers of a bank that outsourced software maintenance tasks to Indian vendors. The analysis of respective knowledge transfers from former subject matter experts to vendor software engineers revealed that knowledge transfer occurred through a series of knowledge transfer mechanisms and that the suitability of mechanisms depends on the type of task outsourced and the respective prior expertise of the vendor personnel.

Ning (Chapter "Cultural Intelligence of Offshore IT Suppliers: A Cultural Frame Perspective") examines how an offshore vendor with customers in different cultural regions, can cope with misfits between its own national culture and that of its international clients and how such a process of developing cultural intelligence can even strengthen the vendor's competiveness through transferring best practices of client regions. Based on a case study of a Chinese vendor that experienced significant growth both domestically and internationally, i.e. specifically in the US and Japan, he identifies the process through which the vendor personal gained cultural intelligence over the years by interacting with clients of different cultural backgrounds. Through these interactions, new cultural frames emerged and became embedded into the vendor's work routines. The cultural frames included the development of unique processes, structures, and artifacts that matched the cultural frames of the client regions (i.e. China, Japan, or US). Moreover, transferring cultural capabilities from one client region to another helped increasing the vendor's international competiveness. The findings are summarized in a framework that generalizes the process of achieving cultural intelligence through cultural sense-making.

Fareesa and Nicholson (Chapter "Competing Institutional Logics in Impact Sourcing") also take the perspective of an IT outsourcing vendor, but rather from the perspective of a Western vendor that seeks to extend its operations to offshore regions. In particular, they focus on the rather novel value proposition of impact sourcing, i.e. the practice of building up an offshore service facility not just for business reasons, but also with the objective of contributing to the welfare of marginalized people. Specifically, they seek to examine how vendors can cope with the tensions that arise between competing institutional logics associated with impact sourcing, i.e. the market versus welfare logic. Based on their analysis of a U.S.-based IT outsourcing vendor that has established two offshore centers in Pakistan, they identify the organizational responses taken to address the opposing logics. The responses can be framed as decoupling, combining, and creating compromises between market and welfare logics.

4.4 Part IV: Adopting and Designing Cloud Services

While the previous two sections (Parts II–III) are concerned with evolving themes around the traditional wave of IT outsourcing, the next two sections deal with one of the emergent waves of outsourcing, i.e. cloud computing (Wave B). While some of the unique sourcing challenges associated with cloud computing are already discussed in the first section (i.e. Part I), the two chapters of the next section (Part IV) start with outlining specific challenges concerned with the adoption and design of cloud services.

Gunupudi and Kishore (Chapter "The Differential Benefits of Cloud Computing for Small and Medium Versus Large Firms") provide an introduction into cloud computing and provide initial insights into differences in the adoption of cloud computing services between large firms and small-and-medium-sized enterprises (SME). Specifically, they distinguish key capabilities of cloud services, such as heterogeneity, scalability, consumption based pricing, and accessibility. They argue that organizations differ in their orientation towards exploration and exploitation to leverage these cloud capabilities for deriving strategic and operational benefits from adopting cloud services. Their findings from a survey among 41 firms show support for their supposition that firm size matters for the aspired capabilities and achieved benefits from adopting cloud services.

Hoffmann, Spohrer, and Heinzl (Chapter "Analyzing Usage Data in Enterprise Cloud Software: An Action Design Research Approach") take the perspective of cloud service providers. They seek to gain a better understanding of how cloud providers can make use of their unique immediate access to usage data (i.e. data on how cloud services are being used by the clients' end users) to support the process of requirements engineering for (re-)designing the providers' cloud services portfolios. Specifically, they contrast the new (usage-)data driven approach for requirements engineering with traditional, feedback-based approaches and suggest that the data-driven approach should be supported by a specific tool (i.e. IT artifact). They frame their study as following an action design research approach and outline the stages to take for its implementation.

4.5 Part V: Balancing Risks and Opportunities in Cloud-Based Outsourcing

The fifth section takes the perspective of organizations that already opted for choosing to adopt cloud services and hence face the challenge of exploiting the opportunities while mitigating the risks associated with this novel approach in the provision of external IT services. Addressing this challenge may be viewed as a balancing act that can take different forms depending on the risks and/or opportunities that adopting client firms are confronted with as well as the contextual conditions of the particular firms that often change over time. The following three chapters examine these risks and opportunities associated with cloud computing with special consideration of the outsourcing context, i.e. the fact that cloud computing can also be viewed as a client-vendor relationship, where a client contracts services from an external entity. They can hence also be viewed as being concerned with the crossing of waves, i.e. viewing issues discussed in the traditional wave of outsourcing (i.e. how to manage risks and opportunities of outsourcing) in the context of cloud computing (Waves AB).

Huber, Dibbern and Fischer (Chapter "How and Why Software Outsourcing Projects Drift—An Actor-Network-Theoretic Investigation of Control Processes") examine how organizations can cope with drift in outsourced software development projects, where the software is provided as SaaS by an external vendor. They define drift as a creeping process of targeting emergent goals, which may (or may not) occur at the expense of losing sight of initial goals. Based on a multiple case study of SaaS-based outsourcing projects in an international bank, they found different patterns of responses to emergent goals, representing either balanced drift (reconciling emergent with initial goals) or unbalanced drift (favoring emergent or initial goals). In order to explain the different routes of drift they contrast the traditional view of project control with an actor network perspective. This leads to a novel dynamic perspective on control that emphasizes the role of the non-human actors in shaping the direction of outsourced software projects. In the SaaS context, such non-human actors contribute to shaping the drift process in the form of changes in the outsourcing contract with the SaaS provider and changes in the SaaS, i.e. the software and the tasks. While human agency is an important trigger of control adaptations, the implementation of controls in form of contract changes and changes in the software and task can take on a life of its own with ripple effects for shaping the directions that SaaS projects take.

Benaroch (Chapter "Cybersecurity Risk in IT Outsourcing—Challenges and Emerging Realities") examines how organizations can cope with increasing cybersecurity risks due to the growing reliance on cloud computing. In fact, in today's IT infrastructures, clients often rely on layered cloud supply chains including multiple providers and sub-contractors that host sensitive client data. He argues that such enhanced cybersecurity risks need to be balanced by client-provider trust, which can stem from different sources, such as decision making, transparency, and reliance on market mechanisms that nurture service providers to behave in ways that enhance (i.e. do not destroy) their reputation. Focusing on the latter, they argue that negative effects on firm value due to cyber security failures are more likely when having deficiencies in IT general controls that ensure and signal having proper enterprise IT processes in place (such as measures in accordance with COBIT). These negative consequences, however, are weekend by board IT competencies. The results from 110 cyber failures in public US firms support these arguments showing the importance of investing into process controls and IT board competency for avoiding stock market penalties in case of failures.

Gozman, Machaiah, and Willcocks (Chapter "Cloud Sourcing and Mitigating Concentration Risk in Financial Services") take another view on cybersecurity risks due to the increasing adoption of cloud services by focusing on concentration risks and how to mitigate them. Concentration risks result from the increasing reliance on single dominant players in the cloud services market. Examples of such central hubs are Amazon and Google that provide cloud infrastructures and platforms that are interlinked with various other cloud services. If such central infrastructures brake down, an entire ecosystem of services is effected including its clients. Taking the financial services industry as a point of reference, they gather data from various sources to identify the drivers of concentration risks, the risk of contagion, and mitigating factors. Based on their analysis they discuss the role of regulators in defining standards for ensuring interoperability between service providers, developing a consistent terminology, and specifying the obligations during incidents, e.g. in terms of guidelines of communication and cooperation among the involved and affected parties.

4.6 Part VI: Benefitting from Workforce and Sharing Platforms

The sixth section deals with the emergence of online platforms for conveying workforces and shared assets as new forms of IT-enabled matching of supply and demand of services (Wave B). These platforms take on different forms with varying purposes. One form refers to crowdsourcing platforms, which enable clients to source tasks to a collection of people (i.e. the crowd), registered on the platform. One of the key benefits of engaging the crowd is to benefit from distributed knowledge (i.e. collective intelligence) (Malone et al. 2010). Crowdsourcing is often used with the purpose of developing new ideas or porotypes of innovations that may then be further developed by the client. Such initatives often take place in the form of organized competitions. However, in general, the process of developing ideas and innovations in a distributed fashion can be designed in various forms, e.g. more or less transparent, which may either foster or prevent the exchange of knowledge sharing among the crowd. Crowdsourcing has become relevant for IT outsourcing as far as the crowd can also be used for developing IT innovations or gaining ideas for digital innovations. As such, typical sourcing questions (coming from traditional outsourcing) arise, such as engaging an internal or external crowd, how to incentivize the participants, and how and to what extent to engage a client in supporting the participants in their task execution. Another form of service workforce platforms has emerged under the umbrella of the sharing economy. Such platforms have emerged as rather radical forms of digital innovations or transformations as they can disrupt entire industries. Prominent examples are Uber and Airbnb. These platforms are relevant for sourcing as far as they enable entirely new forms of engaging and governing individuals as service providers. Specifically, the engagement occurs in a triadic relationship between the consumers of the service, the providers of the service and the platform (that serves as an intermediary between consumers and providers). Since all transactions between consumer and platform as well as between provider and platform take place in digital form, the platform provider obtains a large amount of data that can be used for various purposes in the process of matching supply and demand, prizing services, and guarantying service quality, which are key elements of any sourcing arrangement. Two chapters deal with the different types of emergent online platforms for conveying and sharing services provided by individuals. The two chapters can be viewed as dealing with the intersection between the traditional wave of outsourcing and the emergent wave of platform-based sourcing (i.e. crossing Waves AB).

Nevo and Kotlarsky (Chapter "Scoping Review of Crowdsourcing Literature: Insights for IS Research") provide an overview of research on crowdsourcing in the IS context. The authors identified and analyzed 484 papers on crowdsourcing. Nine types of papers could be distinguished based on their research focus, among them papers focusing on crowdsourcing applications, on the design of the online platforms and of competition, or on aspects of participation. They conclude with discussing implications for research, specifically how to cope with the various crowdsourcing models and platforms. For this purpose, they discuss different taxonomies of crowdsourcing and their implications for research on sourcing.

Wiener, Cram and Benlian (Chapter "Technology-Mediated Control Legitimacy in the Gig Economy: Conceptualization and Nomological Network") examine the role of technology-mediated control in online platforms for matching supply and demand of a variety of services provided and consumed by individuals. Such individual-based services are often based on the sharing of assets with different degrees of complementary labor services (i.e. 'gig' workers), such as ride sharing (e.g. Uber) and home sharing (e.g. Airbnb), but also home services (e.g., Handy). The providers of such online platforms possess a considerable amount of control over the service provision process based on their access to data about transactions and services, which raises questions of legitimacy from the perspective of those that provide the service (i.e. the 'gig' workers). The authors explore the foundations of such technology-mediated control legitimacy in terms of autonomy, fairness and privacy. They also developed conceptual foundations on the antecedents, consequences and contextual boundary conditions of control legitimacy.

4.7 Part VII: Replacing Humans by Bots

The seventh section deals with the emergent phenomenon of replacing humans by bots (i.e. Wave C), which can take the form of outsourcing to bots (i.e. interface between Waves A and C). A bot represents a "…software application that runs automated tasks over the Internet. Typically, bots perform tasks that are both simple and structurally repetitive, at a much higher rate than would be possible for a human alone." (Wikipedia on "Internet bot"). While such simple and structurally repetitive tasks have been key candidates for business process outsourcing—increasingly in low wage countries, i.e. outsourcing or offshoring to humans supported by IT-systems (Gewald and Dibbern 2009)—many companies have begun to experiment with and implement bots that replace human workforces (Willcocks and Lacity 2016). Such robotic process automation and outsourcing to bots promises additional cost savings but can also lead to enhanced quality, e.g. by becoming immune to human error in task execution. While it is unquestionable that the reliance on bots provides vast potential that will likely grow substantially with the application of artificial intelligence and machine learning algorithms (van der Aalst et al. 2018), they also pose a number of unique challenges that are yet to be explored. Two chapters contribute to a better understanding of the opportunities and challenges of replacing humans by bots by putting a spotlight on their implementation and use.

Asatiani, Penttinen, Ruissalo, and Salovaara (Chapter "Knowledge Workers' Reactions to a Planned Introduction of Robotic Process Automation—Empirical Evidence from an Accounting Firm") examine the implications of the introduction of robotic process automation for the actual workers that are potentially replaced by bots and/or that have to interact with them. Based on a case of a Finish accounting company that considered introducing RPA into its financial services by drawing on machine learning algorithms, they conducted a series of interviews with affected workers to assess their reactions. The results were rather surprising in that worker reactions were mostly positive, such as seeing the potential for upgrading their jobs, better coping with high workloads, and reducing human errors. The challenges mentioned were related to the integration of robots into the workflow, such as losing control over work through the fragmentation of tasks, i.e. losing sight of the process as a whole. The authors discuss the implications of their findings in terms of cooperation between humans and machines and work atmosphere.

Rutschi and Dibbern (Chapter "Towards an Understanding of Scaling the Software Robot Implementation") focus on the process of implementing bots. Specially, they examine the scaling of bots, which essentially refers to the reuse of already implemented bots or parts thereof in new contexts. Such scaling is motivated by the objective of reducing the costs of implementation in the course of replacing more and more human-executed tasks by machines. The authors identify different scaling modes as well as their triggers, impediments, and enablers (i.e. mitigating factors). Each iteration of reusing components of already existing bots represents a scaling stage, so that over time, the basis for scaling becomes larger thus reinforcing the process of growth in implementing and using bots. The authors illustrate this scaling process with a real-world case of the scaling of a chatbot in a bank that moved from a simple one-language chatbot to one being bilingual, integrated into e-banking, and becoming also a voicebot. They conclude with an outlook for future research on robot scaling.

4.8 Part VIII: Final Synthesis

Our edited book ends with a synthesis reflecting on the presentations and papers submitted to the 5th International Conference on Information Services (ICOIS 2019), which formed the basis of this edited book. In fact, 17 of the 22 presented papers at ICOIS have found their way into this book as updated, or refined versions.

Sabherwal (Chapter "Synthesis: Outsourcing of Information Services: Where Are We?") contributes the final chapter and concluding chapter of this book, by reflecting on the 22 research studies on IS outsourcing presented at ICOIS 2019, in terms of a final synthesis. In doing so, he discusses the state of the art of IS outsourcing research by asking six questions. The first four questions help to grasp the phenomenon of

IS outsourcing and the directions it has taken in recent years, asking: (1) what to outsource, (2) to whom to outsource, (3) how to outsource, and (4) “so what”, i.e. what the implications of outsourcing are. The fifth and sixth questions are oriented towards research issues, asking: (5) what literature bases do current IS outsourcing research draw on, and (6) what empirical methods are being used. The chapter ends with broader reflections on how IS outsourcing has transformed in recent years in terms of the actors involved and the governance of outsourcing arrangements. These also have important implications for practice.

5 Conclusion

Overall, in reading the various chapters in this book, we reflected upon what we know and what we don’t know about the field. Although this fifth edition of the book did much to document what has been learned about IT outsourcing since our last edition in 2013, numerous interesting questions remain. In this book we have framed the contributions under the topic of IS outsourcing in the era of digital transformation. This is not merely a shift towards a new perspective on strategic outsourcing. Instead, digital transformation is a multifaceted issue that manifests itself in various aspects that have to do with a new value proposition of IT outsourcing, a new way of IT-enabled service delivery (i.e. cloud computing), fundamental changes in the IT service industry (i.e. towards platforms and ecosystems), and a new role of IT as an agent rather than passive object of sourcing (i.e. IT increasingly takes on the role of the agent in the typical principal-agent relationship taking over tasks in the form of bots or software robots that perform tasks autonomously).

As this book clearly shows, outsourcing has evolved dramatically over the last 30+ years—in some planned and unplanned directions. In general, the future of IT outsourcing appears wide open with many unanswered questions. For example, will more and more traditional IT outsourcing services move into (standard) cloud services? Will the increasing disaggregation of services lead to a new form of complexity that requires entirely new approaches of decision-making, orchestration, governance, and control? Will robotic process automation continue to evolve to take over more and more intelligent rather than repetitive and standard tasks? In addition, even if robots take over more and more (intelligent) tasks, who will be building them, providing them as a service, implementing them, and maintaining them? How will organizations and humans keep control over outsourced bots that become more and more autonomous in carrying out tasks and in adapting to the environment? How can privacy, property rights, and security of data be maintained in a world of cloud computing and bots? And, finally, what task will remain for humans?

We have tried to articulate some of these important questions but there are many more. Hopefully, this book will help motivate individuals to either begin research in the field or continue engaging in outsourcing research. Much has been done, but there is still much more to be done. We hope the reader enjoys the papers in this volume. Happy reading!

References

Apte, U. M., & Davis, M. M. (2019). Sharing economy services: Business model generation. *California Management Review, 61*(2), 104–131.
Apte, U. M., Sobol, M. G., Hanaoka, S., Shimada, T., Saarinen, T., Salmela, T., & Vepsalainen, A. P. J. (1997). IS outsourcing practices in the USA, Japan and Finland: A comparative study. *Journal of Information Technology*, *12*(2), 289–304.
Aubert, B. A., Kishore, R., & Iriyama, A. (2015). Exploring and managing the "innovation through outsourcing" paradox. *Journal of Strategic Information Systems, 24,* 255–269.
Barrett, M., Davidson, E., Prabhu, J., & Vargo, S. (2015). Service innovation in the digital age: Key contributions and future directions. *MIS Quarterly, 39*(1), 135–154.
Blohm, I., Leimeister, J. M., & Krcmar, H. (2013). Crowdsourcing: How to benefit from (too) many great ideas. *MIS Quarterly Executive, 12*(4), 199–211.
Brynjolfsson, E., & McAfee, A. (2016). *The second machine age: Work, progress, and prosperity in a time of brilliant technologies*. Norton & Company.
Brynjolfsson, E., & Mitchell, T. (2017). What can machine learning do? Workforce implications. *Science*, *358*(6370), pp. 1530–1534.
Capgemini. (2011). *Digital transformation: A roadmap for billion-dollar organizations*. Capgemini Consulting Study Report.
Carmel, E., Lacity, M. C., & Doty, A. (2014). The Impact of Impact Sourcing: Framing a Research Agenda. In R. Hirschheim, A. Heinzl, & J. Dibbern (Eds.), *Information systems outsourcing: Towards sustainable business value* (pp. 397–429). Berlin, Heidelberg: Springer.
Constantinides, P., Henfridsson, O., & Parker, G. G. (2018). Introduction—Platforms and infrastructures in the digitalage. *Information Systems Research, 29*(2), 381–400.
Davenport, T., & Ronanki, R. (2018). Artificial intelligence for the real world. *Harvard Business Review, 96*(1), 108–116.
Dibbern, J., Goles, T., Hirschheim, R. A., & Jayatilaka, B. (2004). Information systems outsourcing: A survey and analysis of the literature. *ACM Sigmis Database: The DATA BASE for Advances in Information Systems, 35*(4), 6–102.
Dibbern, J., Winkler, J., & Heinzl, A. (2008). Explaining Variations in Client Extra Costs between Software Projects Offshored to India. *MIS Quarterly, 32*(2), 333–366.
Dijkman, R., Sprenkels, B., Peeters, T., & Janssen, A. (2015). Business models for the internet of things. *International Journal of Information Management, 35*(6), 672–678.
DiRomualdo, A., & Gurbaxani, V. (1998). Strategic intent for IT outsourcing. *Sloan Management Review* (Summer), 67–80.
Foerderer, J., Kude, T., Mithas, S., & Heinzl, A. (2018). Does platform owner's entry crowd out innovation? Evidence from google photos. *Information Systems Research, 29*(2), 444–460.
Fogarty, D., & Bell, P. (2014). Should you outsource analytics? *MIT Sloan Management Review, 55,* 41–45.
Gallivan, M. J., & Oh, W. (1999). Analyzing IT outsourcing relationships as alliances among multiple clients and vendors. In *Annual International Conference on System Sciences*. Hawaii: IEEE.
Gartner. (2017). Market share analysis: IT outsourcing services, worldwide, 2016.
Gawer, A., & Cusumano, M. A. (2002). *Platform leadership*. Cambridge, MA: Harvard Business School Press.
Geiger, D., & Schader, M. (2014). Personalized task recommendation in crowdsourcing information systems—Current state of the art. *Decision Support Systems, 65,* 3–16.
Gewald, H., & Dibbern, J. (2009). Risks and benefits of business process outsourcing: A study of transaction services in the german banking industry. *Information & Management, 46*(4), 249–257.
Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: The boundary resources model. *Information Systems Journal, 23*(2), 173–192.

Gopal, A., & Koka, B. R. (2012). The asymmetric benefits of relational flexibility: Evidence from software development outsourcing. *MIS Quarterly, 36*(2), 553–576.

Gregory, R. W., Keil, M., Muntermann, J., & Mähring, M. (2015). Paradoxes and the nature of ambidexterity in IT transformation programs. *Information Systems Research, 26*(1), 57–80.

Groover, M. P. (2014). *Fundamentals of modern manufacturing: materials, processes, and systems*. New York: Wiley.

Huber, T. L., Kude, T., & Dibbern, J. (2017). Governance practices in platform ecosystems: Navigating tensions between co-created value and governance costs. *Information Systems Research, 28*(3), 563–584.

IDC. (2019). *Worldwide semiannual digital transformation spending guide*. International Data Corporation (IDC).

Kaganer, E., Carmel, E., Hirschheim, R., & Olsen, T. (2013). *Sloan Management Review, 54*(2), 23–32.

Kakabadse, A., & Kakabadse, N. (2002). Trends in outsourcing: Contrasting USA and Europe. *European Management Journal, 20*(2), 189–198.

Kelly, T. (2002). A brief history of outsourcing. Global Envision.

Lacity, M. C., & Willcocks, L. (2018). *Robotic process and cognitive automation: The next phase*. Stratford-upon-Avon: Steve Brookes Publishing.

Lacity, M. C., & Willcocks, L. P. (2016). A new approach to automating services (58), 40–49.

Lacity, M. C., & Willcocks, L. P. (2001). *Global information technology outsourcing: In search of business advantage*. Chichester: Wiley.

Langley, P., & Simon, H. A. (1995). Applications of machine learning and rule induction. *Communication of the ACM, 38*(11), 55–64.

Linder, J. (2004). *Outsourcing for radical change*. NY: AMACOM.

Lusch, R., & Nambisan, S. (2015). Service innovation: A service-dominant logic perspective. *MIS Quarterly, 39*(1), 155–175.

Mears, J., & Bednarz, A. (2005, May 30). Take it all' outsourcing on the wane.

Oshri, I., Kotlarsky, J., & Gerbasic, A. (2015). Strategic innovation through outsourcing: The role of relational and contractual governance. *The Journal of Strategic Information Systems, 24*(3), 203–216.

Overby, S. (2018). It outsourcing: Multi-Billion-Dollar Mega Deals End in Break Up, CIO, March 31. http://www.cio.com/article/2419347/outsourcing/it-outsourcing--multi-billion-dollar-mega-deals-end-in-break-up.html.

Oshri, I., Dibbern, J., Kotlarsky, J., & Krancher, O. (2019). An information processing view on joint vendor performance in multi-sourcing: The role of the guardian. *Journal of Management Information Systems, 36*(4), 1248–1283.

Pastin, M., & Harrison, J. (1974). Social responsibility in the hollow corporation. *Business & Society Review*, 54.

Rottman, J. W., & Lacity, M. C. (2004). Twenty practices for offshore sourcing. *MIS Quarterly Executive, 3*(3), 117–130.

Rutschi, R. C., & Dibbern, J. (2020). Towards a framework of implementing software robots: Transforming human-executed routines into machines. *ACM SIGMIS Database: The DATA BASE for Advances in Information Systems, 51*(1), 104–128.

Sabherwal, R. (1999). The role of trust in outsourced IS development projects. *Communications of the ACM, 42*(2), 80–86.

Sahay, S., Nicholson, B., & Krishna, S. (2003). *Global IT outsourcing*. Cambridge: Cambridge University Press.

Schmeiss, J., Hoelzle, K., & Tech, R. P. G. (2019). Designing governance mechanisms in platform ecosystems: Addressing the paradox of openness through blockchain technology. *California Management Review*, 62(1), 121–143.

Schneider, S., & Sunyaev, A. (2016). Determinant factors of cloud-sourcing decisions: Reflecting on the IT outsourcing literature in the era of cloud computing. *Journal of Information Technology, 31*(1), 1–31.

Svahn, F., Mathiassen, L., & Lindgren, R. (2017). Embracing digital innovation in incumbent firms: How volvo cars managed competing concerns. *MIS Quarterly, 41*(1), 239–254.
Tiwana, A. (2002). *The knowledge management toolkit: Orchestrating IT, strategy, and knowledge platforms*. Prentice Hall International.
Tiwana, A. (2014). *Platform ecosystems: Aligning architecture, governance, and strategy*. Oxford: Elsevier LTD.
van der Aalst, W. M. P., Bichler, M., & Heinzl, A. (2018). Editorial: Robotic process automation. *Business & Information Systems Engineering, 60,* 269–272.
Venters, W., & Whitley, E. A. (2012). A critical review of cloud computing: researching desires and realities. *Journal of Information Technology (Palgrave Macmillan), 27*(3), 179–197.
Weinhardt, C., Anandasivam, A., Blau, B., Borissov, N., & Meinl, T. (2009). Cloud computing—A classification, business models, and research directions. *Business & Information Systems Engineering, 1*(5), 391–399.
Westerman, G., Bonnet, D., & McAfee, A. (2014). *Leading digital: Turning technology into business transformation*. Cambridge, MA: Harvard Business Review Press.
Willcocks, L., & Lacity, M. (2016). *Service automation: Robots and the future of work*.
Winkler, J. K., Dibbern, J., & Heinzl, A. (2008). The impact of cultural differences in offshore outsourcing-case study results from German-Indian application development projects. *Information Systems Frontiers, 10*(2), 243–258.
Yinghui, Z., Deng, R., Liu, X., & Zheng, D. (2018). Blockchain based efficient and robust fair payment for outsourcing services in cloud computing. *Information Sciences, 462,* 262–277.

Emergent Sourcing Challenges

Software Architecture and Outsourcing Governance: Raising Thoroughbreds Versus Cultivating Schools of Goldfish

Karoly Bozan, Kalle Lyytinen, and Gregory Rose

Abstract Software development projects are in need to graft external expertise and knowledge for multiple reasons and under different governance arrangements. The classic outsourcing literature focuses on integrating such knowledge under conditions where a single application is built by an outside vendor under a detailed contract dictating the process, outcomes and governance of such undertaking. In such a situation the client articulates and shares the entire business logic and system requirements of the application to be built with the vendor as dictated by the mutual contract. The growing popularity of deploying a modular micro-service architecture (MSA) questions some assumptions that underlie the classical software development outsourcing model and its governance. While under MSA software developers on the client and vendor side may continue to work on the core business logic of the whole application, multiple microservices will be outsourced from third-party vendors. Transitioning to MSA and sourcing from multiple vendors with short engagement cycles and under arms-length arrangements introduces new levels of complexity to outsourcing governance. This calls for introducing new governance logics and arrangements, new types of organizing and monitoring of software development teams, and addressing new types of risks introduced by microservice architecture and its stronger coupling with commercial service stacks (such as AWS). We introduce a granular, three-layered outsourcing model to analyze make-or-buy decisions when MSA is deployed and seek to understand its benefits and risks while establishing outsourcing arrangements. In conclusion, we identify outsourcing research challenges introduced by the growing use of MSA in software development.

K. Bozan
Duquesne University, Pittsburgh, PA, USA
e-mail: bozank@duq.edu

K. Lyytinen (✉)
Case Western Reserve University, Cleveland, OH, USA
e-mail: kalle.lyytinen@case.edu

G. Rose
Washington State University, Vancouver, WA, USA
e-mail: grose@wsu.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_2

1 Introduction

Software development organizations (SDO) have been sourcing external knowledge, expertise and resources for over three decades to curtail development, implementation and/or maintenance costs (Dibbern et al. 2009) or to gain strategic benefits (Quinn et al. 1994). Traditionally what is being outsourced in software development identifies two types of outsourcing arrangements: (1) infrastructural service outsourcing where application hosting, raw computing power and networking infrastructure and its management are carried out by third party vendors (Babcock 2010; Durkee 2010; Zhang et al. 2010); and (2) application development services outsourcing, where clients outsource all or a main portion of their application development to a single vendor (Ang and Straub 1998; Cross 1995; Huber 1993; Lacity and Hirschheim 1993; Lacity and Willcocks 1998). In the latter case the entire application and its business logic and related requirements are shared with the vendor due to the tightly coupled dependencies in the application software. Under the outsourcing contract the vendor builds the application or a main portion of it, and delivers the application to the client as dictated in the contract and is often responsible for its continued maintenance. In this article we focus on challenges introduced to this outsourcing model by recent innovations in software architectures, but also recognize how it blends with the first type of arrangements due to advances in software technologies and architectures.

The conventional outsourcing arrangement for application development is best suited for developing complex, tightly coupled software applications comprising of a large set of tight-knitted components (Battin et al. 2001; Sabherwal 2003; Bird et al. 2009). These applications are built for a specific business purpose and are deemed expensive to use, operate and maintain due to their inflexibility and complexity (Eden and Mens 2006; Boehm 1981). The dependencies within the software make it challenging to deviate significantly from the application's initial set of requirements when new business rules and application features need to be added. Such software architecture style is known as a monolith architecture. The word monolith originates from Ancient Greek where it means a single stone. Ergo, most outsourced applications can be described to be sculpted from a single stone with a specific purpose and client dictum. Consequently, operating and maintaining them likens owning an expensive thoroughbred race horse, which calls for special care and where any small problem can prevent it from delivering its purpose—that of racing.

Due to fast changing customer demands, turbulent market dynamics and significant changes in computing infrastructures, SDOs now face unprecedented challenges in developing monolith applications. Like thoroughbreds, they come with significant shortcomings- they break down under small changes, are slow to change, and do not scale up with the new user demand. A feasible technological solution to address this problem has been to re-architect both old and new applications by breaking them down along smaller and more distinct business functions and by organizing their development and governance differently along small services and their orchestration at the application level. Such solutions are called Micro-Service based Architectures (MSA). The recent emergence and popularity of such architectures is founded actually on an old idea of developing a single application by orchestrating together multiple loosely coupled and independent (re-used) software services where each such service is responsible for a small, individual and distinct function (Dragoni et al. 2017; Bucchiarone 2018). Each such (small) service is developed, tested and deployed independently under the assumption that its further design and deployment can be done without affecting other services. Any of these small, nimble, and independent services can therefore be replaced with other services without significant side effects. For this architectural style many have used the School of Goldfish metaphor. Each goldfish is relatively inexpensive and can be replaced without disruption to the school but the school as a whole offers the function of the whole system and can afford the temporary loss of one fish while it is replaced.

The recent success of MSA has been largely predicated on the emergence over the last decade of powerful and versatile cloud service providers. Cloud services over the last decade have progressed from lower level infrastructure and computing services to services that help orchestrate families of specialized services using service stacks that cover data base, networking, virtualization, and middleware functions and services for presentation, integration and business logic. These services (and related stacks) extend the infrastructural services beyond offering raw computing services to higher level orchestration and platform capabilities. In other words, cloud service providers currently do not just provide computing infrastructure-as-a-service (IaaS), but also software-as-a-service (Saas), and platform-as-a-service (PaaS) (Höfer and Karagiannis 2011; Yang and Tate 2012). As a result, the service platforms have recently emerged as rich and versatile collections of specialized microservices. To paraphrase the goldfish metaphor, the cloud services are the pond where the school of goldfish can prosper. While a few large and vertically integrated SDOs can still build their own service platforms and related stacks for certain fields (such as financial services, transportation), most SDOs are likely to source the development and delivery of their microservices and related orchestration on the behemoth and integrated cloud service platforms offering full service stacks. Hence we have few good ponds to swim with the school of goldfish each with their own ecological characteristics.

Several providers of service stacks compete now in the market with somewhat varying core expertise and capability. Well-known and popular service stacks currently are Amazon's AWS, Microsoft's Azure, and Google's Cloud. Google Cloud is known for its deep learning, AI and ML capabilities, while Azure from Microsoft is focused on integrating and offering a wide range of enterprise related services including all Microsoft application integration. Finally, Amazon Web Services (AWS) has the largest variety and depth of scalable services. To give a sense of the range of services offered in such platforms, Microsoft Azure offers now close to 20 different categories of high level business related services covering Analytics, Containers, Mixed Reality AI or Machine Learning. Within each category, up to 50 different software services such as Anomaly Detection, Azure Maps, Load Balancer, containers are offered (Azure Products 2019). These services enable microservice based development. Each cloud service provider also offers and hosts third party microservices (as platforms). Such hosted services are smaller in scope but deeper in specialization and often represent state-of-the art technology solutions built with highly specialized expertise for a particular business domain such as credit card processing, credit ratings and so on.

When a SDO runs and delivers its microservices and related applications on such single cloud platform it reduces interoperability challenges, but increases resource—and platform—dependency. Though multi-platform applications and services, hybrid clouds are emerging (Leitner et al. 2016; Toosi et al. 2018), their long-term sustainability and related governance forms remain poorly understood. Another challenge in service clouds and their service stacks is that service pricing remains non-transparent and unpredictable. It is often founded on complex algorithms which charge based on usage across a wide variety of services. This increases operating cost of software for any SDO using such services- and now it may cover 20% or more of the actual software life-cycle costs and has become a main concern of SDOs (Villamizar et al. 2016).

Governance of software development outsourcing has already been experiencing a shift due to these changes in the application software architecture and operating model. Traditionally, operating the tightly coupled thoroughbred application utilized an evolving computing infrastructure, which could be partially or entirely sourced from third-party providers. The deployment of MSA introduces different kind of granularity, new governance modes and logics. To our knowledge, there are currently no studies highlighting and identifying novel challenges related to how to govern outsourcing application development under MSA. Managing a school of goldfish (MSA) in the context of outsourcing software development will come with unique challenges when compared to governing the traditional governance of raising thoroughbreds (tightly coupled applications). In this essay we seek to identify such novel challenges concerning the outsourcing arrangements and governance under MSA style. Broadly, we are guided by the following questions: *How does MSA influence the way SDOs will need to manage their software development outsourcing in future? How will the overall software development outsourcing landscape be impacted if and when MSA prevails?* We will address these questions by reviewing the state of the art of software development outsourcing and identifying challenges that will emerge in outsourcing arrangements at multiple levels when MSA is deployed. Based on this review we formulate a set of research questions forming a research agenda that articulates new questions related to MSA based governance.

The remainder of the article is organized as follows. In the next section we review extant literature on software development outsourcing and describe gaps related to effects of technical and architectural choices on software development outsourcing. Next, we review the emerging outsourcing practices in the era of microservices and identify key outsourcing decisions and governance challenges at three granular development layers introduced by MSA.

2 Literature Review

Software development outsourcing can be defined as a contract-based development of a software application between a client and vendor organization where a vendor delivers all or part of the client's software requirements for a mutually agreed fee (Ali-Babar et al. 2007; Kern and Willcocks 2000). In the past, large multi-year contracts either for a fixed or variable fee have characterized software outsourcing engagements between clients and vendors (Sahay and Krishna 2003). Such arrangements can also lay a foundation for a long term strategic relationship between a client and a well-established vendor (Quinn et al. 1990, 1994; Fitzgerald and Willcocks 1994; Marcolin 2002; Kedia and Lahiri 2007; Søderberg et al. 2013).

Past software outsourcing literature has been mainly concerned with governing organizationally distributed development of thoroughbreds to the benefit and needs of the client (the outsourcing company). The client-vendor relationship is viewed as a dynamic affiliation where cultural adjustments and sense making are important on both sides (Rai et al. 2009; Su 2015) and which involves sticky knowledge transfer from the client to vendor, and vice versa (Bailey 2008). The outsourcing research has evolved from examining the determinants of initial sourcing decisions to multifaceted analyses of the impact of different contractual structures to highly finessed views of how to govern dynamically outsourcing relationships (Kotlarsky et al. 2018). Most such studies have been concerned with the *social* dimension of outsourcing such as the effects of geographical distance on client-vendor collaboration and coordination (Kotlarsky et al. 2014), and the effects of various client's governance and control decisions on project outcomes (Benaroch et al. 2016), or the impact of contract design for the development of success (Chang and Gurbaxani 2012). Some studies have examined alternative ways of governing long-term engagements between the client and the vendor (Oshri et al. 2018), or how to deal with intellectual property governance (i.e. transfer of expertise and its protection) during outsourcing arrangements (Chen et al. 2017).

The social dimension of client-vendor relationships is currently relatively well understood in the context of monolith application development. Given the relatively fixed nature of the architectural assumptions and style, outsourcing research generally lacks a careful analysis of the impacts of the *technical* and architectural change on organizational aspects of software development outsourcing. The current analyses of outsourcing decisions are typically carried out without any explicit analysis of how the outsourcing of a monolith shapes coordination and governance needs other than the size or volatility of the software development task (the complexity of dependencies to be managed within the software). Early studies on outsourcing, for example, have concluded that lowered transaction costs (Ang et al. 1998) and specific vendor skills (Quinn et al. 1994) drive the build or buy decisions for monolith application development. The buy decision can be followed by a decision of where and how to split a tightly coupled application for separate development across sites and places and to what extent the infrastructural needs of the application may have to be outsourced. The underlying assumption has also been that outsourcing decisions focus on transferring tasks of developing (business) functions of a tightly coupled system to a vendor and trying to understand how to govern related business process (Ross and Beath 2006). Knowledge of application dependencies has been deemed crucial when deciding which business functions and applications can and should be outsourced and which need to be kept in-house. The coordination has primarily focused on how to effectively integrate such monolith applications being developed with other applications (Tanriverdi et al. 2007; Schilling 2000). In line with this thinking Ross and Beath (2006) studied the effects of the maturity of enterprise architecture on the success of outsourcing engagements. Tanriverdi et al. (2007) posit that the success in business function outsourcing is dependent on the modularity and separability of the enterprise architecture components. A proper enterprise architecture with a modular design has been deemed necessary to ensure successful outsourcing governance at the application level (Shi et al. 2005). Enterprise architecture also improves vendor relationship management (Choudhury and Sabherwal 2003; Koh et al. 2004) and impacts positively performance (Ross and Beath 2006; Shi 2010). Overall, this research has largely ignored the role and impact of inner application dependencies on the outsourcing relationships and their governance. A segment of outsourcing research has also analyzed inherent risks of such application development outsourcing including the loss of organizational competencies (Earl 1996; Lacity and Willcocks 1998), rising switching costs (O'Leary 1990), cost escalations and hidden service costs (Lacity and Hirschheim 1993), inability to control technical quality (Sabherwal 2003), and increased uncertainties (Adeleye et al. 2004).

To sum up the extant literature on software development, outsourcing has so far focused on the effects of modularity at the application level and its effects on how to cut the 'cake' for software development outsourcing for improved performance. This focus on between-application-dependencies is insufficient in the context of MSA. Applications are now increasingly built fully or partially by using MSA style where the application gets orchestrated as a collection of independent microservices using 'glue' code between them that introduces dependencies between the services at the application level. Governing outsourcing arrangements therefore needs to be extended to incorporate decisions on the use of MSA and what its impact is on outsourcing governance. Multiple new types of dependencies need to be managed at the service level, if they are outsourced, similar if a specific service stack is deployed. Overall, the governance need to cover service governance, service orchestration, and the impact of deploying a specific set of service stack(s). The interdependencies across these three layers also create a complex and dynamic landscape of outsourcing decisions, which has been largely ignored in past outsourcing literature. Governing the outsourcing of these three layers and the changes they introduce to current software development outsourcing arrangements as well as their long terms effects on software development outsourcing governance, are poorly understood. We will next highlight some of these new research challenges

3 MSA Driven Outsourcing

Generally, cloud computing opens up opportunities for outsourcing for both thoroughbred and goldfish. Both types of applications assume the presence of some infrastructural services. Currently the thoroughbred type applications are more common. Traditionally, a loose coupling has prevailed between the application and computing infrastructure layers when using thoroughbred applications. Therefore, porting such applications to the cloud has been relatively simple as appropriate standardization is in place at the level of code interoperability and data base portability. Generally, the code base can be run on the same database (SQL) and networking standards (TCP/IP) on any cloud provider's infrastructure (Papazoglou 2012).

Unlike thoroughbreds, MSA style applications assume additional ecological 'elements' that allow schools of goldfish to live and grow. Microservices need to be now developed, hosted and sold on specific service stacks with related services. At this level different cloud service providers come with different orchestration capabilities and the microservices developed on one platform typically has native interoperability only with microservices developed on the same platform. A shift to MSA, hence, results in much tighter coupling at the ecological level and produces new types of dependencies between the application and cloud service layers. Orchestration is however more effective and comes with high velocity and is now only a feasible alternative on a common single cloud platform. When SDOs outsource microservices, they need to be therefore aware of these new dependencies where the chosen service-stacks creates vertically a tightly coupled ecology while the services at the application level can become horizontally more loosely coupled. Metaphorically, the goldfish can only swim in one pond at a time as a school.

We next review outsourcing arrangements for each layer separately and discuss in this context the build versus buy decision criteria for services at that layer. Such decisions were chosen as the focal unit of analysis, because they impact greatly the goals and logic of outsourcing governance (Badampudi et al. 2016; Petersen et al. 2017). Due to the increasing volume and variety of arm's length contracts while using MSA this will result in increased multi-vendor interactions and grow the complexity of outsourcing arrangements. The combination of built and outsourced functions and their dynamic relationship will emerge as a pivotal research issue. We also discuss the benefits and risks of each buy or make decisions. This discussion identifies pertinent research issues and suggests that outsourcing scholars may need to think differently about outsourcing governance in future given the new level of service granularity and new types of dependencies produced in MSA based outsourcing. Table 1 summarizes these themes across the three layers.

Table 1 Three-layer MSA outsourcing decisions

Levels of outsourcing decision in MSA	Make or buy	Risk	Benefit	Key research issues
Microservice layer	Make: core business service—unique, high customization, domain expertise	Rapidly applying new skills, resource constraints, culture shock	Faster development, focus remain on core competency, long term cost, clear ownership, best fitting technology	Building up expertise—will it be core competence? Shift is dynamic between make/buy
	Buy: Generic, common business functions Competitive necessities suite of microservcies)	Vendor dependence, weakening skills, organizational "forgetting", higher cost, data sharing/ownership, data standards for diverse technology, complex vendor management	Faster development, freed from maintenance, more secure	Which service to outsource Service guarantee Quality guarantee Challenges of replacing vendor Outsource development, in-house maintenance Managing the increasing number of arm's length contracts Diversity of contracts (volume and type) Catalog for reuse Quality criteria
Application orchestration layer	Make: core competency Buy: can't support this new role Not economic to maintain these skills	Loss of control Agency problems Information asymmetry Resource dependency	Faster development Control	Variability across applications Client involvement Who has transfer of control? One or multiple vendors? Transparency of orchestration (buy) Managing multi-level contracts (subcontracts) Governance of orchestration when outsourced
Service stack layer	Make: under what conditions do you build Buy: services from the same MS Stack provider (platform)	Interoperability if in-between platforms, resource dependency, increasing operating cost - pay per service, accountability, lack of transparency Thinning of tech expertise	Interoperability if within same platforms, Maintenance Scalability Technical expedience Reliability State of the art technology Access to host of services (network effect) Security	Issue of bond: which MS stack to use? Within or in-between service usage? Governing compliance across vendors (regulated industries)

4 Microservice Level

Specialized skills and deep expertise are instrumental in building high quality microservices responsible now for narrow, dedicated business functions. There are several reasons why an SDO would build a microservice internally, versus why it would outsource these parts of the application. The organization should focus on its core competencies building the main business functions with unique business logics requiring unique solutions and high levels of customization and domain knowledge. Generally, the domain expertise for rare, unique and difficult to imitate business logics should be kept "in-house" as a competitive advantage (Bakshi 2017). The long term cost of such a solution may be better justified through internally built microservices as multiple "pay-per-use" services can add significant variable cost to the software operation. If the service is internally developed, the best fitting technology can be applied. At the same time it requires to build related internal technical expertise. If the tightly coupled thoroughbred application is gradually transitioned into MSA type application, skills to decompose related functions and building scalable microservices can be built incrementally by examining the strengths and weaknesses of current software assets. However, development methodologies and practices related to MSA deployment may challenge the introduction and proper internal governance of software assets and calls for outsourcing.

Generic, non-core items or lower-level common business functions, such as credit card payment, product rating are recommended to be outsourced. A suite of microservices which form competitive necessities should also be outsourced as they usually do not represent the firm's core competency. For example, online banking is expected by a bank customer and banks should not build such features from scratch internally. Rather they should outsource those to experts who have experience in building and maintaining a suite of highly secure and scalable microservices. These third party microservices accelerate application development because internal resources can focus on improving core expertise. However, some outsourced microservices can become expensive to use in the long run and create vendor dependencies. They can also result in "organizational forgetting" of key aspects of business process.

The microservice layer of outsourcing raises several areas future research. The decision between building and outsourcing a particular microservice is likely to change over time. One main research challenge is to understand how SDOs can identify critical services to outsource or insource. Core competencies will shift and in-house built application may become obsolete when an outsourced alternative is more advanced and can be better maintained. There is no clear rule that can withstand extended period of time between build or buy micorservice decisions due to the dynamic nature of current business strategies and a growing number of specialized vendors (Hackett 2019). Moving outsourced microservice to in-house creates new challenges as previously not utilized skills need to be now built up internally. Similarly, outsourcing internally built microservices may have significant impact on the team learning which previously owned that service. These situations have analogies with existing research which has analyzed nearshoring effects or the impact of insourcing applications development in-house for quality, competitive, or regulatory reasons (Tafti 2005; Bryan 2016; O'Byrne 2018).

- *Core competencies and governance of microservices development*: How can SDOs better understand what continues to be their core competency and what will become obsolete? What areas they should focus on in the future? For which types of applications can an SDO claim core competency and developed related microservices? How can SDOs identify the services to outsource or insource? What are the impact on the organization and application of moving previously outsourced microservices back to in-house?

Outsourcing microservices raises the concern of quality and the existence of sustained support and maintenance for the service. While a number of alternatives exist to address these concerns for most business functions, the features important for a SDO to make decisions on such things is limited to a small number of vendors. Their longevity and willingness to maintain and update the microservice is difficult to assess, especially with recently established vendors. Challenges associated with replacing a vendor is another emerging concern, which was not common in the traditional outsourcing arrangements. Due to the presence of loose coupling, a microservice should be easy to replace but the implications of such replacement are not well understood. For example, microservices often store their own data in different formats (Messina et al. 2016) and data ownership is later challenging to establish when dealing an increased number and variety of vendor contracts. We can reasonably expect that specialized vendors will be bought increasingly by larger competitors to grow customer base and domain knowledge. This may affect the services used in many applications of the SDO.

- *Governance challenges across multiple vendors and services on multiple levels*: How can a SDO ensure service guarantees around microservices from vendors? If a vendor goes out of business or stops maintaining the service, or the product becomes orphaned, what happens with client data? What challenges are present when client wants to replace a vendor or a service (change a goldfish in the school)? How can SDOs evaluate the quality guarantee of microservices? If a microservice is outsourced, under what circumstances should an SDO consider to maintain it in-house?

When SDOs choose to outsource microservices to multiple vendors, a variety and large number of service level agreements (SLA) and arm's length contracts need to be governed (Costa et al. 2014). The vendors may have different terms and certifications, which make it increasingly challenging to enforce varying licensing agreements. Working with one or few vendors who provide the required microservices and maintain them simplifies the governance arrangements, but it may limit the choices and functionalities offered by services. Such arrangement can come also with a premium price. Such arrangements allow the client to have a single or few points of contact if complications arise with the offered services.

- *Governance of high volume of microservices and arm's length contracts*: Under what circumstances should SDOs outsource to multiple vendors or seek to source from few? How to govern a variety of SLA and arm's length contracts?

The different combination of internally developed and microservices pose a challenge when SDOs want to reuse them across applications. The coordination of similar services across multiple applications pose a new kind of many-to-many relationship between applications and services, which results in increasingly complex governance decisions.

- *Governance of microservice reuse*: How should SDOs catalog the diverse outsourced microservices to enable effective reuse? Should SDO reuse outsourced services across applications or have a new outsourced instance used for the same service? Are the quality criteria for deployment of the same outsourced service across applications?

5 Application Orchestration Level

Software applications built using MSA assume that developers weave together multiple microservices to perform specific tasks in an established, 'orchestrated' order. Orchestration is accountable for calling the right microservices in the right order and roll back, if the service fails. Therefore, application orchestration is responsible to maintain the lifecycle of any business transactions and call proper microservices both in the prepare and commit phases (Alam et al. 2018). Building this orchestration service requires novel architectural skillsets to ensure that the application is configurable when microservcies are added or changed. The design should also be simple in so that developers can trace failure points and recover the application if it fails to proceed properly.

SDOs can orchestrate microservices by themselves or they can outsource partially or entirely this task. The expertise for orchestration can be built in-house, if application orchestration for this specific domain is considered a core competency for the organization or the organization owns a large number of microservices in that domain which allow highly dedicated and scalable orchestration within the organization. Orchestration may need to be kept in house, if high velocity or local responsiveness is expected- especially on topics which are close to competitive needs. Such skills however, may be costly to maintain if the orchestration assumes high level technical skills and deep domain knowledge, which are often hard to maintain internally. This may warrant also for outsourcing the maintenance on-demand. Specific business interests, little concern for speed but for development scalability and the current lack of skills can also be good reasons for outsourcing orchestration as offered to us during our field studies. However, bases of the decisions concerning orchestration outsourcing need to be further studied.

- *Governance of make or buy decisions around orchestration*: How do SDOs determine whether orchestration for specific application is their core competency? Does it differ among different types of applications? When external vendors are chosen to orchestrate, should one vendor be tasked with the orchestration for the entire application, or across applications, or should this task be distributed among multiple vendors? Should the client be involved in orchestrating to alleviate loss or transfer of control? Would vendors who develop the microservice used for an application be responsible for orchestration? Under what circumstances would an SDO encourage multi-level contract based application orchestration as one form of outsourcing?

Information asymmetry during outsourced orchestration will create agency problems where the vendor is likely to act in its own advantage. Such problems will arise from the knowledge and goal differences between the client and the vendor making each side to feel vulnerable to opportunism or shirking of responsibilities by the other (Lacity and Hirschheim 1993; McFarlan and Nolan 1995). As a result, the sides may coordinate projects in opposing directions (Heiskanen et al. 1996). Therefore, transparency of outsourced orchestration is vital when and internal teams should be involved in orchestration activities to an extent. However, monitoring high velocity orchestration may be very different from traditional monitoring of software vendors

- *Monitoring of vendor orchestration:* Are the forms of monitoring and incentivizing vendor based orchestration different from traditional forms of monitoring vendor performance? How do agency problems differ in high velocity environments characterized by service orchestration?

6 Service Stack

The service stack provides a computing platform for application developers to build, host, and introduce their software products to their users and/or marketplace. Due to their two sides nature cloud platforms have also positive indirect network effects as offering more microservices on the platform increases the value of the service stack for all participating SDOs. The technical expedience of the platform provider also gives guarantees for reliability, scalability, maintenance, and interoperability of native microservices (Grozev and Buyya 2014). Such interoperability may also be viewed as a risk in engaging with a service stack as orchestrating microservices in-between service stacks is still complex undertaking to the extent that SDOs may be "locked in" (Petcu 2013; Opara-Martins et al. 2015) limiting options to the service offerings available on the chosen service stack (Mezgar and Rauschecker 2014). Governing the issue of how isolated the 'pond' is resulting in service stack exclusivity is a new challenge to consider in outsourcing governance. This echoes similarities with the strong and broad standard wars of the 80s and 90s between different operating system ecosystems (Apple vs. Microsoft; IBM vs. DEC). Generally, we need to better understand how SDOs choose which service stacks they will use and for what purposes and how this shapes their outsourcing governance.

- *Governance of service stack provider:* What are the main decision factors to consider for SDOs when they choose between service stacks? Are the provider's core technical competency, or price structure, or various hardware/software combinations (IBM, Oracle, for example) or the availability of specific microservices the main driver for an SDO in selecting a service stack? Do SDOs consider complementary services within a platform in case one needs to be replaced?

To what extent do the long term viability and SLA features influence such decisions? To what extent do competitive logics (e.g. Walmart selecting Microsoft's Azure) and the path dependencies within the firm's current computing infrastructure influence these decisions?

Governing compliance across vendors, especially in the regulated industries, has emerged as an additional challenge when multi-vendor contracts are executed under MSA. Security, for example, can become a double-edged sword in that it should be cared for by the cloud service vendor, but the vendor may not confirm to the same standards as the SDOs in a given industry. To overcome such risk, groups of SDOs in regulated industries have collectively decided to introduce common security frameworks, which are built as amalgams of federal, state, and industry requirements. Such frameworks are certified by independent third-party assessors for vendor compliance (Business Wire 2018). For example, health insurance providers use common requirements for their cloud vendors who access protected health information (https://hitrustalliance.net/). This reduces the client's burden and also that of the cloud provider who only has to comply once while several clients accept the same certifications. Yet, it is important to better understand how this impacts the cloud providers and deployment of MSA and how they need to alter their service offerings under MSA, and whether need for higher compliance impacts the software development performance under MSA. Furthermore, what are the long term impacts of inter-platform service usage and how governance of compliance across vendors, especially in regulated industries, impact the application development and outsourcing decisions?

- *Governance challenges of vendor compliance*: How can SDOs ensure that third party vendors at the service and orchestration levels are capable to comply with internal regulatory requirements? Will these requirements change and vendors will accommodate and make appropriate updates to the stack? Will service quality be impacted if inter-platform services need to be added?

The thinning technical expertise as a result of outsourcing all cloud services to the external vendor raises issues for SDOs in terms of new and complex resource dependencies. For example, database administrator skills will be less needed if the whole task is outsourced to cloud service provider. But it will be difficult to bring such service back, when the data governance and performance becomes an issue. Similarly, if all technical networking experts leave the SDO, new employees may soon only know how higher levels of the service stack (so called lambda functions) work as expressed by the cloud service provider, but they will not have a deeper understanding of all back-end operations at the infrastructure level. SDOs therefore need to face critical decisions of what level of outsourced expertise they can maintain internally.

- *Governing outsourced skills*: How can SDOs identify core technical skills that need not be maintained internally? What is the short and long term impact of new resource dependencies created by outsourcing technical operations and related expertise? Under what circumstances would a SDO decide to insources technical skills that are necessary to operate the stack? What skill inertia can be expected within and between applications if skills need to be rebuilt within the SDO? To what extent does cost/risk ratios influence resource dependency decisions?

7 Interaction Across the Three Layers

The change of architecture styles from thoroughbreds to schools of goldfishes will produce new ecologies of software development with their own outsourcing arrangements. Governing outsourcing in this type of ecological environment requires better understanding of interactions between the three vertical layers. The MSA style creates new dependencies across the layers and is a manifestation of a new type of architectural policies where SDOs experience laterally looser couplings, velocity and flexibility but vertically they are anchored more tightly to specific ecologies. The effect of orchestrating multiple microservices under an outsourcing arrangement on such a service stack (pond) is overall poorly understood in terms of how to arrange and coordinate dependencies between the levels.

- *Governing lateral and vertical balance*: To what extent are SDOs aware and constrained by the new vertical inflexibility of outsourcing under MSA? Under what conditions and for what types of applications the new vertical constraints will offset the benefits of increased lateral flexibility and velocity? What are the impacts of the limitations of vertical coupling on governing application design and related performance?

8 Discussion

Outsourcing has been an area of interest since the first software projects were built. All types of software and their enabling technologies have created a constant need to re-evaluate the idiosyncratic attributes of these technologies and their effects for managing software development in SDOs. MSA is yet another permutation of this phenomenon and it is likely to have significant impact on the organization, process characteristics and division of labor related to software development. It is also likely to increase the range of software based activities which are mediated by market based mechanisms as shown in this chapter. The characteristics of MSA offer fascinating opportunities and new risks and challenges for SDOs and open many new and old areas of software development and its management for re-investigation.

Through this essay, we have sought to identify what has and is likely to change in the outsourcing arena when architecture style of applications shift from raising tightly coupled thoroughbreds to the nimble and modular schools of goldfish. Our analyses offer rationales for why outsourcing with MSA will be a novel area and what new challenges the research community needs to address in the future. The traditional focus solely on the social aspect of outsourcing is shifting toward understanding how technical design decisions and architectural choices impact outsourcing governance with unique and previously disregarded consequences. Furthermore, the architectural shift will result in a move from a previous two 'layer' outsourcing logic to a more complex three layered outsourcing logic with more complex horizontal and vertical dependencies. The impacts of this change are manifold. Unlike in traditional outsourcing arrangements, SDOs no longer face the burden of deciding to what extent the infrastructure should be outsourced, and deciding where to split the application for partial outsourcing and then integrating the completed work back into the application with complex internal dependencies. In contrast, they are freed from such constraints embedded in a two-layer outsourcing world. These freedoms include a call for a clearer understanding of service boundaries, ownership, increased reliability, scalability, and presence of technical expedience that increases velocity.

At the same time the new freedoms for SDO outsourcing in using MSA comes with severe new problems and the management of MSA based outsourcing is ripe with new challenges. First and foremost, deciding which microservice should be developed in-house versus which to outsource has to be strategically governed. It cannot be left for developers to decide as it has great impact on the assembly of internal skills and team and cost structures. Secondly, choosing the service stack where to host or buy microservcies will have a significant impact on orchestration capabilities and complexity and related cost. Finally, finding the right balance of internally custom built and sourced microservcies will have a great impact on cost and governance complexity. The interaction of three layers introduce a new type of vertical dependency, which SDO management has to be aware before fully committing to MSA in hope for greater flexibility and speed.

The three-layer outsourcing model will exponentially increase the decision tree branches related to outsourcing decisions. Currently we lack a clear understanding of what conditions warrant for each choice and their long term impact on how organizations manage software assets. The architectural configuration options are now more granular, to which SDO leadership needs to become accustomed, if they want to minimize adverse effects of growing complexity. Our study highlights some major options and research directions that need to be addressed to better understand the new and still partially unknown challenges in the new emerging software development landscape.

As with all "call to arms" research agenda papers, we observe limitations in our treatment. Our analysis is based on our own observations and notes that emerged during our field work in several organizations assimilating MSA type development practices and then triangulating these observations with what is known of software development outsourcing arrangements in small but growing MSA literature and then juxtaposing it with the established stream of software development outsourcing research. Such broad 'scouting' of the lay of the land comes with limitation and this article is meant to open a stream of deeper empirical investigations and theory development. After theoretical and speculative review conducted here a field based qualitative research is generally accepted as the next appropriate next step to address research questions being raised. Inductive, theory generating case studies founded on semi-structured interview and open-coding and grounded method appear therefore as an appropriate next step. The issues raised here suggest that outsourcing under MSA forms a radical shift that calls for understanding based on idiographic and detailed assessments of observed changes brought by MSA. The proposed research project should theoretically sample large and small software firms working across all three layers with varying outsourcing arrangements. Ages of SDOs and their clients need to be considered to detect differences in the willingness, capacity, and extent to adopt MSA and elated outsourcing arrangements. This helps tease out differences in the willingness and capacity to be MSA native and identify key challenges shifting from monolith to MSA development and related outsourcing arrangements.

Qualitative research historically is best suited for inductive, theory building research. Many avenues are ripe for extending existing outsourcing research to the new world of MSA outsourcing. Established theories of governance and its organizational, social, and cognitive dimensions have applied to understand client-vendor relationships in software development outsourcing. It will be fascinating to see, if established theoretical models will hold up within the more complex MSA outsourcing arrangements where new forms of arms-length, and complex infrastructural arrangements dominate and where flexibility and velocity are emphasized at the level of orchestration. This will help also validate to what extent extant research from salient areas outside of software engineering and IS can be integrated to established outsourcing theory and tested quantitatively.

References

Adeleye, B. C., Annansingh, F., & Nunes, M. P. (2004). Risk management practices in IS outsourcing: an investigation into commercial banks in Nigeria. *International Journal of Information Management, 24*(20), 167–180.

Alam, M., Rufino, J., Ferreira, J., Ahmed, S. H., Shah, N., & Chen, Y. (2018). Orchestration of microservices for IoT using Docker and edge computing. *IEEE Communications Magazine, 56,* 118–123.

Ali-Babar, M., Verner, J., & Nguyen, P. (2007). Establishing and maintaining trust in software outsourcing relationships: An empirical investigation. *The Journal of Systems and Software, 80,* 1438–1449.

Ang, S., & Straub, D. W. (1998). Production and transaction economies in IS outsourcing: A study of the US banking industry. *MIS Quarterly, 22*(4), 535–552.

Azure Products (2019). Retrieved from https://azure.microsoft.com/en-us/services/.

Babcock, C. (2010). *Management strategies for the cloud revolution: How cloud computing is transforming business and why you can't afford to be left behind*. New York: McGraw-Hill.
Badampudi, D., Wohlin, C., & Petersen, K. (2016). Software component decision making: In house, OSS, COTS or outsourcing—A systematic literature review. *Journal of Systems and Software, 121,* 105–124.
Bakshi, K. (2017). Microservices-based software architecture and approaches. *In 2017 IEEE Aerospace Conference. IEEE*, pp. 1–8.
Battin, R. D., Crocker, R., Kreidler, J., & Subramanian, K. (2001). Leveraging resources in global software development. *IEEE Software, 18*(2), 70–77.
Benaroch, M., Lichtenstein, Y. & Fink, L. (2016). Contract design choices and the balance of ex ante and ex post transaction costs in software development outsourcing. *MIS Quarterly, 40*(1), 57–82.
Bird, C., Nagappan N., Devanbu P., Gall H. & Murphy, B. (2009). Does distributed development affect software quality? An empirical case study of windows vista. *In ICSE '09: Proceedings of the 2009 IEEE 31st International Conference on Software Engineering*, *IEEE Computer Society*, 518–528.
Boehm, B. (1981). *Software engineering economics*. Englewood Cliffs: Prentice-Hall.
Bryan, G. (2016). Why are so many organizations bringing outsourced IT back in-house? *Computer Weekly.*
Bucchiarone, A., Dragoni, N., Dustdar, S., Larsen, S., & Mazzara, M. (2018). From monolithic to microservices: An experience report from the banking domain. *IEEE Software, 35*(3), 50–55.
Business Wire (2018). Nation's Health CISOs Take Lead to Manage Third-Party Risk. *Business Wire.*
Chang, Y. B. & Gurbaxani, V. (2012). Information technology outsourcing, knowledge transfer, and firm productivity: an empirical analysis. *MIS Quarterly, 36*(4), 1043–1063.
Chen, Y., Bharadwaj, A. & Goh, K. Y. (2017). An empirical analysis of intellectual property rights sharing in software development outsourcing. *MIS Quarterly, 41*(4), 131–161.
Choudhury, V., & Sabherwal, R. (2003). Portfolio of control in outsourced software development projects. *Information Systems Research, 14*(3), 291–314.
Costa, B., Pires, P. F., Delicato, F. C., & Merson, P. (2014). Evaluating REST Architectures Approach, Tooling and Guidelines. *Journal of Systems and Software, 112,* 156–180.
Cross, J. (1995). IT outsourcing: British Petroleum's competitive approach. *Harvard Business Review, 73,* 94–102.
Dibbern, J. & Heinzl, A. (2009). *Business & Information Systems Engineering, 1*(1), 101–110.
Dragoni, N., Mazzara, M., Giallorenzo, S., Montesi, F., Lluch Lafuente, A., Mustafin, R., et al. (2017). *Microservices: yesterday, today, and tomorrow*. Springer, Berlin Heidelberg: In Present and Ulterior Software Engineering.
Durkee, D. (2010). Why Cloud Computing Will Never Be Free. *Communications of the ACM, 53*(5), 62–69.
Eden, A. H. & Mens, T. (2006). Measuring software flexibility. *IEE Proc.-Softw*., 153(3).
Fitzgerald, D. & Willcocks, L. (1994). Outsourcing information technology: contracts and the client/vendor relationship. *Oxford Institute of Information Management*, 1–20.
Grozev, N & Buyya, R. (2014). Inter-Cloud architectures and application brokering: taxonomy and survey. *Software—Practice & Experience*, 44(3), 369–390.
Hackett, R. (2019). Buying Stock in Tech Building Blocks. *Forbes.*
Heiskanen, A., Newman, M., & Similä, J. (1996). Software Contracting: A Process Model Approach. In: *ICIS 1996 Proceedings*, p. 3.
Höfer, C., & Karagiannis, G. (2011). Cloud computing services: taxonomy and comparison. *J Internet Serv Appl, 2*(2), 81–94.
Huber, R. L. (1993). How continental bank outsourced its crown jewels. *Harvard Business Review, 71,* 121–129.
Kedia, B. L., & Lahiri, S. (2007). International Outsourcing of Services: a partnership model. *Journal of International Management, 13,* 22–37.

Kern, T., & Willcocks, L. (2000). Exploring information technology outsourcing relationships: theory and practice. *Journal of Strategic Information Systems, 9,* 321–350.

Koh, C., Ang, S., & Straub, D. W. (2004). IT outsourcing success: a psychological contract perspective. *Information Systems Research, 15*(4), 356–373.

Kotlarsky, J., Oshri, I., Dibbern, J., & Mani, D. (2018). IS Sourcing. *MIS Quarterly Research Curations*. Retrieved from http://misq.org/research-curations.

Kotlarsky, J., Scarbrough, H. & Oshri, I. (2014). Coordinating expertise across knowledge Boundaries in offshore-outsourcing projects: the role of codification. *MIS Quarterly, 38*(2), 607–627

Lacity, M. C., & Hirschheim, R. (1993). *Information systems outsourcing*. Chichester, UK: John Wiley.

Lacity, M. C., & Willcocks, L. P. (1998). An empirical investigation of information technology sourcing practices: lessons from experience. *MIS Quarterly, 22*(3), 363–408.

Leitner, P., Cito, J. & Stöckli, E. (2016). Modelling and managing deployment costs of microservice-based cloud applications. In *2016 IEEE/ACM 9th International Conference on Utility and Cloud Computing (UCC)*, 165–174.

Marcolin, B. L. (2002). Spiraling Effect of IS Outsourcing Contract Interpretations. In R. Hirschheim, A. Heinzl, & J. Dibbern (Eds.), *Information Systems Outsourcing*. Heidelberg: Springer, Berlin, pp. 223–256.

McFarlan, F. W., & Nolan, R. L. (1995). How to manage an IT outsourcing alliance. *Sloan Management Review, 36*(2), 9+.

Messina, A., Rizzo, R., Storniolo, P. & Urso, A. (2016). A simplified database pattern for the microservice architecture. In: *The Eighth International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA)*, pp. 35–40.

Mezgar, I., & Rauschecker, U. (2014). The challenge of networked enterprises for cloud computing interoperability. *Computers in Industry, 65*(4), 657–674.

O'Byrne, R. (2018). 8 Reasons Companies Bring Outsourced Operations Back In-house. *Logistics Bureau.*

Opara-Martins, J., Sahandi, R., & Tian, F. (2015). Critical review of vendor lock-in and its impact on adoption of cloud computing. *International Conference on Information Society, i-Society*, 92–97.

Papazoglou, M.P. (2012). Cloud Blueprints for Integrating and Managing Cloud Federations. *In: Heisel M. (eds) Software Service and Application Engineering. Lecture Notes in Computer Science*, 7365.

Petcu, D. (2013). Multi-Cloud: Expectations and Current Approaches. *In Proceedings of the 2013 International Workshop on Multi-cloud Applications and Federated Clouds, MultiCloud '13*, 1–6.

Petersen, K., et al. (2017). Choosing Component Origins for Software Intensive Systems: In-house, COTS, OSS or Outsourcing? - A Case Survey. *Transactions on Software Engineering, 99,* 1.

Quinn, J. B., & Hilmer, G. (1994). Strategic Outsourcing. *Sloan Management Review, 35*(4), 43–55.

Quinn, J.B., Doorley, T.L. & Paquette, P.C. (1990). Technology in Services: Rethinking Strategic Focus. *Sloan Management Review*, 79–87.

Ross, J., & Beath, C. M. (2006). Sustainable IT outsourcing success: let enterprise architecture be your guide. *MIS Quarterly Executive, 5*(4), 181–192.

Sabherwal, R. (2003). The evolution of coordination in outsourced software development projects: A comparison of client and vendor perspectives. *Information and Organization, 13,* 153–202.

Sahay, S., Nicholson B. & Krishna, S. (2003). Global IT Outsourcing. *Cambridge University Press.*

Schilling, M. A. (2000). Toward a general modular systems theory and its application to interfirm product modularity. *Academy of Management Review, 25*(2), 312–334.

Shi, Z. (2010). The role of IS architecture planning in enhancing IS outsourcing's impact on IS performance: its antecedents and an empirical test. *Journal of Enterprise Information Management, 23*(4), 439–465.

Shi, Z., Kunnathur, A. S., & Ragu-Nathan, T. S. (2005). IS outsourcing management competence dimensions: instrument development and relationship exploration. *Information & Management, 42*(6), 901–19.

Søderberg, A.-M., Krishna, S., & Bjørn, P. (2013). Global Software Development: Commitment, Trust and Cultural Sensitivity in Strategic Partnerships. *Journal of International Management, 19*(4), 347–361.

Tafti, M. H. A. (2005). Risks factors associated with offshore IT outsourcing. *Industrial Management & Data Systems, 105*(5), 549–560.

Tanriverdi, H., Konana, P., & Ge, L. (2007). The choice of sourcing mechanisms for business processes. *Information Systems Research, 18*(3), 280–99.

Toosi, A. N., Sinnott, R. O., & Buyya, R. (2018). Resource provisioning for data-intensive applications with deadline constraints on hybrid clouds using Aneka. *Future Generation Computer Systems, 79,* 765–775.

Villamizar, M., Garcs, O., Ochoa, L., Castro, H., Salamanca, L., Verano, M., Casallas, R., Gil, S., Valencia, C., Zambrano, A. & Lang, M. (2016). Infrastructure Cost Comparison of Running Web Applications in the Cloud Using AWS Lambda and Monolithic and Microservice Architectures. In: *16th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pp. 179–182.

Yang, H., & Tate, M. (2012). A descriptive literature review and classification of cloud computing research. *Commun. Assoc. Inf. Syst., 31,* 2.

Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud Computing: State-of-the-art and research challenges. *Journal of Internet Service Applications, 1*(1), 7–18.

The Outsourcing of IT Governance

Benoit A. Aubert and Suzanne Rivard

Abstract For the last 50 years, IT researchers have investigated IT governance; the apparatus of structure, processes, and relational capabilities needed to ensure expected benefits are generated from IT resources. This include looking at the structure of IT departments, the use of outsourcing, the role of overseers, etc. Notwithstanding accumulated knowledge, IT governance as an object of study is changing and this is challenging our understanding of governance. Recent changes in the environment have created a context where an increasingly large part of the decisions related to IT governance are now made by instances outside the firm. These changes include outsourcing, offshoring, growth of platforms, IT services evolution, as well as new regulations and standards. We argue that focusing on governance structures may lead to being concerned with only a small portion of the IT resources used by an organization. In contrast, a focus on governance processes is likely to offer a broader reach for governance purposes.

1 Introduction

Since the inception of computers for organizational data processing 50 years ago, IT research has been investigating how to best leverage IT resources (Henderson and Venkatraman 1999). A significant portion of that research has focused on IT governance, the apparatus of structure, processes, and relational capabilities needed to ensure expected benefits are generated from IT resources. Researchers and practitioners[1] alike have been interested in the structure of IT departments, the use of

[1] http://www.isaca.org/cobit/pages/default.aspx.

B. A. Aubert (✉)
Rowe School of Business, Dalhousie University, Halifax, Canada
e-mail: benoit.aubert@dal.ca

S. Rivard
Department of Information Technology, HEC Montréal, Montréal, Canada
e-mail: suzanne.rivard@hec.ca

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_3

outsourcing, the role of overseers, etc. Given those important efforts, one could assume that we now have a better understanding of IT governance.

It is the thesis of this paper that, notwithstanding accumulated knowledge, IT governance as an object of study is changing and this is challenging our understanding of governance. Recent changes in the environment have created a context where an increasingly large part of the decisions related to IT governance are now made by instances outside the firm. This could be labelled an outsourcing-of-IT governance. In this new context, the current predominant role of structure in governance might not be as critical as the role of appropriate governance processes. Indeed, it is argued that focusing on governance structures may lead to being concerned with only a small portion of the IT resources used by an organization. In contrast, a focus on governance processes is likely to offer a broader reach for governance purposes. Processes can encompass elements controlled directly by the firm and elements for which the firm is using suppliers' assets.

The following example (Markus 2017) shows how the IT environment transformation leads to unintended consequences and a loss of control over decisions. Although the example was originally aimed at illustrating the consequences of work automation, it offers a parallel with similar changes happening to IT governance discussed in this paper.

> An automated mortgage underwriting software was originally built by Fannie Mae and Freddie Mac (mortgage securitizers) to support the work of expert underwriters who were working in mortgage banks. These experts could decide whether or not they followed the advice of the software when doing their work (Straka 2000 cited in Markus 2017). Automation gradually gained acceptance. It was easier to accept a software recommendation than to challenge it. Subsequent outsourcing of mortgage underwriting further increased adoption of automated decisions to minimize the cost associated with hourly rates of contract underwriters. Later, the software was embedded directly in loan software systems, so the use of the automated mortgage underwriting software was now done implicitly by the loan officers when they were using the loan software. A final evolution happened when the software was integrated in websites. Then it became used by real estate brokers and potential borrowers. These numerous shifts mean that the incentives of the "users" were very different at each step since users themselves were very different at each step. This can compromise the use of the technology. It also mean that users could at time be individuals or organizations (through other software) (Markus 2017). The example illustrates how the overseeing and usage of the software changed according to outsourcing decisions, technology evolution, or web platform creation.

This example is interesting because the loss of control over the underwriting decision seemed to happen almost invisibly. At no point there was a conscious decision to give away the decision authority. a manual decision became an IT supported one, which was then automated completely. The resulting software was transferred, integrated into other applications and platforms, and in the end it is not entirely clear who controls the software or the business decision coded in the software. Responsibilities seem to dissolve along the chain of events. As described in the following pages, an analogous evolution pattern seems to apply to IT governance and associated IT decisions.

We suggest that five important trends contribute to the erosion of control over IT governance in contemporary organizations: outsourcing, offshoring, platforms, IT activities conducted outside the IT department, and increased regulations.

- Outsourcing has blurred many traditional organizational boundaries. Companies are now embedding in their processes and activities people, data, apps, and other resources coming from other organizations. Contracting is also becoming increasingly dynamic, suggesting that what a company has included in its process could be provided by different suppliers over time, even if this change is not always visible.
- Offshoring has blurred geographical boundaries. Company activities are conducted in multiple countries, therefore being subject to different jurisdictions, different intellectual property regimes, different human resources constraints, etc.
- An increasing number of IT services are offered through platforms. This provides rapid access to resources, but it constrains interactions though specific interfaces, following specific rules. Positive network externalities and rapid access to innovation are often seen as outweighing the possible disadvantages associated with the control ceded to the platform operators.
- IT evolution, notably through the consumerization of IT, has blurred the boundaries of IT departments. The IT expenses controlled by IT departments only represent a fraction of the IT used in an organization. Other departments access Software as a Service (SaaS) without seeing it as an IT component. It is simply seen as a marketing, finance, or supply chain service.
- Finally, there is a larger than ever presence of regulations and standards. The introduction of regulations like the Sarbanes–Oxley Act or the Basel accords, combined with the various industry standards, create external pressures on organizations to standardize some IT practices in order to comply with these constraints.

The paper first looks at the definition of governance, and then proceeds to explore how those five trends are challenging traditional views of governance, or the accepted practice of governance in organizations.

2 What Is Governance?

Governance is formally defined as the "*processes, structures, and relational mechanisms in the organization that enable both business and IT people to execute their responsibilities in support of business/IT alignment and the creation of business value from IT enabled business investments*" (Wilkin and Chenhall 2010, p. 3; Van Grembergen and De Haes 2009). The governance of IT activities has been a research topic in Information Systems for a very long time. While it may have been done under a label that is different than governance, numerous studies were trying to understand how we could ensure that IT resources were used to achieve business objectives. The following paragraphs give examples of such studies.

A first group of studies has examined the allocation of responsibility for IT activities. The responsibility of activities was investigated through the lens of centralization and decentralization, assessing if activities are better managed by a central unit or by sub-units of the organization (Ahituv et al. 1989; Ein-Dor and Segev 1978). Centralization decisions were also analyzed in data processing activities that were done manually (Campbell-Kelly 1998). Responsibility of activities was also investigated by looking at what should be done by IT departments and what could be done by end-users (Rivard and Huff 1988; Rockart and Flannery 1983). The structure of the IS department was also examined (Ein-Dor and Segev 1982) to determine under which structure activities should be managed. This line of inquiry extended beyond the firm's boundary. It sought to understand not only how activities could be organized within an organization, but also what should be kept inside the firm, and what could be given to external parties (Loh and Venkatraman 1992; Aubert et al. 1996; Lacity and Hirschheim 1993). It extended our understanding of the outsourcing phenomenon (Lacity et al. 2009; Aubert et al. 2012).

A second group of studies examined the processes that could be put in place to manage IT activities. These studies examined, for instance, the effectiveness and the impact of chargeback schemes for IT resources (Drury 1980; Hufnagel and Birnberg 1989). They investigated the use of steering committees formed with members from IT and top management to manage IT operations (Drury 1984) or to select IT projects (McKeen and Guimaraes 1985). Some of that literature is also linked to standards and framework, like COBIT for example (Joshi et al. 2017).

Finally, researchers investigated relational characteristics that could be added to structures and processes to improve the management of IT. These characteristics are associated with the individuals participating in processes, rather than being a characteristic of the processes themselves (Petersen 2004). This can be illustrated by the examination of the extent to which IS and business executives understood each other (Reich and Benbasat 1996), or by the capacity of CIOs to discuss with business executives (Li et al. 2006).

This broad tradition of information systems research pertains to the three main components of governance: structure, processes, and relational capability (Van Grembergen and De Haes 2009; Wilkin and Chenhall 2010). Current views of governance tie together all these elements.

Governance includes the allocation of decision rights: which part of the organization should make IT decisions. This allocation leads to a variety of structures which have been described in well-established frameworks for IT governance (Weill 2004; Weill and Ross 2005). The structural elements were regularly researched in studies on governance. They include the mechanisms completing the structure like formal roles, positions, committees, and councils (Brown and Grant 2005).

Structures need to be supplemented with committees, collaboration mechanisms, etc. in order to enable a working governance. These process elements can be seen as tools used to implement governance. For example, having a steering committee is a structural element, but having a performance monitoring process relying on balanced scorecards is a governance process that can be used by the steering committee (De Haes et al. 2013). Having a chargeback system for IT activities would be another example of such governance processes (De Haes and Van Grembergen 2009). The COBIT 5 framework, proposed by ISACA, describes these processes as "enablers" of good governance (De Haes et al. 2013). There are 37 processes in COBIT 5 (Joshi et al. 2017). These processes will support decision making and monitoring activities.

Finally, the relational capabilities are means used by the stakeholders involved in the governance processes (Petersen 2004). These would include for example participation by stakeholders, active conflict resolution, job rotations, colocation, informal meetings, leading by example activities, and many others (De Haes and Van Grembergen 2009).

Good governance does not mean the same thing as good decisions. The governance represents the initial set of conditions. The decisions are the outcomes. Good governance means that the right people have the means to make the right decisions, but they can still make mistakes.

Governance choices have been seen as influencing strategic alignment, resource management, risk management, and the delivery of value (Wilkin and Chenhall 2010).

In recent years, the changing nature of IT and IT management suggests that we need to adopt a broader view of IT governance. The control of assets is softening because outsourcing has blurred traditional organizational boundaries. Offshoring has dissolved many geographical boundaries, weakening traditional legal and regulatory protections. The control of intellectual property is also challenged by the growth of platforms, crowdsourcing, and other forms of sharing. The evolution of technology and its pervasiveness in all activities has made the line of responsibility for IT decisions increasingly difficult to define as decisions made by business departments have more and more an embedded IT component. Finally, the increasing role of standards creates a pressure for conformity, which reduces the decision margin for organizations. These elements are explored in the subsequent pages, along with the impact these changes have on the idea of governance.

3 Erosion of Control: Five Trends

3.1 Outsourcing

Outsourcing is, it itself, a governance decision. The firm decides if an activity is managed in-house or if it is managed by a contractor (Aubert et al. 1996). It is a decision about the allocation of responsibility. However, outsourcing has at times been presented as an independent construct from governance (Ali and Green 2012). This suggests that "governance" was implicitly referring to "internal governance" in many instances. Considering the frequent use of external services for IT in organizations, it seems logical to include in a governance umbrella the activities managed internally, and activities provided by a supplier. From a governance point of view, once a contract is signed, the responsibility for the activity and the management mechanisms are defined for the duration of the contract. This implies that governance adjustments are less likely for that period. However, it is still a governance choice.

In various IT governance approaches, once the outsourcing decision has been made, it is more or less treated as a black box. Weill and Ross (2005) included outsourcing in the infrastructure decisions. However, their archetypes did not include a space for the role of the vendor, even if some decisions described would include choices made by the vendor. Yet when looking at outsourcing arrangements, the organization of the arrangement clearly describes the governance structure of the arrangement (for example Ross and Beath 2006).

Outsourcing has taken a variety of forms, and the rapid advancement of cloud services has enabled companies to customize their portfolio of activities as never before. The embeddedness of outsourcing in the organizations' activities suggest that it should be explicitly included in an IT governance framework. This is also supported by the results coming from a Delphi study investigating the topics practitioners thought the most important to understand formally (Gewald and Schäfer 2017). These included several elements suggesting that outsourcing should be explicitly considered in organization governance frameworks, for instance:

- integrating outsourcing into enterprise risk management,
- implementation of legal and regulatory requirements,
- governing multi-sourcing arrangements,
- developing a multi-vendor outsourcing landscape,
- developing outsourcing networks, etc.

All these illustrate how outsourcing arrangements and other arrangements, inside and outside the firm, are interdependent. The structure of the outsourcing arrangements, and the associated implications for the governance decisions, should be made explicit in a governance framework.

3.2 Offshoring

Offshoring, whether it is done through captive centres or through offshore providers, also has implications on governance. Location of activities is not solely a cost decision. Moving activities, systems, or data to a different country has implications for privacy, ownership, security, enforcement of regulations, etc. (Blackburn 2015; Nassimbeni et al. 2012). Such approaches require the development of new tools to manage distributed data for instance (Charlesworth and Pearson 2013). It also means that companies will take into account legal elements (like property rights protection) when deciding in which locations they will offshore a service (Jandhyala 2013). Thus, offshoring has a direct impact on the governance choices.

Offshoring is also analogous with a centralization—decentralization decision. Such decisions remain important when considering the impact of linkages between different activities. Decision makers have to consider the modularity and the maturity levels of the activities (Pisano and Shih 2012). Moving IT activities away could have a mid-term detrimental effect on other activities. For example, the offshoring of manufacturing activities is seen as having reduced the ability of several firms to perform hi-level R&D activities (Denning 2013). Conversely, some authors argue that offshoring is actually a way to access rare talent everywhere on the planet, thus improving R&D activities (Lewin et al. 2009). Detailed examination of innovation performance showed that the impact of offshoring on performance was not linear (Steinberg et al. 2017). Both opposing views are probably valid. We would need to investigate under which conditions one is more salient than the other.

This suggests that, in any governance framework, localization of activities should be addressed explicitly. Traditional archetypes (Weill and Ross 2005 for example) were considering the level of decentralization but were not looking specifically at the geographical position. We could imagine interesting combinations. For instance, an "IT Monarchy" in the Weill and Ross (2005) language, could be located in a wholly-owned offshore subsidiary. We would still have centralized IT decision making, but remote from the head-office of the organization. It would create a very different risk profile for the IT governance than a similar centralization in the same country as the head office.

Evidence suggests that companies are finding advantages with offshore delivery of IT and business services and are not much influenced by pressure from their environment when considering offshoring (Khan and Lacity 2012). This suggests that offshoring may continue to increase. Therefore, a governance approach should explicitly consider the geographical component of the structure.

3.3 Platforms, Crowds, and Open Source

Governance choices are also increasingly influenced by the growing importance of collective decisions made outside the firms. As companies are joining platforms, relying on crowd sourcing, and integrating open source software into their portfolio, they are integrating into their portfolio elements over which they have limited influence or control.

Platforms can be defined as "*a set of digital resources—including services and content—that enable value-creating interactions between external producers and consumers*" (Constantinides et al. 2018, p. 381). They provide an interface for organizations to have access to resources, crowds, suppliers, and customers. Using platforms as a strategy was linked with modularity of product development. Several examples have shown that it was possible to have a platform where external participants would be providing modules and innovating in the production of these modules. As long as the interface between the module and the platform remained the same, it enabled a decentralized form of innovation (Langlois 2002). Examples of such innovation have been observed in the automotive industry (Muffatto 1999) or computer industry (Langlois and Robertson 1992). Such platforms can display strong network effect and have strategic competitive importance (Gawer and Cusumano 2014). It can be difficult for firms to operate outside established ecosystems enabled by platforms.

This would suggest that joining an IT platform can be seen as ceding a certain degree of freedom about the governance of IT. It involves an externalization of a business process (Segev and Gebauer 2001). When joining a platform, an organization agrees to adhere to the interface defined for the platform (Tiwana et al. 2010). The organization may have limited or even no control on the platform and its evolution. Joining a platform is very similar to outsourcing the functions that are provided by the platform.

Traditionally, the choices of IT governance were implicitly seen as under the firm's boundary of control. However, with the advent of platforms discussed in the previous paragraphs, more and more decisions linked to IT and data governance are collective decisions. Companies are part of alliances and groups who collaborate through platforms. The ownership structure of these platforms vary greatly, going from cooperative-types to ones owned by a few entrepreneurs and venture funds (Kenney and Zysman 2016). This means that governance decisions for the IT components within the platforms are outside the scope of control of the firm using it. Crowd or open source approaches, which are increasingly part of the software portfolio of many organizations (Thakker et al. 2017), follow the same logic. The evolution of the software used by the firm is controlled by an outside entity.

Governance in a shared environment (platform, crowd, or open source) is different from governance in an environment controlled by the organization. In situations where the firm has a voice in the evolution of the platform, decisions must be made in collaboration with numerous partners who have very different, potentially conflicting, interests. In cases where the platform is controlled by outside investors, there is no control possible. It means that the ideas, the intellectual property, the evolution path of technology, and the investments associated with those platforms are the result of some collective or external decision process, over which a firm has a very limited or no influence. If the portion of the firm's activities transferred to various platforms increases, it would suggest that the portion of the firm's activities remaining under the control of that firm decreases. This suggests that governance should have an explicit consideration for the "distribution of control and participation" observed in all forms of shared environment.

3.4 IT Evolution and the Boundary of IT Departments

In 2011, IT World Canada was reporting that users and core business departments were increasingly taking control over their IT budgets and activities, thus threatening IT departments. This would suggest that some control over IT governance has been transferred to user groups and core business departments. Several factors may contribute to the acceleration of this process.

We are observing an increasing embeddedness of IT into business activities (fintech, social media marketing, etc.). When looking at the fintech sector for instance, these initiatives are bringing together start-ups, IT firms, customers, regulators, and traditional financial institutions (Lee and Shin 2018). While fintech start-ups were initially seen to threaten traditional financial companies (banks or insurance companies), they are now used by these traditional companies to enhance their operations and offerings (Puschmann 2017). Such collaboration between traditional providers and new entrants enables each group to gain: banks benefit from accelerated innovation while fintechs get access to the banks' market (Bömer and Maxin 2018).

Interestingly, when such agreements are negotiated, they are often not under the responsibility of the IT department like an outsourcing contract would be. These agreements are often seen as strategic and can bring to the table senior executives of the bank (see example described in Hung and Lo 2016). They can be transformative since they alter the relationship of the bank with its clients (Dapp et al. 2015). They can also be seen as simple business service provision when they are provided through SaaS agreements. They then become invisible for IT managers (Gozman and Willcocks 2015).

The proliferation of mobile apps, which is part of the consumerization of IT, also comes into play when considering the boundary of IT departments. IT consumerization is a challenge for IT governance (Gregory et al. 2018). Workers are using various mobile platforms in their personal lives, and they bring those platforms in the work environment to enjoy the benefits associated with them. This behavior typically creates conflicts with the values held by corporate IT departments (Koch et al. 2014). In this case, the transfer of activities is not done by the firm to a specific platform (as discussed in the preceding section) but by the workers in the firm to a variety of platforms of their choice, using their own personal devices. This brings several security challenges (Garba et al. 2015). Evidence suggests that this trend is growing and that governance frameworks will require adjustments or transformation (Gregory et al. 2018). When employees use personal devices with a variety of mobile software providers, there is no control on the evolution of these apps, and limited control over the data that is shared on those platforms.

This means that an increasingly large portion of the decisions traditionally made under the "IT department" umbrella, concerning technology, data, systems, applications, and related elements now falls under other departments' authority, or under the "discretionary authority" of individual users. These usages of information technology are often not visible centrally and are not formally included in an IT governance framework. A traditional framework thus includes a smaller and smaller portion of "IT activities" of the firm every year. This suggests that IT governance should have an explicit consideration for the "IT/Data" elements that do not fall within the traditional IT department span of control (or budget). There is a need to control the "shadow IT" as labelled by Gozman and Willcocks (2015).

This idea is not unique. It may be similar in some way to finance departments which have to report on all financial transactions in an organization, no matter in which department the transaction was made. Therefore, we could think of an IT governance looking at "IT activities" with a definition that is not dependent on the link of the activities with the IT department.

3.5 Conformity Pressures (SOX, COBIT, Basel III, etc.)

There is an increasingly large number of standards, conformity requirements, and sets of "best practices" influencing firms' choices of processes and technologies. Some of these are mandatory, for example the requirements introduced by regulations like the Sarbanes–Oxley Act (United States 2002). Other standards like ISO ones are not mandatory but chosen by companies to improve their performance, send positive signals to markets, or respond to institutional pressure (Bansal and Bogner 2002). Even when standards are optional, there is pressure on organizations to adopt them in order to show good practices.

The influence of standards and regulations can simply stem from the need to demonstrate compliance. If a standard is adopted, it becomes required to show to external parties that it is followed. This is one of the reasons why regulations of various forms are seen as increasing control in organizations and generating higher levels of bureaucratization for IT management (Cleven and Winter 2009).

The introduction of standards and regulations is likely to increase standardization of governance and processes (in addition of the standardization intended). Frameworks like COBIT provide a view of the IT domain and of the various elements that have to be controlled (De Haes et al. 2013). While the implementation of a framework leaves room for a variety of control processes, it is providing a common view of the IT function and its governance. Such standardization of the view adopted by different units can lead to more commonalities in the decisions made by these units when they manage IT. In addition, the introduction of regulations makes it more attractive for organizations to standardize business processes. Organizations take advantage of triggers like Sarbanes Oxley Act to revise processes to ensure they are performed in the same manner across all units. It reduces errors, increases efficiency, and facilitates reporting (Stephen and Ditmar 2006).

Regulations can also lead to centralization of the IT decisions. Some regulations, like Basel for instance, will put constraints on reporting cycles, disclosure of data, and conservation of data for defined periods of time (Luthy and Forcht 2006). Implementing the processes to ensure regulations are followed may be more efficient in a centralized environment than in a decentralized one, especially when it comes to audits. Centralization would facilitate demonstrating companies are meeting these requirements.

This suggests that there could be conformity pressures on governance choices coming from the numerous regulations and standards introduced in the last two decades. This influence is not limited to regulations pertaining specifically to IT. The need to report on those regulations may also contribute to centralization of some IT decisions.

There is no judgement made on the possibility that there may be a form of standardization of governance choices. It is impossible to know if this increased conformity is desirable or not. On one hand, if best practices are adopted by more institutions, and if adding governance features increases performance, then conformity pressure would lead to improved outcomes. On the other hand, if governance is a configuration that must be well suited to specific strategic goals (Weill and Ross 2005), then standardization may be detrimental because it would prevent adaptability in governance choices.

4 Rethinking IT Governance

The previous elements discussed suggest that the notion of IT governance needs to be reconceptualized so as to be more congruent with the more complex environment of IT. Going back to the definition, structure, process, and relational mechanisms may take a new variety of shapes and forms, and the relative importance of each may change.

4.1 Structure

Defining a governance structure was traditionally seen as deciding on the location of control over activities. It seems that choices of location are more numerous than before, and that the control itself is more blurry than before. In addition to the traditional internal choices associated with various levels of centralization or decentralization, organizations can include in the mix external partners (vendors, platforms, regulatory bodies).

This would suggest that contracts with external parties, vendors, platforms, etc. would need to include a governance component. Service level agreements typically include elements associated with the performance of the activities and the relational governance—the relationship between the two parties—Goo (2010), Goo et al. (2009). In order to address long term corporate IT governance such contracts should probably include a component much broader, enabling an overall view of the IT activities outsourced to ensure their visibility in the IT governance framework chosen by the firm. This could be done within the realm of increased contract completeness (Aubert et al. 2017).

However, when joining a platform, adopting a standard, or when letting users adopt technologies of their choice, it is much less clear that organizations assess (or even recognise) the implications for long-term control of their architecture or software principles. They may well be unsuspectedly entering into a web of control where all parties have only partial control over components of the technologies in place.

It also seems that in addition to a decision on the location of control, there is also a decision on the level of control. It looks like there could be many levels of “partial control” adopted by the firms. When organizations decide to cede control over some decisions, they can still influence those decisions. For instance, an organization could decide to use a platform for an activity, therefore agreeing to cede decision rights to the platform for the conduct of the activities. In parallel, the organization can take an active part in the platform consortium, trying to influence its evolution in ways that would be favourable for the company.

Discussing structure also raises questions about the role of the IT department in the IT governance process. To what extent should the IT department oversee the services (involving data transfers) bought through IT channels by other departments of the organization? Since many services which used to be provided internally (with the support of IT departments) are now offered by suppliers transacting directly with functional departments of the firm, the footprint of IT governance is shrinking. One could argue that hardware and software are simply the tools (and the responsibility) of the provider. So as long as the service is provided, governance should not be the concern of the client firm. However, this reasoning is difficult to apply to data. The use of external services (SaaS) means that company data is moving out of the company toward external servers. Protection, secondary usage (mining), re-use, or data structure are all elements over which the client firm used to have control and has lost it. Does it become the responsibility of the department (marketing, finance, logistics, or another) that is purchasing the service to ensure that these concerns are addressed? Does it remain the responsibility of IT? If so, how can the IT department take responsibility for an activity happening without its involvement?

If we make the analogy with finance, organizations have control mechanisms ensuring that financial standards and rules set by the department of finance are followed no matter which component of the organization is initiating the transaction. There are also thresholds above which transactions need the scrutiny of the finance department before being approved. Shall we develop a similar logic for data? In that case, the structure of IT governance may not matter as much as the processes of IT governance. That could be a way to ensure that IT governance includes data formally, and that this resource is managed with appropriate care.

4.2 Process

At the process level, a revised governance framework would need to consider all processes and activities established beyond the organization. Traditionally we included in the implementation tools of governance several committees to facilitate coordination between the business domain and IT. However, Ali and Green (2012) could not find a link between IT strategy and IT governance committees and the perceived effectiveness of IT governance in organizations. This could suggest that if we need to expand the processes around governance to include vendors, standardization organizations, and other outside stakeholders, multiplying committees may not be the ideal strategy.

Thinking about the goals of these processes, they would probably vary from one situation to another: reporting, controlling, influencing, staying informed, etc. Financial processes could offer paths to follow in order to build IT governance processes. Accountants look at governance to enable the best use of financial resources: "*A fundamental objective of governance research in accounting is to provide evidence* *on the extent to which information provided by financial accounting systems mitigate agency problems ... facilitating the efficient flow of scarce human and financial capital to promising investment opportunities*" (Bushman and Smith 2001, p. 238). This is similar to the goal of IT governance with respect to IT resources. Ironically maybe, accounting seems to have put more importance on reports (data collection, data organization, and data analysis) than IT when defining governance processes. If such reports were developed for IT resources, they may include a standardized way to disclose information about IT resources, for example:

- Data: Which data is owned, which data is used, which data is in the company, in which country is the data stored, etc.
- Ownership and responsibility of data and systems: for financial data, it is clear in a company who has authority to sign for a given budget item. A similar clarity should be defined for data and systems. When an employee starts a job, the employee is informed about the authority over IT resources associated with the role (no matter in which department is the role).
- Contract structure: financial information will have details about supply chain flows of resources and the risk they may pose to the organization. IT contract structure, along with the risk they may create for the organization, should be visible too.
- Processes to audit IT reports (internally or externally) should be formally established, as well as associated reporting mechanisms.

These are only a few suggestions offered as examples. These processes would provide a way to track information assets and their location, knowing who is managing them, and under which regulatory regime they are managed.

Ensuring visibility of IT resources would also highlight which resources are under shared control (or no control) because they are governed by other parties. The organization would then decide on the activities it can undertake to influence the evolution of these assets. For example, there are panels to influence software evolution[2] or standardization boards. This may mean partnering formally with associates (or even competitors) to influence third parties more effectively.

4.3 Relational Capabilities

A more encompassing view of IT governance will also widen the understanding of relational elements. There are probably new types of skills and behaviours that will be required in order to support the new processes in this new governance approach. Just like IT managers had to learn contract management when they started to outsource IT, IT managers will have to learn new skills to manage this IT governance where so much of the control has been "outsourced" to a variety of entities.

IT managers have a long tradition of active management of software projects (Barki et al. 2001). It is possible that the abilities developed for the management

---

[2]For example: https://www.sap.com/canada/about/customer-involvement/user-groups.html.

of projects could be reused to actively manage IT resources outside the realm of projects. While some of the activities may be different, there would still be a need to understand usage and processes, linking with user departments, or convincing and managing change, that would be analogous. This could mean that the responsibilities of overseeing data resources need to be clarified for the entire cycle of the systems, from development of the project to the decommissioning phase at the end. It does not mean that the IT department must do it all, but it probably has to provide the tools to do it, as well as ensuring that the information is collected and reported appropriately.

This would probably suggest that there is an increased need for boundary spanners. These roles, often studied in the context of knowledge management, could be essential elements in governance if governance of IT resources flows across business units. It also suggests that governance is something that requires active management, rather than periodical assessment or revision.

4.4 Final Comments

Observation of recent trends indicate that some components of traditional IT governance may have been "outsourced" outside the realm of the IT department, consciously or not. Considering the increased dispersion of IT resources, and their embeddedness in all the activities of the firm, the analysis suggests that the future of governance may lie more in appropriate processes rather than on well-defined structures.

The development of these processes may not happen without external pressure. Several governance processes in the corporate and financial world were pushed by legislation, for example when the Sarbanes Oxley Act was introduced following several major scandals, with Enron being probably the most visible one. For IT resources, there does not seem to be a similar impetus at the moment. Pressure related to IT and data seem to be mostly associated with personal data and social media influence.[3]

In parallel, waiting for a scandal or a major crisis may be risking severe damage for organizations. In the short term, it could be interesting to research the use of some of those processes in forward-thinking organizations to assess if they are seeing benefits in such an approach to IT governance. Mapping the changes in data and IT resources usage to start establishing an inventory of the land would also be a valuable step forward. It would give an idea of the scope of the work ahead.

[3]See for example: Hughes, C. (2019).

References

Ahituv, N., Neumann, S., & Zviran, M. (1989). Factors affecting the policy for distributing computing resources. *MIS Quarterly*, 389–401.

Ali, S., & Green, P. (2012). Effective information technology (IT) governance mechanisms: An IT outsourcing perspective. *Information Systems Frontiers, 14*(2), 179–193.

Aubert, B. A., Rivard, S., & Patry, M. (1996). A transaction cost approach to outsourcing behavior: Some empirical evidence. *Information & Management, 30*(2), 51–64.

Aubert, B. A., Houde, J. F., Patry, M., & Rivard, S. (2012). A multi-level investigation of information technology outsourcing. *The Journal of Strategic Information Systems, 21*(3), 233–244.

Aubert, B. A., Houde, J. F., Rivard, S., & Patry, M. (2017). Determinants of contract completeness for information technology outsourcing. *Information Technology and Management, 18*(4), 277–292.

Bansal, P., & Bogner, W. C. (2002). Deciding on ISO 14001: economics, institutions, and context. *Long Range Planning, 35*(3), 269–290.

Barki, H., Rivard, S., & Talbot, J. (2001). An integrative contingency model of software project risk management. *Journal of management information systems, 17*(4), 37–69.

Blackburn, C. (2015). New technology, personal data protection and implications' for financial services regulation'. *JASSA, 4,* 59.

Bömer, M., & Maxin, H. (2018). Why fintechs cooperate with banks—evidence from germany. *Zeitschrift für die gesamte Versicherungswissenschaft, 107*(4), 359–386.

Brown, A. E., & Grant, G. G. (2005). Framing the frameworks: A review of IT governance research. *Communications of the Association for Information Systems, 15*(1), 38.

Bushman, R. M., & Smith, A. J. (2001). Financial accounting information and corporate governance. *Journal of Accounting and Economics, 32*(1–3), 237–333.

Campbell-Kelly, M. (1998). Data processing and technological change: The post office savings bank, 1861–1930. *Technology and Culture, 39*(1), 1–32.

Constantinides, P., Henfridsson, O., & Parker, G. G. (2018). Introduction—platforms and infrastructures in the digital age. *Information Systems Research, 20*(2), 381–400.

Charlesworth, A., & Pearson, S. (2013). Developing accountability-based solutions for data privacy in the cloud. *Innovation: The European Journal of Social Science Research*, *26*(1–2), 7–35.

Cleven, A., & Winter, R. (2009). Regulatory compliance in information systems research–literature analysis and research agenda. In *Enterprise, business-process and information systems modeling* (pp. 174–186). Berlin, Heidelberg: Springer.

Dapp, T., Slomka, L., AG, D. B., & Hoffmann, R. (2015). Fintech reloaded–Traditional banks as digital ecosystems. *Deutsche Bank Research*, 261–274.

De Haes, S., & Van Grembergen, W. (2009). An exploratory study into IT governance implementations and its impact on business/IT alignment. *Information Systems Management, 26*(2), 123–137.

De Haes, S., Van Grembergen, W., & Debreceny, R. S. (2013). COBIT 5 and enterprise governance of information technology: Building blocks and research opportunities. *Journal of Information Systems, 27*(1), 307–324.

Denning, S. (2013). Boeing's offshoring woes: seven lessons every CEO must learn. *Strategy & Leadership, 41*(3), 29–35.

Drury, D. H. (1980). A survey of data processing chargeback practices. *INFOR: Information Systems and Operational Research*, *18*(4), 342–353.

Drury, D. H. (1984). An evaluation of data processing steering committees. *MIS Quarterly*, 257–265.

Ein-Dor, P., & Segev, E. (1978). Centralization, decentralization and management information systems. *Information & Management, 1*(3), 169–172.

Ein-Dor, P., & Segev, E. (1982). Organizational context and MIS structure: Some empirical evidence. *MIS quarterly*, 55–68.

Garba, A. B., Armarego, J., Murray, D., & Kenworthy, W. (2015). Review of the information security and privacy challenges in Bring Your Own Device (BYOD) environments. *Journal of Information privacy and security, 11*(1), 38–54.

Gawer, A., & Cusumano, M. A. (2014). Industry platforms and ecosystem innovation. *Journal of Product Innovation Management, 31*(3), 417–433.

Gewald, H., & Schäfer, L. (2017). Quo vadis outsourcing? A view from practice. *Journal of Global Operations and Strategic Sourcing, 10*(1), 2–17.

Goo, J. (2010). Structure of service level agreements (SLA) in IT outsourcing: The construct and its measurement. *Information Systems Frontiers, 12*(2), 185–205.

Goo, J., Kishore, R., Rao, H. R., & Nam, K. (2009). The role of service level agreements in relational management of information technology outsourcing: An empirical study. *MIS Quarterly*, 119–145.

Gozman, D., & Willcocks, L. (2015). Crocodiles in the regulatory swamp: navigating the dangers of outsourcing, SaaS and shadow IT. In *International Conference on Information Systems*, Fort Worth, USA (20 pp.).

Gregory, R. W., Kaganer, E., Henfridsson, O., & Ruch, T. J. (2018). IT consumerization and the transformation of IT governance. *MIS Quarterly, 42*(4), 1225–1253.

Henderson, J. C., & Venkatraman, H. (1999). Strategic alignment: Leveraging information technology for transforming organizations. *IBM Systems Journal*, *38*(2.3), 472–484.

Hufnagel, E. M., & Birnberg, J. G. (1989). Perceived chargeback system fairness in decentralized organizations: An examination of the issues. *MIS Quarterly*, 415–430.

Hughes, C. (2019). Is it time to break up Facebook, *New York Times*, May 9th. Accessed May 11, 2019. https://www.nytimes.com/2019/05/09/opinion/sunday/chris-hughes-facebook-zuckerberg.html?.

Hung, J. L., & Luo, B. (2016). FinTech in Taiwan: a case study of a Bank's strategic planning for an investment in a FinTech company. *Financial Innovation, 2*(1), 15.

Jandhyala, S. (2013). Property rights and international investment in information technology services. *Strategic Management Journal, 34*(7), 877–889.

Joshi, A., Huygh, T., & De Haes, S. (2017). Examining the association between industry IT strategic role and IT governance implementation.

Kenney, M., & Zysman, J. (2016). The rise of the platform economy. *Issues in Science and Technology, 32*(3), 61.

Khan, S., & Lacity, M. (2012). Survey results: are client organizations responding to anti-offshoring pressures? *Strategic Outsourcing: An International Journal, 5*(2), 166–179.

Koch, H., Zhang, S., Giddens, L., Milic, N., Yan, K., & Curry, P. (2014). Consumerization and IT department conflict. In *International Conference on Information Systems*, Auckland, New Zealand (15 pp.).

Lacity, M. C., & Hirschheim, R. A. (1993). *Information systems outsourcing; myths, metaphors, and realities*. New York: Wiley.

Lacity, M. C., Khan, S. A., & Willcocks, L. P. (2009). A review of the IT outsourcing literature: Insights for practice. *The Journal of Strategic Information Systems, 18*(3), 130–146.

Langlois, R. N. (2002). Modularity in technology and organization. *Journal of Economic Behavior & Organization, 49*(1), 19–37.

Langlois, R. N., & Robertson, P. L. (1992). Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries. *Research Policy, 21*(4), 297–313.

Lee, I., & Shin, Y. J. (2018). Fintech: Ecosystem, business models, investment decisions, and challenges. *Business Horizons, 61*(1), 35–46.

Lewin, A. Y., Massini, S., & Peeters, C. (2009). Why are companies offshoring innovation? The emerging global race for talent. *Journal of International Business Studies, 40*(6), 901–925.

Li, Y., Tan, C. H., Teo, H. H., & Tan, B. C. (2006). Innovative usage of information technology in Singapore organizations: Do CIO characteristics make a difference? *IEEE Transactions on Engineering Management, 53*(2), 177–190.

Loh, L., & Venkatraman, N. (1992). Determinants of information technology outsourcing: A cross-sectional analysis. *Journal of Management Information Systems, 9*(1), 7–24.

Luthy, D., & Forcht, K. (2006). Laws and regulations affecting information management and frameworks for assessing compliance. *Information Management & Computer Security, 14*(2), 155–166.

Markus, M. L. (2017). Datification, organizational strategy, and is research: What's the score?. *The Journal of Strategic Information Systems, 26*(3), 233–241.

McKeen, J. D., & Guimaraes, T. (1985). Selecting MIS projects by steering committee. *Communications of the ACM, 28*(12), 1344–1352.

Muffatto, M. (1999). Introducing a platform strategy in product development. *International Journal of Production Economics, 60,* 145–153.

Nassimbeni, G., Sartor, M., & Dus, D. (2012). Security risks in service offshoring and outsourcing. *Industrial Management & Data Systems, 112*(3), 405–440.

Peterson, R. (2004). Crafting information technology governance. *Information Systems Management, 21*(4), 7–22.

Pisano, G. P., & Shih, W. C. (2012). Does America really need manufacturing. *Harvard Business Review, 90*(3), 94–102.

Puschmann, T. (2017). Fintech. *Business & Information Systems Engineering, 59*(1), 69–76.

Reich, B. H., & Benbasat, I. (1996). Measuring the linkage between business and information technology objectives. *MIS Quarterly*, 55–81.

Rivard, S., & Huff, S. L. (1988). Factors of success for end-user computing. *Communications of the ACM, 31*(5), 552–561.

Rockart, J. F., & Flannery, L. S. (1983). The management of end user computing. *Communications of the ACM, 26*(10), 776–784.

Ross, J. W., & Beath, C. M. (2006). Sustainable IT outsourcing success: Let enterprise architecture be your guide. *MIS Quarterly Executive*, *5*(4).

Segev, A., & Gebauer, J. (2001). B2B procurement and marketplace transformation. *Information Technology and Management, 2*(3), 241–260.

Steinberg, P. J., Procher, V. D., & Urbig, D. (2017). Too much or too little of R&D offshoring: The impact of captive offshoring and contract offshoring on innovation performance. *Research Policy, 46*(10), 1810–1823.

Stephen, W., & Ditmar, L. (2006). The unexpected benefits of Sarbanes-Oxley. *Harvard Business Review,* April, 133–140.

Straka, J. W. (2000). A shift in the mortgage landscape: The 1990s move to automated credit evaluations. *Journal of Housing Research, 11*(2), 207–232.

Thakker, D., Schireson, M., Nguyen-Huu, D., (2017, April 7). Tracking the explosive growth of open-source software, Techcrunch.com. Retrieved April 15, 2019 from https://techcrunch.com/2017/04/07/tracking-the-explosive-growth-of-open-source-software/.

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Research commentary—Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research, 21*(4), 675–687.

United States. (2002). *Sarbanes–Oxley Act of 2002: Conference Report (To Accompany H.R. 3763)*. Washington, D.C.: U.S. G.P.O..

Van Grembergen, W., & De Haes, S. (2009). *Enterprise governance of information technology: achieving strategic alignment and value*. Springer Science & Business Media.

Weill, P. (2004). Don't just lead, govern: How top-performing firms govern IT. *MIS Quarterly executive, 3*(1), 1–17.

Weill, P., & Ross, J. (2005). A matrixed approach to designing IT governance. *MIT Sloan Management Review, 46*(2), 26–35.

Wilkin, C. L., & Chenhall, R. H. (2010). A review of IT governance: A taxonomy to inform accounting information systems. *Journal of Information Systems, 24*(2), 107–146.

Data Sourcing and Data Partnerships: Opportunities for IS Sourcing Research

Sirkka L. Jarvenpaa and M. Lynne Markus

Abstract A void exists in information systems (IS) sourcing research: Organizations increasingly source data for varied purposes, but IS sourcing literature has not focused on data sourcing nor sourcing partnerships. We examine some of the implicit views in the IS literature regarding data that have not yet been well articulated. Exploring these views in terms of data sourcing arrangements offers future research opportunities.

1 Information Systems (IS) Sourcing Research Needs a Focus on Data Sourcing

A focus on data in information systems (IS) sourcing research has slipped into a void. This void contrasts with the rich literature on sourcing of information technology services and business process services—what Lacity et al. (2016: 270) calls "business services." In their comprehensive model involving the determinants of sourcing outcomes, Lacity et al. (2016) highlight client and provider firm characteristics, their capabilities, and partnership decisions in terms of motivation, type, duration, location, and governance, among others. The model also identifies transaction attributes, such as types, costs, formalization (codification), standardization, risk, and uncertainty. However, transaction types characterized as data were not explicitly considered. They were only indirectly discussed—for example, in terms of the providers' capabilities to protect client data. This void is not only in IS sourcing, but also in related research domains, such as platforms and infrastructures.

This void is unfortunate given the strategic importance of sourcing for data in businesses, academia, and the public sector (Shantz 2018). This importance is conveyed in statements like "algorithms without data are just a mathematical fiction" (Constantiou and Kallinikos 2015). Although organizations traditionally have insourced

S. L. Jarvenpaa (✉)
McCombs School of Business, University of Texas at Austin, Austin, USA
e-mail: sirkka.jarvenpaa@mccombs.utexas.edu

M. L. Markus
Bentley University, Waltham, USA
e-mail: mlmarkus@bentley.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_4

their data—that is, they themselves generated the data that they then integrated for their intended uses (Brohman et al. 2003)—data now increasingly are sourced externally and reused for purposes that were unimaginable at the time of the original data generation.

In the recent curation that reviewed studies on sourcing published in MIS Quarterly since 1990s, Kotlarski et al. (2018) define IS sourcing as "the contracting or delegating of IS- or IT-related work (e.g., an ongoing service or one-off project) to an internal or external entity (a supplier)." The curation highlights three foundational clusters in IS sourcing literature: (1) making the sourcing decision, (2) designing contractual structures, and (3) managing the sourcing relationship. The curation identified only one paper that explicitly dealt with data. Chen et al. (2017) examined intellectual property rights (IPR) sharing in a software development project. Databases were one type of IPR they examined, including their data structures, data contents, and reports. The paper found that joint ownership of data by the client and vendor was contingent on the vendor's bargaining power and the relationship-specific investments required. However, the paper provided no discussion of how data or how perspectives on data shaped the power–dependence relationship between the client and vendor.

Konning et al. (2019) reviewed recent developments in IT business services outsourcing—both infrastructural and knowledge-intensive sourcing. Two developments were particularly highlighted. First, IS sourcing literature has increasingly examined innovation effects from sourcing. Konning et al. (2019) synthesize that the relationship characteristics of communication, knowledge sharing, and trust increase the innovation intensity in sourcing. Second, the sourcing decisions has become more multifaceted. Although sourcing historically was associated with cost reduction, other motivators were increasingly salient such as access to "expertise/skill, quality improvement, and focus on core capabilities." Interestingly, sourcing studies for innovation did not address the role of data in sourcing. Neither were data access nor data use among the motivators included in sourcing decisions.

Scholars in information systems increasingly examine sourcing-related questions in the context of platforms. Platforms are complex, distributed arrangements of technological and organizational systems and subsystems that support "new and flexible means for inter-organizational relations" (de Reuter et al. 2017). Platforms commonly are defined as a shared set of services and architecture, wherein the architecture includes technological modular systems and multiple actors in "multi-sided" roles (Tilson et al. 2010; Parker et al. 2016). Platforms also are prevalent in the literature on digital innovation and digital entrepreneurship, where they help to pull together resources and capabilities, including data from various external sources (Huang et al. 2017).

Ghazawneh and Henfridsson (2013) examined boundary resources in platforms, defining boundary resources as "the software tools and regulations that serve as the interface for the arm's-length relationship between the platform owner and the application developer." However, the authors do not consider data as a boundary resource; instead, data simply is enabled by the application program interfaces (APIs) or toolkits. In their review paper on platforms, Schreieck et al. (2016) state that "no article explicitly analyzes the role of data as a boundary resource in platform ecosystems."

The authors found this lacuna surprising because so many digital platforms are fueled by data sales. In a recent article on platforms and infrastructures, Constantinides et al. (2018) merely mention data and mainly focus on proprietary rights to data and value that such rights can generate. Neither data sourcing nor data partnerships were mentioned as major platform challenges. A recent platform review article by de Reuver et al. (2017) was similarly devoid of data or data partnership perspectives and suggested that data are rather unproblematic because digital technologies (providing the infrastructure for platforms) imply "homogenization" of data. An exception includes the study by Lee et al. (2017), which advances factors germane to data for platforms: (1) data ownership and access and (2) data use.

The literature on IS infrastructures has focused on hardware and software decisions, rather than on data (Lyytinen et al. 2017). Decisions on data sourcing have been in the background, or they have been assumed to be routine; partners are assumed as predetermined or fixed (Steinfield et al. 2011; de Corbiere and Rowe 2013). Research on organizational data flows examines data from the legal and societal perspectives, including privacy, ownership, and security (Markus 2016); it does not consider how data flows are affected by sourcing decisions and partnerships.

2 Studies on Data Lacks a Sourcing Focus

Research on data infrastructures has not examined the perspective of sourcing decisions, designing contractual structures, and managing sourcing relationships. The primary focus has been on how to design the relevant software systems and how to make the data available via tools and standards (Star and Ruhleder 1996). Questions concentrate on adaptability, robustness, and extensibility of infrastructures (Ribes and Polk 2014, 2015). Some health economists focus on the exchanges of health data (Miller and Tucker 2014), without zeroing in on the sourcing decision or the organizational arrangements.

Research on data governance and data quality is similarly void of the sourcing perspective. Research on data governance in IS has focused on "who holds the decision rights and is held accountable for an organization's decision-making about its data assets" (Khatri and Brown 2010). Data governance research is deduced from IT governance (Khatri and Brown 2010; Otto 2011; Tallon et al. 2013). Research also focuses on data storage and infrastructure requirements of hardware and software (Tallon et al. 2013). As a result of these perspectives, considerations such as data indeterminacy and contextual interpretation challenges have been overlooked, as have questions about how these considerations affect data sourcing and sourcing relationships.

Inter-organizational sourcing considerations are largely missing in the literature on data quality as well. According to Wang and Strong (1996), data quality involves fitness of use by users or consumers of data. Data quality is a multidimensional construct that includes four attributes: intrinsic character, context, representation, and accessibility. Data of high quality in one context might be lower quality in another context. For instance, Gainer et al. (2016) report that the codes used by clinicians to designate patient diseases in electronic medical record (EMR) systems often describe possible rather than definitive diagnoses: They tend to serve administrative and insurance billing purposes and generally are not accurate enough for research purposes. The implementation of more complex coding systems, such as ICD-10, might increase the granularity of diagnoses and potentially the accuracy and reliability of the data on the diseases of individuals. Yet granularity also brings complexity and can increase the possibility of further data errors. Moreover, different organizations deal with indeterminate or ambiguous data differently, which leads to the application of different processes to the data. Some might rely on steering groups to sift through volumes of unstructured data (e.g., doctors' and patients' notes), and others use rule-based artificial intelligence systems. Issues arise when data from these different knowledge-processing systems are merged internally, and these issues are exacerbated when data travel across organizations because the knowledge-processing systems used remain unknown or hidden to the external sourcing organizations.

In the remainder of the chapter, we proceed as follows. First, we define data sourcing and sourcing partnerships. Second, we discuss the importance of sourcing of data and sourcing partnerships in contemporary organizations. Third, we then advance three implicit—often not well-articulated—views of data in the IS literature and how these views introduce considerations often not addressed by sourcing researchers. Fourth, we examine how these views of data shape sourcing partnerships—or what we call organizational arrangements. Fifth, we conclude by highlighting opportunities for future research.

3 Data Sourcing and Data Sourcing Partnerships: Definitions

Our concern in this paper is on external sourcing of data and the related relationships and processes. Although many considerations that are important in intra-organizational contexts also prevail in inter-organizational contexts, these considerations grow in complexity and new issues arise when data travels across organizational boundaries.

Data sourcing is not data sharing. Data sharing implies direct or indirect person-to-person reciprocal exchange with voluntary intent. It also implies that meta-data—regarding origins, characteristics, and history—are shared either as tacit or explicit knowledge among the sharers. Hence, data sharing often occurs in communities of practice. By contrast, data sourcing requires that meta-data travels with the data, and this meta-data is critical to reducing contextual and interpretative challenges in use. Many industry verticals are developing standards including meta-data to promote sharing. For example, the Global Alliance for Genomics and Health (GA4GH)—the standards-setting body in genomics for healthcare—is developing global standards, but such standards are still in the making (Birney et al. 2017).

Following Kotlarski et al.'s (2018) definition of IS sourcing, we refer to data sourcing as procuring, licensing, and accessing data (e.g., an ongoing service or one-off project) from an internal or external entity (supplier). Sourcing of data includes reuse. Reuse can be for varied purposes, including purposes unforeseen and unknown at the time of the original data collection. Data sourcing constitutes inter-organizational processes and relationships. Although data sourcing can involve person-to-person interactions, such interpersonal relationships are supportive or representative of the broader interorganizational relationship. Our definition of data sourcing excludes data warehousing applications, data hosting, and data calling. This understanding parallels Kotlarski et al. (2018)'s exclusion of highly structured and routine gig work in their definition of IS sourcing.

Many IS sourcing partnerships involve dyadic customer–supplier relationships. Even in multi-sourcing contexts, contracts are negotiated at least to some extent between the client and each supplier. In contrast, data sourcing relationships are often triad "brokerage" relationships. Data brokers—data aggregators, sellers, and consolidators—source, package, and sell, rent, or reveal data from those who generate the data, which is then offered to third parties for use and reuse (Kitchin 2014). The practitioner literature has identified an internal facilitator role for data sourcers in organizations, and in this role they help to verify the quality of datasets being sourced internally and externally. This role is operational and tactical, and it primarily is concerned with data quality and governance. The role does not cover decisions of data sourcing and data sourcing partnerships.

4 Strategic Importance of Sourcing Data in Contemporary Organizations

Organizations are sourcing data for a wide variety of strategic reasons. Crucial among these reasons are the uses in business models and operations around data. Data-driven businesses are associated with innovative and fast-scaling business models. In health, energy, and finance, many algorithmic business models require data for real-time updates and responses in rapidly changing conditions (Constantiou and Kallinikos 2015). "Freemium services" rely on a quid pro quo business model of services for data (Brunton and Nissenbaum 2011). Huang et al. (2017) discuss the business model of WeCash, which offered its online credit rating service for free for customers who submitted personal and social media account data. The company monetized this data through a variety of financial services products to potential credit-hungry customers.

Data-driven servitization involves business models equipping traditional products with complementary data services. Manufacturers might equip their machines with monitoring technologies that generate data to complement offerings with use, malfunction detection, and diagnosis and prognosis services (Altman and Linder 2019). These services help to avoid or improve response times to breakdown, to improve speed of repairs, and to assess a customer's use of the machine (Batista et al. 2017).

Either as a byproduct of operations or intentionally, many organizations generate large quantities of data. But generation of data does not ensure their capability to benefit from that data. Some organizations lack capabilities related to effective internal data strategies, processes, and cultures. Data go unused or are too costly to use because the organization lacks organizational support for data standards, data stewardship, data cataloging, and data cleaning. Organizational units and groups might not even know what data exist in the broader organization, who has what data, or how to access it. In a pharmaceutical company we visited, data generated from clinical trials was subject to both external and internal regulations that required complex approval processes for access. Moreover, data curation and data cataloging were seen as volunteer efforts and consequently were not rewarded or recognized in the organization. "Not my job" was the motto. The organization did not consider itself to be in the data business and hence did not invest in these capabilities. As a result, poor data access, poor data quality, and data incompleteness can render the internal data too difficult to use.

Competitive behaviors also can result in nonuse of internal data. Units or individual employees might treat data as their property and embargo it from broader use. One industrial researcher collaborating with academic counterparts told us that the data needed, even by a research team, too often was located in "a certain professor's hard drive." In some cases, internal data are stored, not for reasons of reuse, but for regulatory reasons—in case of questions or possible errors. Internally, employees show a lack of willingness to render data for reuse, justifying their protectiveness on the grounds of disapprobation of "data parasites." For these and other reasons, even organizations well endowed with internal data often find themselves procuring data externally.

Some organizations well endowed with data, and also equipped with the necessary strategies, processes, and culture to benefit from it, still find external data sourcing to be paramount. In academia, such sourcing decisions are increasingly common because no research team or entity alone has data sufficient to solve complex problems (Vermeulen et al. 2013). The Cancer Moonshot calls for global pooling of data to enable research breakthroughs. In fields such as climate science and molecular biology (e.g., genetics and marine biology), we see the emergence of large open data infrastructures (see, e.g., Vermeulen et al. 2013). Sorescu (2017: 695) discusses that in order to predict patient outcomes and deliver better care, hospitals are building patient profiles that "combine vitals and care history with external data such as diet scores from MyFoodDiary or exercise scores from FitBit". Personalized (or precision) medicine programs call for "breaking down data silos" and moving toward effective sharing of genomic, epidemiological, and clinical data within and across organizations to transform clinical care in healthcare systems (Stark et al. 2019: 18).

The wider and larger accumulation of data and the greater demand for data have increased sourcing opportunities but also the complexities and uncertainties of data use (Leonelli 2015). The complexity stems not just from the various regulatory regimes that restrict access and/or use far beyond the typical contractual agreements in IS sourcing. Complexities arise from the instability and indeterminacy of data, which makes it very challenging to package data as a service. This is in contrast to software applications, business processes, and data center offerings that are packaged as services; contracts are output-based, and expected performance and contracting parties are specified ex ante.

With data, users, inputs, and outputs represent much indeterminacy. Sourcing of data often involves combining data originally generated in different contexts for different purposes with different data collection and processing practices. These co-dependences add to instabilities. Data are used for purposes not even imaginable at the time of data collection. Countless stories in the popular press tout how varied machine learning projects have failed because of unresolved—often hidden—discrepancies between the internal and external data being combined (Korolov 2019). Some even question the rationale for data sourcing because of the risks involved (Soundararajan et al. 2018).

Importantly, these complexities are not discussed in the few studies that have been published in data sourcing. A good illustration is Azoulay (2004: 1591) who found that pharmaceutical firms are more likely to outsource *data production*—"the routine manipulation, storage, and transfer of symbolic information within established categories"—compared to "*knowledge production*—the establishment of novel conceptual categories, hypotheses, and causal associations outside the firm." Azoulay conveys a very unproblematic view of data.

5 Prevailing, and at Times Implicit, Views About Data in the IS Literature

Here, we discuss prevailing, and at times implicit, views of data in the IS literature. We expand beyond the sourcing literature because this literature has been silent about data, including its views of data.

Commodity View. Much of the IS literature presents a token view, or what we call a commodity view, of data. Data refer to databases, as in testing software programs (e.g., Chen et al. 2017); to data traces, as in real-time systems (Pigni et al. 2016); or to data records, as in information artifacts (Tallon et al. 2013). Datasets and data traces view data as raw, unprocessed facts that can be in different formats, such as text, numbers, images, sound; and so on (Pigni et al. 2016; Abraham et al. 2019). Tallon et al. (2013) address data as information artifacts. Information artifacts include data records, files, documents, and images. To Tallon et al. (2013), physical and logical are inseparable; they warn against an examination of information artifacts without considering physical artifacts, such as data storage, virtualizing systems, database technologies, and dashboards.

In contrast, Blair (1984) makes a separation between physical and logical artifacts—for example, data storage as physical and data access as logical. He also differentiates between repositories of documents and repositories of data, in terms of retrieval. He perceives retrieval of data as straightforward because the request for data could be easily formulated in the form of a query. Blair (1984) sees data as objective facts and as originating from a single truth. He sees documents, on the other hand, as indirect and indeterminant because documents could have many access points, and the document could encapsulate different ways of processing data that could have changed the data (or facts) in some way. More recently, McKinney and Yoos (2019) define data as "given"—as "symbolic representations of observable properties; they are facts, measures or descriptions of objects" (McKinney and Yoos 2019). Data are assumed to represent an objective measurement or trace of an entity or phenomenon.

This commodity view of data prevails in the IS sourcing literature (Chen et al. 2017) and in writings about interorganizational supply chain data flows (In et al. 2019). As noted, Azoulay (2004) projected an unproblematic view of data sourcing, based on this same commodity view of data in discussing data sourcing in a pharmaceutical company.

In the context of platforms, scholars in IS view data as unproblematic, structured, and homogenous digital material of 1s and Os (Yoo et al. 2010; Kallinikos et al. 2013). Data are treated as a technical object and are seen as easily editable, processable, and distributable. Such technological artifacts are largely considered to be culturally agonistic and value-unbounded. In the article titled "Big data and management," George et al. (2014) similarly discuss the technical aspects of data. They emphasize the granularity of data in discussing how much data each Formula 1 car generates in a race and in discussing how social media captures every move a person makes. In big data conversations, volume, velocity, variety, and veracity are equated with improved representations and understandings of the world.

The commodity view envisions easy travel for data. For example, Piccoli and Pigni (2013: 56) write, "on TripIt, it [data] can be streamed to other partners, who can harvest it and create value-added services based on the TripIt platform. Expensewatch.com, for example, integrates a TripIt itinerary automatically to compute expense reports." In this commodity view, ownership of data can be determined and enforced via digital rights management systems.

Processual View. In an intra-organizational context, Jones (2019) studies electronic medical records in acute hospitals and paints a rather different view of data, much messier and more complicated than that rendered by the commodity view. Jones (2019: 3) refutes the notions of data as "givens that are out there in the world" and as "being a referential, natural, foundational, objective and equal representation of the world." Jones (2019: 3) calls for more research to focus on understanding both the processes by which data come into being—or what he calls 'data in principle'—and the processes that involve how data are used—or what he calls 'data in practice.' Jones (2019: 10) argues that "unless data are sought, selected, extracted, and interpreted, they cannot inform." Data that cannot inform are not data to a viewer. Rather, a viewer just sees dots, letters, or numbers. Jones (2019) builds on other researchers outside of healthcare who have called for a greater understanding of how data in digitized form are materialized in practice (Orlikowski and Scott 2014); how data's potential becomes actualized as information (Aaltonen and Tempini 2014); and how our current knowledge (e.g., meaning-making systems) circumscribe the data but also control and limit the data (Tuomi 1999–2000). The dynamic interdependencies between data and the processes by which data come into being and are used necessitate huge investments in data update and maintenance processes.

Reuse of data can have transformative implications in the processual view (Jones 2019). As processes are applied to data, new questions arise. These questions can affect the making, sourcing, and use of data (Markus 2001). Alaimo et al. (2019) underscore the ontological instability of data ("what they are, how they are produced") and an epistemological understanding ("what data are supposed to refer to or convey") as data is transformed and reconfigured during its journey from initial generation to delivery of goods and services. Wadmann et al. (2013) discuss how reuse of hospital data can result in changes in data generation and data documentation when health professionals learn that the data are used to monitor their behavior and not just assess patients' progress. Monteiro and Parmiggiani (2019) underscore how organizations regulate how others can leverage open datasets, and by doing so, they ensure that data are used for desired political purposes.

Relational View. Whereas the commodity and processual views are present in the IS literature, the relational view is less so. An exception is Winter and Davidson (2019) who examine how data travel across different use contexts. Outside the IS literature, the relational view is evident with Leonelli (2015), who writes about research data communities (e.g., in biology). Similar to Jones (2019), Leonelli (2015: 810) refutes data as given and as providing truth-value. She sees a difference between the tangible and intangible forms of data, or between tokens and types of information. Tokens reflect a physical instantiation of a form of data that can take on different types. Leonelli (2015) also distinguishes between raw data and data as a unique source of information. Raising the question of "What counts as scientific data?", she explains that what is conventionally thought of as "raw data" are just a collection of symbols that are not organized to represent a phenomenon. As soon as data are arranged as types to represent a phenomenon, data become theory or model laden. For example, a click stream as marketing data is embedding assumptions of marketing.

Leonelli (2015: 817) argues that data that counts is data that are tailored for those who use it, "how and for which purposes." According to her, two aspects render objects as data: (1) promissory status as evidence, and (2) portability. The promissory nature means that objects become data only when those objects fit the stakeholders' or users' purpose, such as an inquiry or question. The inquiry or question reflects the interests or backgrounds of the inquirers, as well as the procedures and mediums that are viewed as legitimate in the inquirer's community. Portability relates to the communal nature of the scientific evidence-making. Others in the research community must be willing to corroborate the claims. Hence, data not only are circumscribed by processes and current knowledge, as in the process view, but data also are "a result of complex processes of interaction between researchers" in evidence making and claiming.

The relational view has implications for data ownership. Leonelli (2015) argues that the distinction between token and types can be useful in determining authorship or other forms of property claims, but the more data travel, the harder the identification of who counts as their author and/or owner becomes. She argues that ownership grows in ambiguity as the distance the data travel from the format and medium in which the data were originally captured grows. New purposes of the data, how data are fitted to new approaches and landscapes, require modifications and reformatting of the data. The relationships (i.e., how data are fit to a specific purpose) are what render data's value but value is also impacted by contextual and interpretative challenges.

6 Data Partnerships: Organizational Arrangements Under Different Data Views

External data relationships, in turn, require organizations to make decisions to source data, establish contractual structures, and manage the sourcing relationship. Next, we view how sourcing decisions and organizational arrangements are affected by different data views.

Commodity View. The commodity view promotes the strategy of external sourcing based on transaction costs. When transactional characteristics take a nuts-and-bolts commodity form (e.g., marketing data), opportunism and investments in relationship-specific assets are minimized. In the commodity view, data are merely a resource for a final good or service. Contracts assume ex ante predefined uses and users. The more homogenous are the data in format and content, the lower are the transaction costs and hence more attractive the decision to source data.

But companies do not just source one type of data from one source. Increasingly, organizations source varied types of data from many sources to integrate the data into a final service or good. Rothe et al. (2019) report how entrepreneurial firms leveraging genomics data from open pools combine these data with other proprietary data from other external proprietary sources. These external data relationships require organizations to make decisions to source data in the first place and then to govern the data asset and manage the sourcing relationship.

When data are heterogeneous in format and content, the transaction costs rapidly escalate and decision to source data become less attractive (Koutroumpis and Leiponen 2013). When the data are originally collected in rather different contexts, with different practices, and for different purposes, costs escalate. Heterogeneous data increase the difficulty of agreeing to ex ante uses and to restrictions to data combinations. Integration of data from different legal use jurisdictions substantively increase liabilities. Winter and Davidson (2019) report on legal proceedings when health data from the restricted public domain travelled to the for-profit corporate domain. Researchers in market research who have studied inter-organizational data exchanges report large conflicts between data contractors and data clients because of the different incentives between the producers of data collections and the users of these collections (French and Ebner 1986).

Organizations might source data from a variety of parties that may be public (e.g., governments and universities), private (e.g., customers and suppliers), or community based (e.g., consortia) (George et al. 2014). Data might be based on exchange that is either closed (i.e., pecuniary) or open (i.e., non-pecuniary). When open, the openness can vary. For example, in terms of access, the data might be accessible but only under certain restrictions that render it pseudo-anonymous.

Under the commodity view, data are offered "as is" in both open and closed exchanges. Citing Koutroumpis and Leiponen (2013: 3222), George et al. (2014) noted that "the realm of big data-sharing agreements remains informal, poorly structured, manually enforced, and linked to isolated transactions." In IS sourcing, sourcing contracts about applications, business processes, or data centers often are performance based and transfer the risks from the buyers to the suppliers; in contrast, in data sourcing, organizations that source have little recourse if the data they have acquired do not meet their needs.

With commodity view, data partnerships are transactional bilateral or "standard" brokered contractual relationships in pecuniary exchanges; if non-pecuniary, then they are often pooled based on open exchanges. A transactional bilateral relationship might be set in the context of data-servitization between a machine supplier and customer, where data are a by-product of products used in the field (Altman and Linder 2019).

A recent report by Wixom (2019) examines data monetization and how to increase return from data and analytics. Rather than a buyer perspective to sourcing, the report takes a seller perspective. Among her recommendations include "developing APIs to extend services to external customers and partners" (Wixom 2019: 9). Interestingly, Wixom's report states that "data partnering is not required for successful data monetization." Still, the report acknowledges, "data partnering can be helpful in situations in which a company needs access to a key resource, such as an external data source…" (Wixom 2019: 14). Wixom lists four key practices for data partnering: "(1) identifying/evaluating new partners, (2) incentivizing organizations to partner, (3) making explicit how value will be shared, and (4) making explicit how partnership conflicts will be communicated and resolved" (Wixom 2019: 14).

On a multilateral level, the pecuniary exchanges of data have many challenges. The challenges include the weak intellectual property protection for data (Koutroumpis and Leiponen 2013), agreeing on technical data specifications, and conflicting data use strategies (Batista et al. 2017). There might be insufficient information about the quality and legal status of data. To provide any sort of guarantees, the contracts would have to impose strict use restrictions, enforcement of data rules, and strong penalties for violations, all of which escalate transaction costs and render data sourcing less attractive.

Example of Commodity View. Open pool exchange is common in academic research partnerships. An open-pooled exchange was practiced in the Human Genome Project. Norms and public funding rules initially helped to grow this public commons and to make data widely accessible to the research community. Research funding agencies mandated that researchers with publicly funded projects deposit their data and make it available for reuse by others (Lee 2015). The research community developed the Bermuda Principle, requiring that DNA sequence data be deposited in an open data repository, such as GenBank in the United States, within 24 h after data collection. The initial set-up assumed a very unproblematic view of data.

However, the wide open access in GenBank led to disputes related to medical, ethical, legal, and privacy concerns (Lee 2015). Over time, the openness narrowed in terms of both access and use. Researchers had to be “approved” for access. The user of data was not allowed to identify or approach research subjects, or make ownership claims for the datasets that were derived from the data (Lee 2015). Other regulations included difficult-to-implement mandates—for example, that the research results needed to benefit individual patients (Lee 2015). The platforms also restricted who could modify the data records. Only the original contributor could update a record in GenBank because the system did not allow peer editing. Incentivizing original data contributors to keep their records up to date proved difficult. Researchers also grew concerned about getting credit for their data collection efforts. Repositories began to establish policies that would embargo the use of the data until the original data generators had published their findings (Lee 2015). But even as the open data approach became a more bounded data approach over time, exchange remained non-pecuniary and helped to facilitate the exploitation of data across thousands of researchers across the world (Wellcome Trust 2003).

Similar to the open data approach in the Human Genome project, the Census of Marine Life in the Ocean Biogeographic Information Systems has developed a world wide open access repository for marine life data. In addition, Leonelli (2016: 13) conducted a multiyear ethnography of the plant science community; this global community, called iPlant, aimed to build a global open data collaboration with “many data types—ranging from genetic to morphological and ecological [data].” iPlant has had mixed success because it faces many heterogeneous stakeholder groups and competitive use challenges.

Process View. The sourcing decisions with the commodity view parallel those of IS sourcing that are concerned with transaction characteristics; in contrast, the process view is driven by the view that data are temporal, co-dependent, indeterminant, and pervasively editable. These characteristics shape both the sourcing decision and sourcing partnerships.

Because data are indeterminant, as are the uses, the process view of data suggests sourcing decisions and organizational arrangements that are control-based, rather than transaction cost-based. This view is less focused on the data as a resource to a delivered service or good and more focused on the value of entanglement of data and operations on data that could take place at any point, from the source to the final reuse. Value is created from various data and operations entanglements, or what Alaimo et al. (2019: 1) call “[d]ata commodities, such as advertising audiences, personalized suggestions, indexes, scores, and rankings.” Remaining in control of these data commodities and appropriating value as they traverse the data value chain are paramount.

The process view of data suggests organizational arrangements that are heavily technological and where data can be parsed and controlled by varied and decentralized stakeholders (Alaimo et al. 2019; Koutroumpis et al. 2017). The account of value creation cannot be predetermined; it might occur at any point as the data are transformed and combined (Alaimo et al. 2019). Such an unfolding process might involve the world of distributed ledger technologies, which can manage the provenance from data source to its final delivery (Kazan et al. 2018; Beck et al. 2018). The arrangements would involve technological capabilities, where the data suppliers would independently verify data, rather than the users or intermediaries, as under the commodity view (Koutroumpis et al. 2017). In essence, the process view would privatize all data movements. Trustworthiness of the actors, data quality, and enforcement of data consent rules and other restrictions for data would be technologically processed in a de-centralized way through executable contracts. The history of data would automatically be produced because all transactions are traceable. Such marketplaces would take not the form of a platform with a central producer—except perhaps one initially managing different sides—but the form of a community where all can register their own data with the distributed ledgers and can decide whether to release it for pecuniary or non-pecuniary exchanges in a direct interaction with any other party. Such organizational arrangements are severely limited by the scalability of such systems because of the high transaction costs involved.

An Example of Process View. One example of a data arrangement that has a strong process view is the Multicenter AIDS Cohort Study (Ribes and Polk 2015). The organizations participating in this study, except one, agreed to high levels of standardization among the samples of eligible men, specimens and data, and calibrated apparatuses. The cross-site standardization of data and of processes applied to data was done to increase the availability of data resources and services for the at-risk population. Yet, the one organization that did not participate in cross-site data and process standardization around at-risk cohorts took a broader population sampling approach and was able to identify the causes of AIDS.

Relational View. The relational view of data involves sourcing decisions and organizational arrangements based on trusted relationships. These relationships would be either bilateral or multilateral, but if they are multilateral, they are brokered through a trusted intermediary. The bilateral relationships could be traditional, firm-sponsored, contract research with academic or industrial researchers. For example, the consumer personal genetics companies, such as 23andMe, establish bilateral research relationships for joint research projects with academic and industrial research organizations. In these bilateral trusted relationships, value from data is co-generated.

At the multilateral level, the organizational arrangements often take the form of a consortium or a common pool with a closed membership. The exchanges can be pecuniary or non-pecuniary. These arrangements are analogous to what Koutroumpis et al. (2017) call a "collective multilateral marketplace," which "adopts strong boundaries via complex contracts, clear rules such as bylaws and procedures to collectively change them, and effective monitoring and enforcement practices." But rather than one central actor making decisions, as in a data marketplace with the central producer, the organizational arrangement involves the collective making decisions, such as agreeing on technical data specifications and legitimate value capture strategies. Hence, the control remains with the members. Still, the organizational arrangements involve higher transaction costs than those with the commodity view of data because partners have to be screened and the complex and comprehensive contracts have to be negotiated.

An Example of Relational View. Perkmann and Schildt (2015) report on the Structural Genomics Consortium, which involved life science research partnerships to facilitate broad research collaboration across industry and academia. The purpose was to speed up research results that would lead to new drug discoveries. The consortium involved semi-open collaboration with the trusted intermediary, which served the role of a boundary organization. The funding came from membership fees from the industrial organizations, government, and charities. To quality for public funding, the research results would be published and would become part of the public domain. The consortium required open releases of data by participating industrial and academic research organizations.

The industrial organizations faced two incentive challenges. First, they had little incentive to participate if they could not shape the agenda, but doing so would also reveal their R&D strategy. Second, the firms had little incentive to participate if they could not get the academic scientists to pursue firm goals, rather than their own or broader scientific goals. To overcome these challenges, the consortium faced difficult negotiations in terms of its bylaws and procedures, as well as complex contracts. Its meeting to establish agreement on the Articles of Association was nearly twice as long as the initial scientific committee meeting. Its meeting to agree on contracting terms was nearly twice as long as its meeting on the Articles of Association.

To address these incentive challenges and avoid contentious negotiations, the consortia worked through a trusted intermediary. Members released their planned research uses (e.g., the wish list of proteins to be studied) only to the intermediary in order to avoid the leakage of members' strategic R&D priorities. The intermediary generated an anonymous master list out of these wish lists and mediated its circulation. Beyond the trusted intermediary, only the board of directors and the scientific community that approved all scientific projects and decisions had access to the master list. The mediated and limited disclosure by the trusted intermediary received pushback from academic researchers, who argued that it led to inefficiencies and overlapping investigations. However, this confidentiality was deemed critical because it was necessary for the research teams to be willing to focus on "sensitive, high-priority targets" (Perkmann and Schildt 2015: 1138).

7 Final Thoughts

This chapter is a call for the IS field to expand its research on IS sourcing to include a focus on data sourcing. Neither IS sourcing nor the related areas of platforms and infrastructures in the IS literature have addressed data sourcing, except peripherally. Scholars researching big data analytics (e.g., Lehrer et al. 2018) focus on algorithms, not data and its sourcing.

To begin to explore data sourcing and data partnerships, we advance some implicit views of data in the IS literature that have not yet been well articulated: commodity view, process view, and relational view. A commodity view of data prevails in the IS research. To some, all that is needed to source data are "APIs for accessing sensor data, clickstream data, [and] social media data" (Lehrer et al. 2018: 448). The right technology is assumed to render low transaction costs in sourcing externally. The process and relational views of data reflect a much more complex and unstable views of data and introduce many more considerations for sourcing and sourcing partnerships.

Our key argument in this chapter is that the key data sourcing decisions are shaped by the viewers' view of data. That is, whether to source, how to contract, and how to manage sourcing relationships are influenced by the view of data. These views bring up considerations that are not obvious or explicit in the IS sourcing literature, which can present opportunities for future research on data sourcing and data partnerships.

In their MIS Quarterly curation, Kotlarski et al. (2018) write that "in various sourcing models that are typically based on the distinction between *ownership* (in-house or third party) and *location* (domestic, nearshore, or offshore), as well as in online sourcing models, the location decision of sourcing has been critical" (italics in original). In terms of data ownership, the commodity view assumes that ownership is fixed, or non-morphing, as data travel. Enforcing ownership depends on the technologies used to trace and control the data. The process and relational views suggest that data are transformed when data are differentially processed and purposed by different stakeholders, and hence the decision rights—as well as the control over data—transform in the process, and often the ownership becomes highly distributed.

The data views also shape the location decision. In the commodity view, the location is immaterial—it does not matter. The process view promotes global but vertical "industry" or discipline data arrangements because meta-data standards are likely to be industry- or discipline-specific and because vertical structures allow for more consistency in terms of understanding the lawful processes on data that can influence interpretation and meaning making. The relational view promotes cross-industry or cross-disciplinary data arrangements because complementary behaviors would be emphasized and competitive behaviors would be contained. In the relational view, decisions are likely to prioritize local sources over global sources to render the complex data contracts more enforceable.

Kotlarski et al. (2018) discuss two other critical decisions: designing contractual structures and managing the sourcing relationship. In designing contractual structures, they address two elements: "the degree of hierarchical elements in different contractual structures" and "the division of risks and incentives between the client and supplier." How the incentive alignment is designed as well as enacted affects interactions.

The nascent literature on data sourcing reports either highly standard, "as is" contracts or informal tacit contracts. Under the commodity view, data sourcing is low in transaction costs because the focus is on the right tools, such as APIs, for sourcing data. The perspective is dismissive or unaware of the challenges of data's arising from different processes and interpretative frames. Neither opportunism nor asymmetric investments are of concern, and hence incentive alignment is not a consideration. This commodity view is projected into research on data collaboratives and data pools, which then assumes that organizations in such collectives simply need to release, post, and or retrieve data (Susha et al. 2017). Only data ownership and various regulatory and ethical constraints to data access are commonly considered as impediments to sourcing of data.

The relational view provides a starkly different view, involving complex contractual structures and interactions with trusted intermediaries. The decisions are far from the simple "buy" transactions of the commodity view. In the relational view, there often is an obligation to share the results (although not necessarily the materials used to generate the results) so that the other stakeholders remain "owners" of data and also participate in future data transformations. Similarly, the process view presents more complex contractual structures but ones that can become highly algorithmic in terms of contract design and execution.

The second decision advanced by Kotlarski et al. (2018) relates to "managing ongoing sourcing relationships"—specifically regarding strategies and practices pertaining to contractual or relational governance and adjustments and mechanisms used. In the commodity view of data, sourcing relationships are either ad hoc or transactionally routine. In the relational view of data, the transformations of data also transform the relationships (Ribes and Polk 2015). Much of the research so far on the relational view comes from science studies that report how the data practices (e.g., standardization of data) have substantively reduced the autonomy of lab researchers (Vermeulen et al. 2013).

Future research is encouraged to study empirically our central tenet: that the view of data fundamentally shapes the sourcing decisions and sourcing arrangements. We particularly encourage studies that explore the process and relational views of data further. For example, understanding the implications of the process view is timely because of the high levels of interest in algorithmic management in academia and industry. Much more is needed to understand how ontological instability and epistemological uncertainty is managed in sourcing decisions and sourcing partnerships.

References

Aaltonen, A., Tempini, N. (2014). Everything counts in large amounts: a critical realist case study on data-based production. *Journal of Information Technology*, *29*(1), 97–110.

Abraham, R., Schneider, J., & vom Brocke, J. (2019). Data governance; A conceptual framework, structured review, and research agenda. *International Journal of Information Management, 49,* 424–438.

Alaimo, C., Kallinikos, J., & Aaltonen, A. (2019). *Handbook of digital innovation*. Oxford University Press.

Altman, P., & Linden, M. (2019). Managing emerging (Mis)alignments in data-driven servitization. *Research-Technology Management, 2019,* 37–45.

Azoulay, P. (2004). Capturing knowledge within and across firm boundaries: Evidence from clinical development. *American Economic Review, 94*(5), 1591–1612.

Batista, L., Davis-Poynter, S., Ng, I., & Maull, R. (2017). Servitization through outcome-based contract—A systems perspective from the defense industry. *Inernational Journal of Production Economics, 192,* 133–143.

Beck, R., Muller-Bloch, C., & King, J. L. (2018). Governance in the BlockChain economy: A framework and research agenda. *Journal of Association for Information Systems*.
Birney, E., Vamathevan, J., & Goodhand, P. (2017). Genomics in healthcare: GA4GH Looks to 2022.
Blair, D. C. (1984). The management of information: Basic distinctions. In *MIT Sloan Review* (Vol. 26, pp. 13–23). Fall.
Brohman, M. K., Watson, R. T., Piccoli, G., & Parasuraman, A. (2003). *Communications of the ACM, 46*(6), 47–51.
Brunton, F., & Nissenbaum, H. (2011). Vernacular resistance to data collection and analysis: A political theory of obfuscation. *First Monday*, *16*(5).
Chen, Y., Bharadwaj, A., & Goh, K.-Y. (2017). An empirical analysis of intellectual property rights sharing in software development outsourcing. *MIS Quarterly, 41*(1), 131–161.
Constantinides, P., Henfridsson, Ol, & Parker, G. G. (2018). Platforms and infrastructures in the digital age. *Information Systems Research, 29*(2), 381–400.
Constantiou, I. D., & Kallinikos, J. (2015). New games, new rules: Big data and the changing context of strategy. *Journal of Information Technology, 30,* 44–57.
de Corbière, F., & Rowe, F. (2013). From ideal data synchronization to hybrid forms of interconnections: Architectures, processes, and data, *Journal of the Association for Information Systems*, *14*(10).
de Reuver, M., Sorensen, C., & Basole, R. C. (2017). The digital platform: A research agenda. *Journal of Information Technology*.
Earl, H., McKinney, Jr., Charles, J., & Yoos II. (2019). Information as a difference: Toward a subjective theory of information. *European Journal of Information Systems*, *28*(4), 355–369.
French, W., & Ebner, M. (1986). A practical look at research ethics. *Journal of Data Collection, 26,* 49–53.
Gainer, V. S., Cagan, A., Castro, V. M. et al. (2016). The biobank portal for partners personalized medicine: A query tool for working with consented biobank samples, genotypes, and phenotypes using i2b2. *Journal of Personalized Medicine*, *6*(11).
George, G., Haas, P., & Pentland, A. (2014). Big data and management. *Academy of Management Journal, 46*(6), 321–326.
Ghazawneh, A., & Henfridsson, O. (2013). Balancing platform control and external contribution in third-party development: The boundary resources model. *Information Systems Journal, 23*(2), 173–192.
Huang, J., Henfridsson, O., Liu, M. J., & Newell, S. (2017). Growing on stereoids: Rapidly scaling the user base of digital ventures through digital innovation. *MIS Quarterly, 41*(1), 301–314.
In, J., Bradley, R., Bichescu, B. C., & Autry, C. W. (2019). Supply chain information governance: Toward a conceptual framework. *The International Journal of Logistics Management, 30*(2), 506–526.
Jones, M. (2019). What we talk about when we talk about (Big) data. *Journal of Strategic Information Systems, 28,* 3–16.
Kallinikos, J., Aaltonen, A., & Marton, A. (2013). The ambivalent ontology of digital artifacts. *MIS Quarterly, 37*(2), 357–370.
Kazan, E., Tan, C.-W., Lim, E. T. K., Sorensen, C., & Damsgaard, J. (2018). Disentangling digital platform competition: The case of UK mobile payment platforms. *Journal of Management Information Systems*.
Khatri, V., & Brown, C. V. (2010). Designing data governance. *Communications of the ACM, 53*(1), 148–152.
Kitchin, R. (2014). *The data revolution*. Thousand Oaks, CA: Sage Publications Inc.
Konning, M., Westner, M., & Strahringer, S. (2019). A systematic review of recent developments in IT outsourcing research. *Information Systems Management, 36*(1), 78–96.
Korolov, M. (2019). 6 Reasons Why AI projects Fail. CIO Magazine.
Kotlarski, J., Oshri, I., Dibbern, J., & Mani, D. (2018). MIS quarterly research curations. IS Sourcing. www.misq.org.

Koutroumpis, P., & Leiponen, A. E. (2013). Understanding the value of (Big) data. In *2013 IEEE International Conference on Big Data* (pp. 38–42).

Koutroumpis, P., Leiponen, A., & Thomas, L. D. W (29 July 2017). The (Unfulfilled) potential of data marketplaces. *ETLA Working Papers No 53*. http://pub.etla.fi/ETLA-Working-Papers-53.pdf.

Lacity, M. C., Khan, S. A., & Yan, A. (2016). Review of the empirical business services sourcing literature; An update and future directions. *Journal of Information Technology, 31,* 269–328.

Lee, P. (2015). Centralization, fragmentation, and replication in the genomic data commons. In *University of California Davis Legal Studies Research Paper Series*, No. 448.

Lee, S. U., Zu, L., & Jeffery, R. (2017). Data governance for platform ecosystems: Critical factors and the state of practice. In *Data Governance for Platform Ecosystems. Proceedings of Pacific Asian Conference on Information Systems*, Association for Information Systems.

Lehrer, C., Wieneke, A., Von Brocke, J., Jung, R., & Seidel, S. (2018). How big data analytics enables service innovation: materiality, affordance, and the individualization of service. *Journal of Management Information Systems*, *35*, 2 424–460.

Leonelli, S. (2015). What counts as scientific data? A relational framework. *Philosophy of Science, 82*(5), 810–821.

Leonelli, S. (2016). *Data-centric biology*. Chicago, IL: The University of Chicago Press Books.

Lyytinen, K., Sørensen, C., & Tilson, D. (2017). The generativity of digital infrastructures: A research note. In R. D., Galliers & M.-K. Stein (Eds.), *Handbook on IS companion book*, Routledge.

Markus, M. L. (2001). Toward a theory of knowledge reuse: Types of knowledge reuse situations and factors in reuse success. *Journal of Management Information Systems, 18*(1), 57–93.

Markus, M. L. (2016). Obstacles on the road to corporate data responsibility. In C. R., Sugimoto, H. R., Ekbia, & M., Mattioli (Eds.), *Big data is not a monolith (Information Policy)*. MIT Press.

Miller, A. R., & Tucker, C. (2014). Health information exchange, system size, and information silos. *Journal of Health Economics*, *33*, 28–42.

Monteiro, E., & Parmiggiani, E. (2019). Synthetic knowing: The politics of the internet of things. *MIS Quarterly, 43*(1), 167–184.

Orlikowski, Wanda J., & Scott, S. V. (2014). What happens when evaluation goes online? Exploring apparatuses of valuation in the travel sector. *Organization Science, 25*(3), 868–891.

Otto, B. (2011). Organizing data governance: Findings from the telecommunications industry and consequences for large service providers, *Communications of the Association for Information Systems*, *29*(3).

Parker, G. G., Van Alstyne, M. W., & Choudary, S. P. (2016). *Platform revolution: How networked markets are transforming the economy and how to make them work for you*. New York: WW Norton & Co.

Perkmann, M., & Schildt, H. (2015). Open data partnerships between firms and Universities: The role of boundary organizations. *Research Policy, 44,* 1133–1143.

Piccoli, G., & Pigni, F. (2013). Harvesting external data: The potential of digital data streams. *MIS Quarterly, 12*(1), 53–64.

Pigni, F., Piccoli, G., & Watson, R. (2016). Digital Data Streams: Creating Value from the Real-time Flow of Big Data. *California Management Review, 58*(3), 5–25.

Ribes, D., & Polk, J. B. (2014). Flexible relative to what? Change to research infrastructure. *Journal of the Association for Information Systems*, *15*(Special Issue), 287–305.

Ribes, D., & Polk, J. B. (2015). Organizing for ontological change: The Kernel of an AIDS research infrastructure. *Social Studies of Science, 45*(2), 214–241.

Rothe, H., Jarvenpaa, S. L., & Penninger, A. (2019). How Do entrepreneurial firms appropriate value in bio data infrastructures: An exploratory qualitative study. In *European Conference on Information Systems*, Stockholm, Sweden.

Schreieck, M., Wiesche, M., Krcmar, H. (2016). Design and governance of platform ecosystems—Key concepts and issues for future research. In *Twenty-fourth European Conference on Information Systems (ECIS) Proceedings*, Istanbul, Turkey.

Shantz, A. S. (2018). Big data, bigger questions: Data-based business models and their implications for organizational boundaries, data governance, and society. *Research in the Sociology of Organizations, 57,* 305–329.

Sorescu, A. (2017). Data-driven business model innovation. *Journal of Product Innovation Management*, *34*(5), 691–696.

Soundararajan, V., Khan, Z., & Tarba, S. Y. (2018). Beyond brokering: Sourcing agents, boundary work, and working conditions in global supply chains. *Human Relations, 71*(4), 481–509.

Star, S. L., & Ruhleder, K. (1996). Steps toward an ecology of infrastructure: Design and access for large information spaces. *Information Systems Research, 7*(1), 111–134.

Stark, Z., Boughtwood, T., Phillips, P., Christodoulou, J., et al. (2019). Australian genomics: A federated model for integrating genomics into healthcare. *The American Journal of Human Genetics, 105,* 7–14.

Steinfield, C., Markus, M. L., & Wigand, R. T. (2011). Through a glass clearly: Standards, architecture, and process transparency in global supply chains. *Journal of Management Information Systems*, *28*(2), 75–107.

Susha, I., Janssen, M., & Verhulst, S. (2017). Data collaboratives as a new frontier of cross-sector partnerships in the age of open data: Taxonomy development. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Tallon, P. P., Ramirez, R. V., & Short, J. E. (2013–2014). The information artifact in IT governance: Toward a theory of information governance. *Journal of Management Information Systems*, *30*(3), 141–177.

Tilson, D., Lyytinen, K., & Sorensen, C. (2010). Digital infrastructures: The missing is research agenda. *Information Systems Research*, *21*(4), 748–759.

Tuomi, I. (1999–2000). Data is more than knowledge: Implications of the reversed knowledge hierarchy for knowledge management and organizational memory. *Journal of Management Information Systems*, *16*(3), 103–117.

Vermeulen, N., Parker, J. N., & Penders, B. (2013). Understanding life together: A brief history of collaboration in biology. *Endeavour, 37*(3), 162–171.

Wadmann, S., Johansen, S., Lind, A., et al. (2013). Analytical perspectives on performance based management: An outline of theoretical assumptions in the existing literature. *Health Economics, Policy and Law, 8*(4), 511–527.

Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems, 12*(4), 5–33.

Wellcome Trust. (2003). Sharing data from large-scale biological research projects: A system of tripartite responsibility. In *Report of a Meeting Organized by the Wellcome Trust and Held on 14–15 January 2003 at Fort Lauderdale*, USA. Williams, H.

Winter, J. S., & Davidson, D. (2019). Big data governance of personal health information and challenges to contextual integrity. *Information Society*.

Wixom, B. (2019). Data monetization: Generating financial returns from data and analytics—Summary of survey findings. In *MIT CISR Working Paper*, CISR WP. No. 427.

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). Research commentary—The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research, 21*(4), 724–735.

Mastering Innovation Through Outsourcing

Innovation in Outsourcing—An Empirical Analysis of Outsourcing Vendors' Innovation Approaches

Selina Meiser and Daniel Beimborn

Abstract Companies must drive innovation to stay competitive in today's fast-changing and highly competitive environment. Therefore, they do also involve their outsourcing vendors as strategic partner for innovation. While recently at least some studies have examined innovation-oriented outsourcing partnerships and their outcomes, almost no research has yet focused on the innovation-related actions from an outsourcing vendor's perspective. Our paper explores what kind of initiatives outsourcing vendors implement to create innovation for their customers. It analyzes a unique dataset of more than 830 innovation initiatives, as reported by global outsourcing vendor firms. The analysis identifies 22 different types of innovation initiatives, which are clustered into six different dimensions. Using these results plus nine interviews with outsourcing experts, we have developed a categorization model which allows for categorizing innovation-related vendor initiatives and provides a basis for evaluating the strategic importance of each category for firms searching for or evaluating an outsourcing vendor.

1 Introduction

During the last decades, outsourcing of IT activities and business processes has become a major and mature managerial option for increasing operational efficiency and excellence. While the focus in earlier years was mainly on saving and variabilizing operational cost and on other efficiency-related criteria (Grover et al. 1996; Dibbern et al. 2004), more recently firms have increasingly shifted their focus towards innovation for staying competitive (Gunday et al. 2011). Driven by the growing

S. Meiser
Frankfurt School of Finance & Management, Frankfurt, Germany
e-mail: selina.meiser@fs-students.de

D. Beimborn (✉)
University of Bamberg, Bamberg, Germany
e-mail: daniel.beimborn@uni-bamberg.de

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_5

awareness that innovation is more effective through collaboration with external partners, companies are increasingly looking at their outsourcing vendors as strategic partner for innovation (Aubert et al. 2015).

'Innovation through outsourcing' has been an interesting field of debate. Originally, the majority of researchers had put forward the position that outsourcing increases the risk of losing innovation capabilities and thus creates strategic risks in the long term (Teece 1987; Verwaa 2017). On the other side, some researchers showed that innovation through outsourcing can be successful (Weeks and Feeny 2008; Oshri et al. 2015) but is facing challenges for both sides (Oshri et al. 2018).

Research in this field primarily focuses on the client side and analyzes their challenges for innovation through outsourcing. Though, almost no research has examined the approaches and initiatives performed by outsourcing vendors to offer innovative services and products to their clients.

To create momentum in this field, our purpose is to explore what kind of initiatives outsourcing vendors perform to drive innovation for themselves and their customers. This research question will be examined by analyzing and categorizing innovation-related activities of outsourcing vendors as reported in a unique dataset of 837 innovation initiatives.

In the following, we will first lay the conceptual foundations for our research and summarize the previous literature on innovation through outsourcing, before we introduce our methodology and report the results of our explorative study. Afterwards, we will use interviews with outsourcing experts to evaluate those results. Finally, we discuss implications and limitations before we conclude with an outlook on future research.

2 Background

In this research, we will analyze the innovation activities of outsourcing vendors both in the BPO and ITO sphere. BPO is the outsourcing of IT-intense business services provided by an external vendor (Dayasindhu 2004) while ITO is specified as an organization's decision to contract-out parts or all of its IT assets, employees and procedures (Grover et al. 1996). In the following, we will use the term outsourcing to embrace both ITO and BPO.

The successful management of innovation is essential for corporate growth and transformation (Zahra and Covin 1994). To be successful, companies have to innovate for both the present and the future, with innovation being defined as the process and outcome of organizations transforming ideas into new/improved products and services or processes (Baregheh et al. 2009). Thereby, product innovation relates to new products, services, programs or technologies, whereas process innovation is characterized by changing the way products are developed or by adopting new methodologies or processes in the organization (Tushman and Nadler 1986).

In recent years, many outsourcing clients have started to change their focus from operational cost reduction to looking for added value and innovation through outsourcing (Whitley and Willcocks 2011). Due to the pressure to innovate and to stay competitive, companies have increasingly been looking at their vendors as a strategic partner for innovation (Aubert et al. 2015; Oshri et al. 2015). Clients aim at tapping into their vendors' knowledge and network to gain access to new ideas and methods and thus to drive innovation (Hoecht and Trott 2006; Boehm et al. 2014; Aubert et al. 2015). Access to highly qualified resources and the possibility to share risks for innovation development are additional reasons to innovate through outsourcing. On the other side, the vendor can distribute the innovation among multiple clients for its own economic (and its customers') benefit.

So far, researchers have not agreed whether outsourcing and innovation are positively related (Oshri et al. 2015, 2018), whether outsourcing reduces the innovativeness of the company in the long-term (Teece 1987; Bettis et al. 1992; Whitley and Willcocks 2011; Verwaal 2017), or whether it has no impact on innovation (Weeks and Feeny 2008). Weeks and Feeny (2008) established three types of customer-oriented innovation: *IT operational innovation* refers to generic changes in technology that do not (directly) affect client-specific businesses but only enable the vendor to provide the service at higher quality or lower cost. *Business process innovation* describes developments that change client-specific procedures, whereas *strategic innovation* is characterized by developments that have a significant impact on the client's product and service range (Weeks and Feeny 2008). Weeks and Feeny (2008) point out that *IT operational innovation* through outsourcing is relatively common, *business process innovation* is rare and *strategic innovation* does almost never occur.

This suggests that innovation through outsourcing involves challenges for the vendor as well as for the client (Oshri et al. 2018). Through relying on outsourcing partners and transferring the expert knowledge to them, the company assumes the risk of information leakage and of losing core competencies and its competitive edge to its rivals (Hoecht and Trott 2006; Weeks and Feeny 2008; Boehm et al. 2014). This risk increases when companies involve multiple vendors and more short-term partnerships. Innovation through outsourcing also embraces the risk that vendors transfer innovative practices to the client's competitors (if those are also among the vendor's client portfolio), which thus might even evolve into industry standards. Vice versa, the upside is that the client might obtain additional innovation from the vendor by getting insights from competitors through their vendor (Hoecht and Trott 2006). Accordingly, finding the right balance of sharing risks and implementing adequate incentives for the vendor becomes a substantial challenge in outsourcing management (Weeks and Feeny 2008; Kotlarsky et al. 2015).

To address those challenges, researchers have identified various factors to improve innovation through outsourcing, with trust, leadership, knowledge, governance and the contract being the most common ones (Whitley and Willcocks 2011; Boehm et al. 2014).

Trust is the key enabler to overcome any difficulties in vendor-client relationships with an open knowledge exchange methodology (Boehm et al. 2014). To enable collaborative innovation, a high level of trust has to be built even before a relationship is entered (Weeks and Feeny 2008), and it has to be established on a personal, a competence-based and a motivational level.

Good, top-management-driven *leadership* provides an environment for innovation by creating the possibility to quickly react to changes through effective change management (Whitley and Willcocks 2011; Boehm et al. 2014; Lacity and Willcocks 2014). Leadership focusing on innovation through outsourcing is characterized by effective risk management and a jointly agreed definition of an innovation goal (Whitley and Willcocks 2011). The aspect of *knowledge* involves all issues regarding knowledge gain, loss and know-how sharing. The company has to either retain or regain its in-house technical and business competence (Weeks and Feeny 2008) and it must implement a regular process for exchanging information both between client and vendor, and between IT and business departments. The aim is to generate or customize the innovation strategy based on the gained knowledge (Boehm et al. 2014) and to create mutual familiarity between the partners (Weeks and Feeny 2008; Oshri et al. 2018).

An appropriate *governance* has to be implemented to organize the company for innovation through outsourcing (Whitley and Willcocks 2011; Lacity and Willcocks 2014). This involves supporting the team work between client and vendor (Whitley and Willcocks 2011) and answering questions regarding the process to successfully manage the selection of ideas being developed, their implementation and the compensation for each party (Weeks and Feeny 2008). Moreover, specific measures have to be set up to monitor the innovation activities. This gives the company a possibility to control innovation costs and benefits (Weeks and Feeny 2008).

Overall, this summary of previous research on innovation through outsourcing reflects the predominant focus on the client side; Kotlarsky et al. (2016) is one of the rare studies focusing on the vendor's side. By analyzing an outsourcing vendor. They discovered four categories of activities which drive innovation for the clients: the *innovation network* facilitates the awareness about innovations within the organization and future events; the *innovation process* consists of a five-step approach which allows to determine the innovation scope, the way to deliver it and evaluate the feasibility; the *innovation assets* category represents R&D labs, dedicated innovation resources, innovation events etc., to enable innovation and collaboration. Vendors do also train their employees in innovation topics which belongs to the *educational and training assets* category (Kotlarsky et al. 2016).

In our research, we want to extend this perspective and look at the activities performed on the vendor side in order to develop capabilities that increase their innovativeness for innovation through outsourcing.

3 Methodology

Our research focuses on the identification of innovation-related initiatives of outsourcing vendors. We used a data set consisting of 365 outsourcing provider performance reports from 2016 to 2018. Each report contained short descriptions of several innovation initiatives which are aimed at increasing the general innovation capability of the vendor in order to serve innovation through outsourcing to their clients. This data was qualitatively analyzed using MAXQDA and following the suggestions from Auerbach and Silverstein (2003). We used open coding as explorative approach to identify organizational, innovation-related, activities (e.g., implementing an innovation lab) in the dataset. In later rounds, categories were formed from comparing these activities. To improve the reliability of our categories we used a sequential approach: the data from 2016 functioned as initial 'training' set. Differences and similarities among identified activities were outlined and for each common activity a new code was generated. Each of the identified text sequences were assigned to one of the codes. After coding the datasets of 2016, around 40 codes were identified. Comparison of the generated codes resulted in 22 separable categories with different specific characteristics (cf. Table 1 in the Findings section below). The codes and categories were then applied to the data from 2017 and 2018, which were coded using the generated category codes. If a described innovation initiative did not fit to one of the existing categories, a new code was generated, which only happened once during the analysis of the 2017 data. Finally, we generated definitions and descriptions for the different categories. The following sections describe the findings of this analysis in detail.

4 Findings

As result of our analysis, 22 categories for innovation initiatives were identified. These categories can be split into innovation enablers, i.e., organizational activities that contribute to a vendor's innovativeness, and innovation outcomes. *Innovation enablers* in turn can be categorized into four dimensions: *Collaboration, People, Structures, and Events*:

- *Collaboration* contains all categories, where the company works together with different external parties. This includes being part of an association, cooperating with academia, startups or involving customers in the innovation process.
- *People* comprises internal procedures of the vendor which enable innovation for people development such as *Employees coaching, Talent management* and *Design Thinking*.
- *Structures* embraces changes of the corporate structure such as setting up new units, launching accelerator and incubator programs or an innovation lab, or acquiring innovation-oriented firms.
- *Events* covers one-time or periodical events improving innovation like c*onferences or hackathons*.

The following figure displays all four dimensions with their corresponding innovation categories (Fig. 1).

Innovation outcomes can be separated into those being implemented on the vendor side vs. those that are client-related.

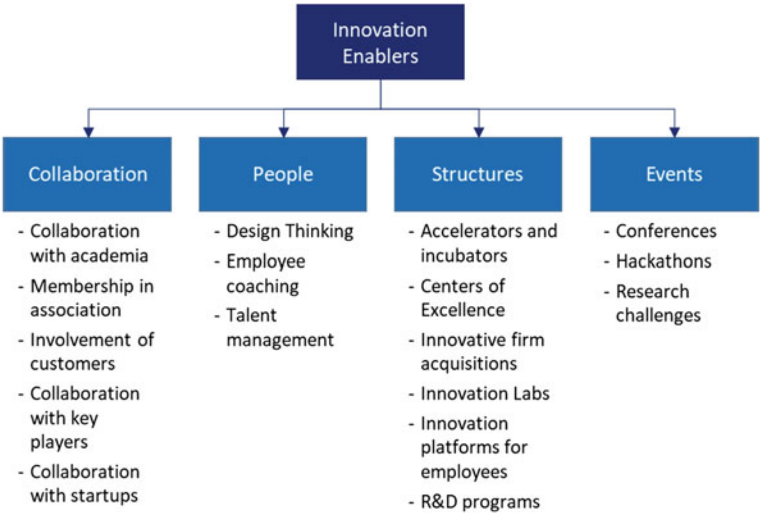

Fig. 1 Enablers of innovation initiatives

- *Client-related*: innovative approaches, or new product & service offerings to be implemented on the client's side.
- *Internal*: innovations implemented internally to increase internal efficiency or effectiveness of serving the clients.

The following tables outline the 17 + 5 categories for innovation initiatives (enablers + outcomes) and describe them in detail. Moreover, the number of occurrences across the full dataset are reported. In total, 837 initiatives were assigned to the innovation initiatives, with the majority of them being categorized as innovation outcomes (477 initiatives = 57% of all initiatives) (Tables 1 and 2).

Since the concrete outcome-oriented innovation initiatives vary widely and depend strongly on the particular fields of practice of the vendors (e.g., BPO vs. ITO), we do not go into more detail here; instead, we focus on the innovation enablers in the following.

Most of the innovation-enabling dimensions focus on vendors' collaboration with external partners and sources, reflecting the strong trend towards open innovation. Moreover, vendors do also implement internal structures and processes to create a substantive innovation capability within their organization. Events, which have a less sustainable impact, were mentioned less although, particularly, the organization of conferences was raised more frequently than the median.[1] When it comes to innovation outcomes, the majority of the outsourcing vendors has its key focus on developing innovative products and approaches for their customers. With 201 and

[1]Median of number of occurrences of innovation enablers across categories is 19; median of number of occurrences of innovation outcomes across categories is 68; median of number of occurrences across all categories is 23.5.

Table 1 Innovation enabler categories

Category	## of occurrences	Description
Collaboration dimension (122 occurrences)		
Collaboration with academia	24	Collaborating with – Students to gain fresh ideas, mindsets and approaches and to discovery new talents – Universities for training programs for employees – Universities to get case studies solved by students – Professors to receive external consulting and learn about different approaches – Research institutes to conduct market analyses to reflect on improvement potentials
Membership in association	9	Being an established member of a cross-company community-of-practice or an association to: – Devise innovative strategies for a specific industry or geography – Solve a problem which will result in benefits for the entire industry or market – Develop regional, national or international standards in their field of expertise
Involvement of customers	23	– Involving clients in the innovation process – Setting up brainstorming sessions and collaborating with clients to develop or improve products and services – Conducting client surveys and meetings to share experiences, best practices, thoughts about improvement opportunities and to discuss industry trends – Creating a platform for clients to give feedback to solutions and propose ideas for new products – Inviting customers to hackathons and research challenges

(continued)

Table 1 (continued)

Category	## of occurrences	Description
Collaboration with key players	54	– Working together with international or branch specific business or technical key players in order to develop a joint product, framework or service – Building alliances with other firms which have the same interests
Collaboration with startups	12	– Collaborating with startups for developing new ideas and solving existing problems – Startups offer the vendor new angles, technological insights and a flexible way of working
People dimension (60 occurrences)		
Design Thinking	8	– Implementing the Design Thinking approach and training own employees in using DT for ideation and generation of innovation
Employee coaching	30	– Developing own employees' personal and professional skills regarding new technologies, software, programming languages and other entrepreneurial approaches excluding Design Thinking – Performing employee coaching through workshops, mentoring, case study trainings and courses performed by internal or external lecturers – Employees participating in startup weeks to nurture the entrepreneurial mindset

(continued)

Table 1 (continued)

Category	## of occurrences	Description
Talent management	22	– Collaborating with universities to set up student programs or boot camps to engage with the most talented people before they enter the job market – Using analytical tools to track employees' skills and project assignments in order to get better insights into the employees' potential – Applying new programs or algorithms to discover and employ highly skilled and motivated people
Structures dimension (107 occurrences)		
Accelerators and incubators	14	– Investing in, partnering with or mentoring accelerator/incubator programs – Developing an own accelerator/incubator program and helping startups to launch their product – Functioning as an incubator for ideas of the customer – Providing clients with R&D services for technology research, new product ideation, proof of concept and prototyping
Centers of Excellence (CoE)	17	– Setting up a CoE to ensure and build up expertise in a specific field, e.g., IoT – CoE supports and provides customized services in their specific field of excellence – CoE enables to reach more market and customer segments by bringing innovation
Innovative firm acquisitions	9	Acquisition of an innovative firm in order to: – Access highly skilled entrepreneurial people and domain expertise – Offer the acquired innovative products or services to the vendor's customers

(continued)

Table 1 (continued)

Category	## of occurrences	Description
Innovation Labs	32	Innovation Lab is – Launched to drive disruption, change the way business is done and to coordinate all pilot activities – Focusing on a specific topic i.e. big data, IoT or blockchain – Designed as an incubation center for new internal ideas, products and services and to create intellectual property – Enables creativity, forward thinking concepts and experimentation and creates an environment for collaboration – Finding promising startups to cooperate with and exploring how innovation affects collaboration – Working on extended testing, piloting and development of prototypes as well as proof of concepts for new ideas within a short timeframe
Innovation platforms for employees	27	– Launching a company-wide platform available for every (vendor-side) employee to share their ideas for new products and services as well as improving existing products or processes – Setting up innovation sessions for brainstorming and pitching of employees’ ideas
R&D programs	8	– Preliminary step of an Innovation Lab – Exploring new technology and frameworks to improve customer satisfaction – Investigating the latest industry trends
Events dimension (71 occurrences)		

(continued)

Table 1 (continued)

Category	## of occurrences	Description
Conferences	38	– Organizing or attending specialized conferences, summits, forums, or technology events and fairs including tech talks and workshops – Receiving a broad overview of current innovations and identifying weaknesses – Networking and peer to peer knowledge exchange for best practices
Hackathons	14	– Hosting or attending events where developers work collaboratively in teams to develop prototypes for a given business challenge. Employees, students, experts or startups might participate
Research challenges	19	– Setting up or participating in research challenges – Advertised either internally for the vendor's employees or externally for students, startups or experts – Goal is to develop an approach or even a prototype for solving a specific business challenge – Winner receives an award and the idea often gets realized collaboratively by vendor and originator
Sum of all occurrences:	360	

Table 2 *Innovation outcome categories*

Category	## of occurrences	Description
Client-related dimension (361 occurrences)		
Development of new approaches for clients	98	– Development of innovative approaches or processes which are then executed by clients and help improve their procedures, adopt new technologies or solve specific problems
Development of new products for clients	201	– Development of new, innovative products for the client. The product idea can be totally new or already existing, however, it has to be implemented in a different way
Innovative service development for clients	62	– Development of innovative activities which are performed by the vendor as service for the client
Internal dimension (116 occurrences)		
Development of new internal approaches	68	– Vendor develops a new, innovative approach or process to solve internal problems, reconfigure existing business processes or to develop new strategies
Development of new products for internal use	48	– Vendor develops new, innovative products to solve internal problems, to improve its efficiency or to offer better and less expensive services to the customer
Sum of all occurrences:	477	

98 assigned datasets, they take up around 24 and 12% of all mentioned innovation initiatives.

Overall, these explorative findings show which innovation initiatives have been implemented by the outsourcing vendors. The question remains how meaningful they are for the vendor's innovativeness and innovation contribution with regard to the client. In order to address this question, we interviewed nine outsourcing experts from three different (client-side) companies. Eight of these experts work as vendor managers in IT outsourcing settings and one in the field of facility management (BPO). Seven are employed by large international companies while two are managing directors at medium-sized companies, managing the outsourcing contracts for their firms. While the value of innovation *outcomes* for the client is both obvious and too context-specific in order to evaluate them in general matters, we focused on let the outsourcing experts rank the 17 categories for innovation-*enabling* initiatives in terms of relevance when selecting a long-term outsourcing vendor.

At the beginning of the individual interviews, the purpose of our study and the 17 categories were explained to the experts. The experts got time to form their own opinion and rank the categories, written on index cards, in their order of priority for the innovativeness of vendors and each enabler's meaningfulness for the eventual innovation benefit of the vendor's clients. After completion, they explained their evaluation results and named the reasons for their specific ranking orders.

Since the experts came up with slightly differing ranking orders, we used average ranks for sorting the categories. Figure 2 shows the results with the categories being sorted by average rank (with 1 representing the highest-possible rank). The categories *Involvement of customers*, *Innovation Labs* and *Membership in association* were rated as the most important while *Collaboration with academia*, *Employee coaching*, and *Innovation platforms for (vendor) employee*s were ranked as least important initiatives by the outsourcing experts.

To receive more detailed insights on the opinions of the outsourcing experts, we also uncovered how many experts evaluated a certain category to be more vs. less important than the other ones. To do so, we put the original ranking positions into relation with the respective median ranking position. Figure 3 shows the results.

This figure discloses the differing opinions of the outsourcing experts. As additional 'meta-result', we found that at least some of the experts seemed not too much to care about what the vendor does, at all, if it does not turn into a direct benefit for the client. Arguments were like "if the vendor applies such initiatives [in general], good innovative services will yield automatically", "innovation outcomes are more important", and "however, the price has to be right", i.e., reflecting a still strong focus on cost effectiveness.

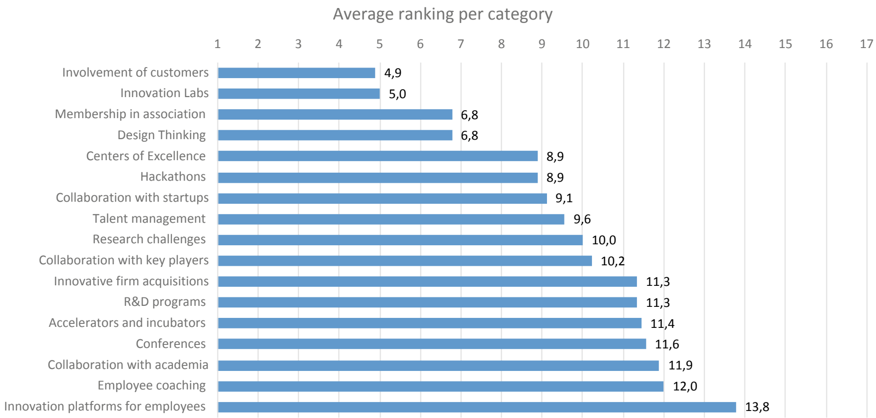

Fig. 2 Average ranking position of innovation initiatives, based on experts' evaluation, scaled from 1 (very important) to 17 (least important)

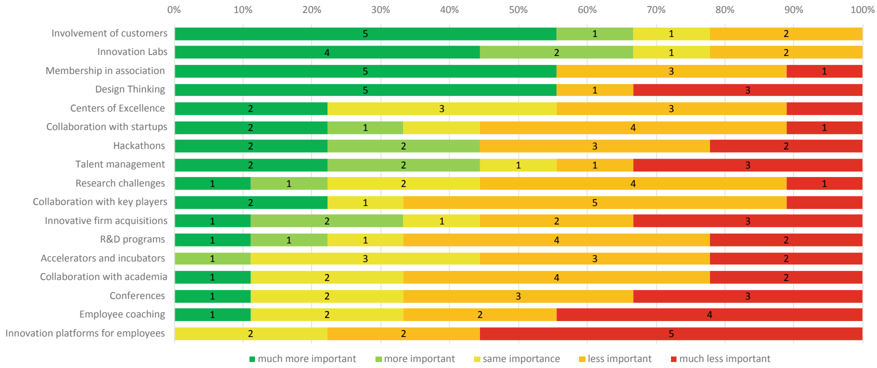

Fig. 3 Individual relative evaluations of importance of innovation initiatives

Among the innovation-enabling initiatives, one of the most consistent opinions was that *Innovation Labs* are a highly effective initiative to drive innovation because they combine a variety of inventive methods, different kinds of collaboration and provide approaches and diversity to develop innovative products. Similarly, collaborating with clients seems to be a powerful open innovation approach and is ranked top-most—which is however not too surprising if asking the client side. Collaboration ensures the development of superior services which fit better to the clients' needs and makes the vendor more stable and innovative resulting in better services for the customer. *Talent management, Employee coaching* and *Involvement of customers* are, in contrast to the rest of the enabler categories, for the majority of the outsourcing experts, basic preconditions for hiring an outsourcing vendor, anyway. In any case, *Employee coaching* and *Talent management* allows involvement of new talents and fresh ideas and it reduces employee turnover, which results in benefits for the clients.

Interestingly, when looking at the variation of the evaluation responses, there is a slight tendency towards more congruence among the answers with regard to structural measures (*Innovation platforms for employees, Accelerators and incubators, and Innovation Labs* were among the five categories with the lowest standard deviation of answers (not shown in the table)) while least congruence appeared for people-related categories (*Talent management* and *Design thinking* among the three categories with highest standard deviation). It seems that outsourcing experts have a better understanding of more 'costly' and permanent approaches (i.e., structural initiatives) no matter whether they believe in rather high (*Innovation Labs*) or low potentials (*Innovation platforms for employees, Accelerators and incubators*).

5 Discussion

All innovation initiatives are provider-initiated measures to drive innovation for the sake of the client. Some initiatives represent outcomes themselves whereas others are enabling approaches that help create an entrepreneurial mindset and prepare the company for innovative approaches.

When evaluating the occurrences of each category, as reported by the outsourcing providers, it can be assumed that the numbers reflect the categories' importance from the vendors' perspective. We can see that the vendors primarily focus on the development of innovative products, processes and services for their customers. This customer-centric view is 'natural' since it reflects the nature of providing outsourcing services, which relies on being contracted by clients. Moreover, enablers can only be reported once (e.g., the setting up of an innovation lab), while outcomes happen more frequently and thus can be reported more often.

When we compare the vendors' perspective (reflected by the frequency of occurrences of each category) with the clients' perspective (reflected by the ranking position of each category as defined by the experts) on the importance of innovation enablers, it becomes apparent that they partially focus on different categories. The clients consider *Involvement of customers* and *Innovation Labs* as very important, while not many vendors have implemented these approaches. Other categories like *Employee coaching* and *Innovation platforms for employees* are the least important for the clients, however, the outsourcing providers deem them relatively important, i.e., use them frequently. When it comes to external collaboration, the two sides favor different partners, too. While the vendors predominantly collaborate with key players and, a bit less, with academia, clients deem those quite unimportant and would rather favor vendors to collaborate with customers and with startups. Similarly, the role of membership in associations is seen differently (low on the vendor side, high on the client side). Of course, one should keep in mind that the vendor-side results are biased by what the vendors perceive to be important from the client side and thus have reported; however, if there is a mismatch in vendors' vs. clients' perception on which vendor activities are most effective for innovation, no matter whether this is only about reporting and signaling or about true innovation capability building activities, this is worthwhile being investigated in more depth in the future.

Contrasting our results with the extant literature, we can first reassure the conclusions of Weeks and Feeny (2008) and Oshri et al. (2015, 2018) that innovation through outsourcing can be achieved. Outsourcing providers establish many initiatives to drive innovation for themselves and their customers. Taking a closer look at the single dimensions of the *Outcome-oriented* dimension, the findings of Weeks and Feeny (2008) regarding the occurrence of the different types of customer innovation cannot be fully supported. They stated that *IT operational innovations* are common innovation outcomes, *business process innovations* are not that often performed *and strategic innovations* do almost never take place. Our data demonstrate that, while *IT operational* and *business process innovations* are performed quite frequently, *strategic innovations* occur several times, too. These deviations do very likely reflect the progress that the outsourcing industry has made during the last years with regard to innovation orientation. Besides, clients could have gained higher confidence in their outsourcing partner and therefore, trust them with assigning more strategic projects to them.

The outcome categories of product, process and service innovation were already established by earlier innovation researchers (Baregheh et al. 2009). We divided these categories into a customer and a vendor view. Baregheh et al. (2009) discovered that product innovation arises most frequently, followed by service and process innovation. This matches partly with our data in which we also found that product innovation is by far the most frequent innovation type, followed by process innovation and, more rarely, service innovation.

Comparing our results with the research of Kotlarsky et al. (2016), the innovation activities described in their categories *innovation assets* and *educational and training assets* were also reported within our data. However, our data provided no evidence for their other categories of *innovation networks* and *specific innovation process approaches*.

While our work is just a first attempt to get insights about the vendor side and vendor perspective on innovation through outsourcing and thus has only preliminary data-driven implications for theory, managerial implications are that our paper can assist companies in the decision making process for finding an adequate outsourcing vendor. Clients get an overview about the wide range of innovation initiatives and can thus make better informed assessments and decisions when searching for a new vendor. In turn, outsourcing vendors make better informed decisions about the importance of the initiatives for their clients and can invest in precisely these initiatives. This again increases the innovation orientation and, thus, the degree of professionalism of the ITO and BPO industry.

Obviously, our research is subject to several limitations. First the data used in the analysis are based on self-disclosure of the vendors. While in most cases, a public source was quoted as reference for validation, the firms might have had strategic reasons to not report other applied innovation initiatives, which might be even more essential. However, since the incentive to completely and truthfully report was very high, we can assume that such a behavior is quite unlikely. Further, our client-side evaluation results are very limited due to the small number of experts involved, the field of expertise and the operating country of the outsourcing experts. Therefore, a next step of our research will be to extend the base and variety of participants representing the client side.

6 Conclusion

As outlined in the beginning, almost no research has focused on innovation through outsourcing from the outsourcing vendor's perspective, so far. Therefore, the goal of this paper was to evaluate what kind of initiatives outsourcing vendors perform to improve their innovativeness and to produce innovation for their clients. Analyzing a dataset of innovation initiatives of globally leading outsourcing vendors, we developed a categorization framework and identified 22 innovation initiatives which vendors perform to enable innovation. 17 categories could be categorized as 'innovation enabling' and were split into four different dimensions: *Collaboration, People, Structures,* and *Events*. The remaining five categories belong to the *Outcome-oriented* dimension and can be further divided into the sub-dimensions *Client-related* and *Internal*.

Subsequently, nine outsourcing experts ranked the innovation-enabling initiatives in regard to the importance for a client looking for a long-lasting outsourcing relationship; *Involvement of customers* and *Innovation Labs* were considered as the most important initiatives.

Although this paper provides some first guidance for outsourcing deciders when evaluating their existing or prospective new vendors, this paper presents only a first step of researching the vendor-side innovation enablers and drivers for an effective and sustainable 'innovation through outsourcing' strategy. Future research needs both to build a sound theoretical basis, linking outsourcing innovation concepts to well-established theories from strategy and innovation management (such as dynamic capabilities view, absorptive capacity etc.), and to strengthen the empirical work by examining relations among vendors' innovation-related activities and both the client's outsourcing success and the financial/market performance of the vendor. Overall, we hope that our research does contribute to the progress of this highly important field of outsourcing research.

References

Aubert, B. A., Kishore, R., & Iriyama, A. (2015). Exploring and managing the "innovation through outsourcing" paradox. *The Journal of Strategic Information Systems, 24*(4), 255–269.

Auerbach, C.F., & Silverstein, L.B. (2003): *Qualitative data: An introduction to coding and analysis. Qualitative studies in psychology*. New York: New York University Press.

Baregheh, A., Sambrook, S., & Rowley, J. (2009). Towards a multidisciplinary definition of innovation. *Management Decision, 47*(8), 1323–1339.

Bettis, R. A., Bradley, S. P., & Hamel, G. (1992). Outsourcing and industrial decline. *The Academy of Management Executive, 6*(1), 7–22.

Boehm, A., Michalik, B., Schmidt, N., & Basten, D. (2014): Innovate on purpose - factors contributing to innovation in IT outsourcing. In *47th Hawaii International Conference on System Sciences (HICSS)*, Waikoloa, USA.

Dayasindhu, N. (2004): Information technology enabled process outsourcing and reengineering: case study of a mortgage bank. In *10th Americas Conference on Information Systems (AMCIS)*, New York, USA.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2004). Information systems outsourcing: A survey and analysis of the literature. *The DATA BASE for Advances in Information Systems, 35*(4), 1–102.

Grover, V., Cheon, M. J., & Teng, J. T. C. (1996). The effect of service quality and partnership on the outsourcing of information systems functions. *Journal of Management Information Systems, 12*(4), 89–116.

Gunday, G., Ulusoy, G., Kilic, K., & Alpkan, L. (2011). Effects of innovation types on firm performance. *International Journal of Production Economics, 133*(2), 662–676.
Hoecht, A., & Trott, P. (2006). Innovation risks of strategic outsourcing. *Technovation, 26*(5–6), 672–681.
Kotlarsky, J., Oshri, I., Lee, J.-N., & Jarvenpaa, S. (2015). Editorial: Understanding strategic innovation in IT and business process outsourcing. *The Journal of Strategic Information Systems, 24*(4), 251–254.
Kotlarsky, J., Rivard, S., & Oshri, I. (2016): On a supplier's paradoxical practices: The case of technological innovations in outsourcing engagements. In *37th International Conference on Information Systems (ICIS)*, Dublin, Ireland.
Lacity, M., & Willcocks, L. (2014). Business process outsourcing and dynamic innovation. *Strategic Outsourcing: An International Journal, 7*(1), 66–92.
Oshri, I., Kotlarsky, J., & Gerbasi, A. (2015). Strategic innovation through outsourcing: The role of relational and contractual governance. *The Journal of Strategic Information Systems, 24*(3), 203–216.
Oshri, I., Arkhipova, D., & Vaia, G. (2018). Exploring the effect of familiarity and advisory services on innovation outcomes in outsourcing settings. *Journal of Information Technology, 24*(4), 255–258.
Teece, D. J. (1987). Capturing value from technological innovation: integration, strategic partnering, and licensing decisions. In B. R. Guile & H. Brooks (Eds.), *Technology and Global Industry: Companies and Nations in the World Economy* (pp. 65–95)., LNCS Washington, D.C.: The National Academies Press.
Tushman, M., & Nadler, D. (1986). Organizing for innovation. *California Management Review, 28*(3), 74–92.
Verwaal, E. (2017). Global outsourcing, explorative innovation and firm financial performance: A knowledge-exchange based perspective. *Journal of World Business, 52*(1), 17–27.
Weeks, M. R., & Feeny, D. (2008). Outsourcing: From cost management to innovation and business value. *California Management Review, 50*(4), 127–146.
Whitley, E. A., & Willcocks, L. P. (2011). Achieving step-change in outsourcing maturity: Toward collaborative innovation. *MIS Quarterly Executive, 10*(3), 95–107.
Zahra, S. A., & Covin, J. G. (1994). The financial implications of fit between competitive strategy and innovation types and sources. *The Journal of High Technology Management Research, 5*(2), 183–211.

Emerging Innovation Ecosystems: The Critical Role of Distributed Innovation Agency

Thomas Hurni, Jens Dibbern, and Thomas L. Huber

Abstract Innovation ecosystems are becoming increasingly important for the co-creation and modification of digital innovation by different and often competing organizational actors. However, how innovation ecosystems emerge between such organizational actors is yet unknown. This article addresses this gap by exploring how central organizational actors create innovation ecosystems, and how and why these innovation ecosystems emerge over time and through the interplay of all involved organizational actors that pursue both common (i.e., cooperate) and own goals (i.e., compete). To answer these questions, we opted for a single-case study of a large software development project, initiated by a major logistics company and implemented in collaboration with its independent IT department, six software vendors, and some field experts. This unique constellation with different coopeting (i.e., simultaneously cooperating and competing) organizational actors is particularly well suited to answer our research questions. Our results show that central organizational actors can create the basic structure and procedures of an innovation ecosystem. However, for an innovation ecosystem to progress in its emergence, central organizational actors need to stabilize the basic structure, while all other organizational actors need to help refine the basic procedures. The better adapted the structure and the procedures, the better organizational actors can exploit them to materialize coherent and customer-oriented digital innovation. We present our findings as a three-phase process model of innovation ecosystem emergence, in which innovation agency is distributed and redistributed among the organizational actors. Our findings have important implications for the literature on innovation ecosystems, the coopetition paradox, and digital innovation.

T. Hurni (✉) · J. Dibbern
University of Bern, Bern, Switzerland
e-mail: thomas.hurni@iwi.unibe.ch

J. Dibbern
e-mail: jens.dibbern@iwi.unibe.ch

T. L. Huber
ESSEC Business School, Paris, France
e-mail: huber@essec.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_6

1 Introduction

Established companies are increasingly failing to keep up with disruptive digital innovation. To yet survive, more and more companies join forces with other organizational actors in innovation ecosystems, using their joint innovative power to co-create and modify digital innovations (Furr and Shipilov 2018). We define innovation ecosystems as the alignment structure of a multilateral group of organizational actors that must cooperate for materializing a coherent and customer-oriented digital innovation (Adner 2006, 2017). To exploit the joint innovative power of multiple organizational actors, an innovation ecosystem must first emerge. However, the underlying emergence process of innovation ecosystems is mostly unknown.

So far, research on innovation ecosystems has contributed valuable insights into how such ecosystems are orchestrated by central organizational actors (e.g., Giudici et al. 2017), or how organizational actors balance their competitive and cooperative interests (e.g., Davis 2016; Hannah and Eisenhardt 2018) as well as develop and negotiate their identities over time (e.g., Lindgren et al. 2015). At the same time, little is known about the emergence of innovation ecosystems and the few studies that have investigated this important issue, have done that from the particular perspective of central organizational actors (e.g., Dattée et al. 2018; Giudici et al. 2017). Conclusions about why and how innovation ecosystems emerge were therefore extrapolated only from the perspective of these central organizational actors, neglecting the perspective of all peripheral organizational actors that are essential for understanding how and why innovation ecosystems progress in their emergence (Lumineau and Oliveira 2018). This one-sided view on the emergence of innovation ecosystems does not do justice to their actual complexity for two particular reasons. First, organizational actors in innovation ecosystems not always pursue the same goals. While some central organizational actors undoubtedly seek to create and orchestrate innovation ecosystems (Dattée et al. 2018; Giudici et al. 2017; Paquin and Howard-Grenville 2013), other peripheral organizational actors pursue common (i.e., cooperate) and own (i.e., compete) goals at the same or various times (Hannah and Eisenhardt 2018). Second, organizational actors in innovation ecosystems are supposed to contribute to the materialization of a coherent and customer-oriented digital innovation. The distribution and redistribution of innovation agency among those organizational actors is, therefore, an essential characteristic of innovation ecosystems, which can only be understood if the different goals, motives, and abilities of all relevant actors are acknowledged (Nambisan et al. 2017). Focusing on just one type of organizational actor is, therefore, not sufficient to understand how and why innovation ecosystems emerge over time and between all involved organizational actors that can pursue both common and own goals. To make the best possible use of existing innovation agency, it is imperative to understand how and why innovation ecosystems progress to emerge between such different types of organizational actors. We, therefore, ask *how central organizational actors create innovation ecosystems and how and why such innovation ecosystems progress in their emergence over time and through the* *interplay of all involved organizational actors that pursue both common and own goals.*

To answer this question, we opted for a single-case study (Yin 2009) of a large software development project, initiated by a major logistics company and implemented in collaboration with its independent IT department, six software vendors, and some field experts. This unique constellation is particularly well suited for understanding how a central organizational actor creates an innovation ecosystem, and why and how such an innovation ecosystem emerges over time and through the interplay of all involved organizational actors. More specifically, this setting allows understanding why and how an innovation ecosystem emerges around a desired coherent and customer-oriented digital innovation in a highly coopetitive environment (i.e., a simultaneously competitive and cooperative environment, see Bengtsson and Kock 2014) with distributed innovation agency. Our results show that a central organizational actor can create the basic structure and procedures of an innovation ecosystem. However, for an innovation ecosystem to progress in its emergence, central organizational actors need to stabilize the basic structure, while all other organizational actors need to help refine the basic procedures. The better adapted the structure and the procedures, the better organizational actors can exploit them to materialize coherent and customer-oriented digital innovation. We present our findings as a three-phase process model of innovation ecosystem emergence, during which innovation agency is distributed and redistributed among central and peripheral organizational actors. The contributions of our study are threefold. First, we contribute to the literature on innovation ecosystems, as we find that innovation ecosystems emerge in three different phases through the creation, adaptation and exploitation of the structure and procedures. Second, we contribute to the literature on the competition and cooperation paradox, as we reveal that only a coexistence of common (i.e., cooperate) and own (i.e., compete) goals promote the emergence of innovation ecosystems. Third, we contribute to the literature on digital innovation, as we highlight the importance of distributing and redistributing innovation agency among organizational actors for the emergence of an innovation ecosystem and the materialization of coherent and customer-oriented digital innovation.

The structure of this article is as follows. First, we shed light on innovation ecosystems, the coopetition paradox, and digital innovation. Second, we provide detailed information about the chosen method, including information about how we collected and analyzed our data, as well as how we derived our process model. Third, we illustrate the context of our case before we present the phases, and integrate them in a three-phase process model about how central organizational actors create innovation ecosystems and how and why such innovation ecosystems emerge over time and through the interplay of all involved organizational actors that pursue both common and own goals. We end with a discussion, where we elaborate on the theoretical contributions, the implications for practice, and promising paths for future research.

2 Background and Conceptual Foundations

2.1 Innovation Ecosystems

Innovation ecosystems describe *the alignment structure of a multilateral group of organizational actors that need to cooperate for coherent and customer-oriented digital innovation to materialize* (Adner 2006, 2017; Adner and Kapoor 2010). Despite their apparent similarity, innovation ecosystems are significantly different from other forms of inter-organizational collaboration (Adner 2017). From platform ecosystems (e.g., Gawer and Cusumano 2002; Tiwana et al. 2010), multisided markets (e.g., Boudreau and Haigu 2009), and buyer-supplier relations (e.g., Porter 1985), for example, innovation ecosystems differ in that they neither rely on one-to-one nor on one-to-many, but on many-to-many relations (Adner 2017). Moreover, innovation ecosystems differ from alliances and networks (e.g., Gulati 1998; Powell et al. 1996), in that they intend to materialize a coherent and customer-oriented digital innovation (Adner 2017).

Innovation ecosystems were first mentioned in practitioner-oriented management literature in the mid-1990s (Moore 1993), and have since become a key concept for IS, management, and organizational scholars (Autio and Thomas 2014). Especially in recent years, this growing importance has enhanced our understanding of innovation ecosystems, such as how organizational actors reconcile their competitive and cooperative interests (e.g., Davis 2016; Hannah and Eisenhardt 2018), and how they develop and negotiate their identities over time (e.g., Lindgren et al. 2015). Innovation ecosystems, however, do not come out of thin air but rather emerge over time and based on the initiative of one or more central organizational actors. We define the emergence of innovation ecosystems as the progressive formation of an alignment structure in which a multilateral group of organizational actors can co-create and modify digital innovation that are only possible in collaboration. Unfortunately, there is little research into the creation and emergence of innovation ecosystems. One notable exception is Dattée et al. (2018) that shows how central organizational actors (i.e., organizational actors that deliberately seek to create innovation ecosystems around their organizations) compel other organizational actors to commit to a creation effort in situations where uncertainty is high. Closely related is Giudici et al. (2017) that focuses on the orchestration of innovation ecosystems by ‘other’ organizational actors, including business incubators and venture associations. A third example is Paquin and Howard-Grenville (2013) that shows how central organizational actors move from ‘blind dating’ other organizational actors toward ‘arranged marriages’ among them. Both studies have in common that they focus on one specific type of organizational actor—more specifically, on central organizational actors that strive to create an innovation ecosystem—from which conclusions are drawn (Lumineau and Oliveira 2018). However, this focus on one particular type of organization actor does not do justice to the complexity of innovation ecosystems for two reasons. First, although all organizational actors in innovation ecosystems should pursue common goals, they often pursue their own. Thus, while some organizational actors undoubtedly seek to create and orchestrate innovation ecosystems (e.g., Dattée et al. 2018; Giudici et al. 2017; Paquin and Howard-Grenville 2013), others are likely to pursue both common (i.e., cooperate) and own goals (i.e., compete) at the same or different times (Hannah and Eisenhardt 2018). Second, although digital technologies facilitate the coordination in innovation ecosystems (Adner 2006), they also allow a distribution of innovation agency among organizational actors with distinct objectives, motives, and capabilities, which further increases the coordination effort (Nambisan et al. 2017). So far, it is largely unknown how and why this complexity created by cooperating and competing organizational actors with distributed innovation agency influences the creation and emergence of innovation ecosystems. Before we address this lack of knowledge, we first discuss the two underlying causes in more detail and introduce a possible approach for tackling the complexity.

2.2 Coopeting Innovation Agents

Although organizational actors in innovation ecosystems need to cooperate for the materialization of coherent and customer-oriented digital innovation, they may also pursue own goals or even compete with one another (Davis 2016; Hannah and Eisenhardt 2018). Cooperation in this context describes the process by which individuals or organizational actors work with each other for the mutual benefit, while competition describes the process by which individuals or organizational actors rival each other for the purpose of selfish benefit (Bengtsson and Kock 2000). As such, cooperation and competition are often considered as two poles of the same continuum (Tjosvold and Choy 1994), meaning that the more individual or organizational actors compete with each other, the less they cooperate and vice versa. This perception contradicts, however, the paradox nature attributed to coopetition, as paradoxes denote the persistent contradiction between independent elements (Schad et al. 2016) that "seem logical when considered in isolation but irrational, inconsistent, and even absurd when juxtaposed" (Smith and Lewis 2011, p. 386). Thus, cooperation and competition are not the ends of the same continuum but their own continuums (Bengtsson and Kock 2014), where coopetition only exists if individuals or organizational actors both cooperate and compete (Luo 2007).

Coopetitive relations, as found in innovation ecosystems, do not necessarily thrive. Indeed, about 50% of all coopetitive relations fail (Lunnan and Haugland 2008; Park and Ungson 2001). Reasons for these failures are the dynamics of the underlying process (Pathakn et al. 2014; Williamson and de Meyer 2012) with actors that simultaneously pursue own and common benefits (Khanna et al. 1998), share and protect knowledge (Ho and Ganesan 2013), and learn from each other (Kale et al. 2000). The dynamics of the underlying processes, therefore, pose serious threats to the emergence of innovation ecosystems. Understanding how and why these dynamics affect the emergence of innovation ecosystems is therefore crucial and requires a closer look.

2.3 Distributed Innovation Agency

At the heart of innovation ecosystems is the materialization of coherent and customer-oriented innovations by multilateral groups of organizational actors (Adner 2017). As these innovations are typically created with the help of digital technologies and are mostly digital technologies themselves (Adner 2006; Adner and Kapoor 2010; Mantovani and Ruiz-Aliseda 2016), we regard them as digital innovations (Nambisan et al. 2017). Digital innovation differs in at least two respects from traditional innovation, which describe the invention, development, and implementation of new ideas or solutions to specific problems (Garud et al. 2013). First, in digital innovation, digital technologies facilitate the distribution of innovation agency (Nambisan et al. 2017). In other words, digital technologies allow the involvement of far more innovation agents with own objectives, motives, and capabilities than in traditional settings, known from research and development departments with predefined innovation agents. Although such a distribution of innovation agency to a large number of organizational actors is appealing, it also increases the coordination effort throughout the innovation process. Second, in digital innovation, the innovation processes are increasingly blurring with the underlying innovation outcomes, which is especially true if the innovation outcomes themselves are digital (Nambisan et al. 2017). The reason for the increasingly blurring boundaries between the innovation processes and outcomes lies in the uniqueness of digital technologies, with regard to their malleability, editability, or transferability (Yoo et al. 2010). This uniqueness of digital technologies allows continuous improvements of the innovation outcome during and even beyond the innovation process (Lyytinen et al. 2016). Although appealing, this increasingly blurred boundary between the innovation processes and outcomes increases the coordination effort. Thus, despite the obvious benefits of digital innovation, the increased coordination effort could jeopardize the emergence of innovation ecosystems. Understanding how and why the characteristics of digital innovation influence the emergence of innovation ecosystems is, therefore, crucial and requires a closer look.

2.4 Artifact Centered Orchestration

Innovation ecosystems are inherently complex in view of the coopetitive relations between the organizational actors, the distributed innovation agency, and the increasingly blurred boundaries between the innovation processes and outcomes. For innovation ecosystems to progress in their emergence, and for materializing coherent and customer-oriented digital innovations, overcoming this complexity is essential. In this regard, previous research on innovation networks has proposed the concept of orchestration with one or few organizational actors taking responsibility for coordinating the value co-creation (Dhanaraj and Parkhe 2006; Nambisan and Sawhney 2011), or matching solutions with problems so that innovations can be materialized

(Nambisan et al. 2017). However, unlike other forms of inter-organizational collaboration, the many-to-many relations make innovation ecosystems inherently more complex (Adner 2017). Although this complexity allows for the initial creation and orchestration of innovation ecosystems by central organizational actors (e.g., Dattée et al. 2018; Giudici et al. 2017; Paquin and Howard-Grenville 2013), their orchestration becomes more difficult as they emerge. This increased complicity indicates the need for further orchestration entities. Ironically, the increasingly blurred boundaries between the innovation processes and outcomes could play a significant role in this regard (Nambisan et al. 2017).

Digital technologies are increasingly blurring the boundaries between innovation processes and outcomes, which is especially true if the innovation outcomes themselves are digital (Nambisan et al. 2017). This increased blurring can be explained by the peculiarities of digital technologies, such as malleability, editability, or transferability (Yoo et al. 2010), which allow the continuous improvement of innovation outcomes during and even beyond the innovation process (Lyytinen et al. 2016). While innovation ecosystems aim to materialize such continuously improving innovation outcomes, they strive for both coherence and customer-orientation. For this purpose, an orchestration of the multilateral groups of organizational actors in innovation ecosystems is essential (Adner 2006, 2017; Adner and Kapoor 2010). As previously noted, central organizational actors can create innovation ecosystems and thus orchestrate them initially (e.g., Dattée et al. 2018; Giudici et al. 2017; Paquin and Howard-Grenville 2013). However, this orchestration by central organizational actors is not enough to obtain coherent and customer-oriented digital innovations, as innovation ecosystems progress in their emergence. This need for more orchestration entities could be countered by technology artifacts that allow matching solutions to problems and joint sensemaking during the innovation process (Nambisan et al. 2017). In the following, we refer to these orchestration entities as common innovation artifacts. Examples for common innovation artifacts are design guidelines, standardized development methods, or shared infrastructures that help materialize coherent and customer-oriented digital innovations. The better these common innovation artifacts complement the organizational actors as orchestrating entities for matching solutions to problems and joint sensemaking, the more progressed an emerging innovation ecosystem is.

3 Method

We conducted a longitudinal single-case study (Yin 2009) about the unique software development project REMO (short for Reorientation Mobile Computing) at the major logistics firm LogCH. This unique case enabled us to understand how central organizational actors create innovation ecosystems and how and why such innovation ecosystems progress in their emergence over time and through the interplay of all involved organizational actors that pursue both common and own goals. Project REMO is particularly well suited to answer these questions, as it pursued the goal of materializing a coherent and customer-oriented digital innovation through the ongoing collaboration of one client (LogCH), its independent IT department (IT LogCH), six competing software vendors with nearly identical capabilities, and a group of field experts.

To ensure that LogCH supports the conduction of a longitudinal study, we first turned to the member of the executive board responsible for project REMO. This member of the executive board then brought us in contact with the responsible project manager. Based on both their assured support, we began purposefully sampling the involved organizational actors. Specifically, we focused on nine different organizational actors, namely the project team (LogCH), its independent IT department (IT LogCH), the six competing software vendors, and the group of field experts. Table 1 provides an overview of these nine organizational actors and the 33 interviewed individuals. Every organizational actor was actively contributing throughout project REMO and part of the emerging innovation ecosystem.

3.1 Data Collection

For triangulation (Eisenhardt 1989; Yin 2009), we collected three types of qualitative data, namely semi-structured interviews, archival data, and observational data. We initiated data collection in November 2014 after the responsible member of the executive board and the project manager assured us to support the conduction of a longitudinal study. Following an informal interview with the project manager, we purposefully sampled (Yin 2011) interview partners from all nine involved organizational actors in project REMO. Following each purposefully sampled interview, we requested the respective interviewee to name additional interview partners worthwhile for our investigation in the sense of a snowball sampling (Yin 2011). The resulting 34 semi-structured interviews[1] lasted 1.5 h on average, with a range from 55 min to 2 h and 15 min. Every interview was conducted in accordance with the recommendations of Myers and Newman (2007) for qualitative interviews and—except for one Skype interview with a shored employee in Germany—conducted on site in the interviewees' native language (i.e., German or Swiss German). Each interview was tape recorded and transcribed immediately after the conduction. We supplemented our interview data with archival data, including project documentations that provided us with rich insights into the overall project, the initial requirements, or involved key personnel, and presentations about the project that provided us with rich insights about the final state. Eventually, we observed a scrum meeting to understand how the software vendors, LogCH, IT LogCH, and the field experts interacted in the predefined meetings.

[1]The project manager was interviewed twice.

Table 1 Studied organizational actors and interviewed individuals

Organizational actor	Description	Interviewees
LogCH	LogCH is a major logistics company. Three of its business units commissioned project REMO. Division A took over the project management and appointed a scrum expert and three product owners, each responsible for two software vendors	• Project manager[a] • Product owner 1 • Product owner 2 • Product owner 3 • Scrum expert
IT LogCH	IT LogCH is the independent IT division at LogCH. During project REMO, IT LogCH was responsible for providing the software development infrastructure and the framework that acted as an intermediary layer between the applications and the software platform	• Team leader • Scrum master • Lead developer
Field Experts	The field experts were users of the old system or very experienced employees of the three divisions who knew their way around with the old system. They took on the roles of requestors, testers and controllers	• Field expert 1 (division A) • Field expert 2 (division A) • Field expert 3 (division B) • Field expert 4 (division C)
Vendor 1	Vendor 1 is a large international software vendor, assigned in the first round of software vendors. During project REMO, product owner 1 supervised vendor 1. Vendor 1 partially developed 2 applications	• Swiss CEO • Scrum master
Vendor 2	Vendor 2 is a large Swiss software vendor, assigned in the first round of software vendors. During project REMO, product owner 2 supervised vendor 2. Vendor 2 developed 4 applications	• Key account manager • Architect • Scrum master • Business analyst 1 • Business analyst 2
Vendor 3	Vendor 3 is a large Swiss software vendor, assigned in the first round of software vendors. During project REMO, product owner 3 supervised vendor 3. Vendor 3 developed 5 application and partially developed another application	• Scrum master 1 • Scrum master 2 • Scrum master 3 • Business analyst • Developer 1 • Developer 2 • Developer 3

(continued)

Table 1 (continued)

Organizational actor	Description	Interviewees
Vendor 4	Vendor 4 is a large Swiss software vendor, assigned in the second round of software vendors. During project REMO, product owner 1 supervised vendor 4. Vendor 4 developed 2 applications and partially developed another application	• Scrum master • Business analyst
Vendor 5	Vendor 5 is a large international software vendor, assigned in the second round of software vendors. During project REMO, product owner 2 supervised vendor 5. Vendor 5 developed 5 applications	• Key account manager • Scrum master • Business analyst • Developer
Vendor 6	Vendor 6 is large international software vendor, assigned in the second round of software vendors. During project REMO, product owner 3 supervised vendor 6. Vendor 6 developed 2 applications	• Scrum master

[a]We interviewed the Project Manager twice

3.2 Data Analysis

We began with continuous data analysis following the first informal interview with the project manager in November 2014. This continuous data analysis allowed us to react early on to new insights and to adapt our semi-structured interview guideline accordingly. To analyze the collected data, we used the NVivo 11 software solution and followed the recommendations by Charmaz (2006) for an iterative coding procedure. In an initial step, we coded each piece of data line-by-line, using process codes and descriptive sub codes. Process codes rely on gerunds to connote observable and conceptual action in the data (Miles et al. 2013). Gerunds ('-ing' words) are particularly well suited for initial coding, as they curb human tendencies to make conceptual leaps and to adopt extant theories before a necessary analysis (Charmaz 2006). Descriptive sub codes are second-order tags assigned to a primary code—in our case process codes—to enrich their significance (Miles et al. 2013). This line-by-line coding procedure offered two distinct advantages. First, line-by-line coding allowed us to identify both beneficial and obstructive events during the emergence of the innovation ecosystem, and second, to order these events chronologically. The identification of both beneficial and obstructive events and their chronological order gave us a first holistic picture of how LogCH, as the central organizational actor, initially created the innovation ecosystem and how it progressed in its emergence over time and through the interplay of all involved organizational actors.

To grasp whether the innovation ecosystem progressed in its emergence or not, we followed the notion of orchestration. Orchestration is performed by orchestrating entities and manifests in a more or less effective matching of solutions to problems as well as a more or less facilitated joint sensemaking (Nambisan et al. 2017). Thus, the more effective orchestration entities match solutions to problems and the better they facilitate joint sensemaking, the more progressed the innovation ecosystem emergence. Accordingly, we have coded the emergence of ecosystems as progressive when the orchestration entities have facilitated both the matching of solutions to problems and joint sensemaking. In this regard, we not only considered the orchestrating roles of the involved organizational actors but also paid particular attention to common innovation artifacts. This focus on both organizational actors and common innovation artifacts as orchestrating entities is particularly well suited for our study, given the complexity of innovation ecosystems with coopeting organizational actors and distributed innovation agency.

In the next step, we proceeded with a more focused coding of our data. For this purpose, we decided about which of our initial codes make the most analytic sense to categorize our data incisively and completely (Charmaz 2006). We then focused on exploring relations in our codes via axial coding, with the goal to identify general patterns. For this purpose, we systematically compared the dynamics within our case using replication logic, memo writing and tables (Miles et al. 2013). This resulted in three major phases that explain how the central organizational actor initially created the innovation ecosystem and how the innovation ecosystem emerged over time and through the interplay of all involved organizational actors, that pursued both common and own goals. We finalized our analysis by theoretically coding our data. This final step allowed us to identify the theoretical mechanisms underlying the three identified phases (Charmaz 2006). By synthesizing and abstracting these findings, we constructed our final process model of innovation ecosystem emergence that explains how central organizational actors create innovation ecosystems and how and why innovation ecosystems emerge over time and through the interplay of all involved organizational actors that pursue both common and own goals.

4 Results—Creation and Emergence of an Innovation Ecosystem

The initial trigger for the creation and the emergence of the analyzed innovation ecosystem was the approaching end-of-life of the mobile computing devices used by around 20,000 LogCH employees during their day-to-day tasks of receiving, processing, transporting, and distributing deliveries. Having a long tradition of supporting its employees with such devices (Fig. 1 illustrates the prior devices), LogCH introduced the most recent device in the same year as Apple released its first iPhone (Block 2007). Despite this long tradition, LogCH was unable to foresee the revolutionary developments triggered by the release of this first mainstream multi-touch

Fig. 1 Past mobile computing devices at LogCH

smartphone. For example, even though transmission technology was rapidly evolving, LogCH stuck with an outdated standard that was unsuitable for contemporary business applications, geolocation, or encrypted payments. The core of the problem, however, was not the rapid evolution, but the monolithic system architecture that made it impossible to replace the hardware while maintaining the software and vice versa. Against this backdrop, LogCH decided to revolutionize its mobile computing strategy.

4.1 Phase 1—Creating Basic Structure and Procedures

In 2012, LogCH launched project REMO to replace the dated mobile computing devices and their monolithic system architecture. The stated goal was the materialization of an innovative, coherent, yet flexible system with strictly modular components (i.e., hardware, software platform, framework, and features) and features (i.e., applications), as has already been the case with contemporary consumer smartphones. To achieve this goal, LogCH invited four consulting companies to leverage their expertise on system architectures. From the resulting proposals, LogCH opted for a modular cross-compiler architecture with independent components and features. This system architecture was the first common innovation artifact defined to set initial procedures—i.e., it specified how the individual components are divided and related.

The modular cross-compiler architecture required a framework as an intermediary layer between the applications and the software platform, as well as multiple applications to support both the employees in their fieldwork and the management in its executive function. However, LogCH alone lacked the expertise to develop such a framework and to specify the applications. For the development of the framework, LogCH therefore decided to distribute innovation agency to its independent IT department, IT LogCH. For the definition of the applications, LogCH carried out a business process redesign, where selected employees of the project team and from the field (i.e., field experts) obtained innovation agency and gathered the business processes in use, to align them with those implemented in the previous systems. This business process redesign allowed LogCH to fathom a target state for defining twenty applications:

> We [LogCH] conducted a business process redesign, where we, or rather the field experts, had a look at the existing processes and gathered the processes in use. We moderated it and abstracted the process model in a spreadsheet, with each process briefly described in 1-2 sentences, in terms of its function, the potential for improvement and critical weaknesses. This procedure eventually resulted in a target process model, which was broken down into several applications. (LogCH – Product Owner 2)

The distribution of innovation agency to IT LogCH and the field experts for developing the framework and defining applications had two implications. First, by distributing innovation agency to IT LogCH and the field experts, Log CH shaped the structure of the innovation ecosystem in terms of the entered cooperation with them. Second, by developing the framework, IT LogCH defined a common innovation artifact that set additional procedures—i.e., it specified interfaces to the components and features.

The fast-approaching end-of-life of the previous system further complicated the situation for LogCH. Still, LogCH endeavored a highly innovative and high-quality system that stays within budget and complies with the World Trade Organization rules for public tenders. For this purpose, LogCH decided to divide project REMO into subprojects. One of these subprojects explicitly dealt with the application development by six selected software vendors. Thus, LogCH distributed innovation agency to six software vendors, each with the same capabilities to design, build, test, deploy, and run the twenty previously defined applications:

> An important success factor was the tender. They [LogCH] have not tendered requirements, but software vendor skills. […] Because of that, the selection of software vendors was significantly better than if they would have taken just the cheapest ones. We can assume that they [the assigned software vendors] were the best. (Scrum Expert)

Distributing innovation agency to six software vendors for developing the twenty applications had paradoxical consequences for the structure of the innovation ecosystem. On the one hand, their involvement increased the need for cooperation in obtaining the desired, coherent system, and on the other hand, their equal capabilities built the basis for competition.

To obtain twenty coherent applications despite this paradoxical coopetitive setting, LogCH needed to take additional measures. One of these measures was the default agile scrum development method, which LogCH hoped would help to detect problems, dissatisfactions, and errors early on. Besides, LogCH considered it somewhat unrealistic to specify all requirements beforehand. However, it quickly became evident that the general understanding of the agile scrum development method was somehow inconsistent and partially incomplete:

> IT LogCH and the software vendors have confirmed to us: "Yes, we do know scrum!" On closer examination, however, it turned out that it was not scrum or that they simply had a different understanding of scrum than we did. (LogCH - Project Manager T1)

To avoid potential issues arising from these inconsistent and partially incomplete understandings, LogCH hired an external scrum expert. This scrum expert had to define an agile scaled scrum development method and enforce it among all involved organizational actors. The resulting method required all software vendors and IT LogCH to develop in bi-weekly sprints and to attend meetings on a team-, functional-, and project level. On a team level, the software vendors and IT LogCH had to gather with their product owners and the field experts for a retrospective of the past sprint, a sprint review, and a sprint planning for the following two weeks. On a functional level, the business analysts, architects, scrum masters, and quality managers had to meet with their peers in alignment meetings. On a project level, the software vendors had to present their development increments to the other software vendors, LogCH, IT LogCH, and the field experts. By defining and enforcing this agile development method, the scrum expert defined various common innovation artifacts that set additional procedures—i.e., it specified how the applications had to be developed. At the same time, the definition of these common innovation artifacts increased the transparency between all organizational actors, thereby further shaping the coopetitive setting.

Defining a development method with three different meetings every other week further increased the coordination effort for all organizational actors. LogCH aimed to reduce this coordination effort and shape the coopetitive setting between the organizational actors by providing the needed localities and infrastructures:

> We wanted the software vendors to work on our platforms, which means that the documentation is with us, development is with us, and testing is with us. […] We also set up a project office near our headquarters and asked the software vendors to be there with their key roles. (LogCH - Project Manager)

More specifically, LogCH rented a floor in an empty business complex close to its headquarters, with individual offices for the organizational actors, and shared areas for informal interactions and the scheduled meetings. Besides, LogCH provided a standard technology stack for the application design, development, and testing and made the code and documentation repositories accessible to all organizational actors for complete transparency. By providing the needed localities and infrastructures, LogCH defined additional common innovation artifacts that set additional procedures—i.e., it specified by which means the applications had to be developed. At the same time, defining these common innovation artifacts aimed to shape the coopetitive setting among all organizational actors.

Although LogCH has defined multiple common innovation artifacts for materializing the desired, coherent innovation outcome, it chose to ensure the coherence of the desired system even better through guidelines: *"One has chosen a technocratic* *approach with guidelines and templates." (LogCH—Product Owner 3).* More specifically, LogCH stipulated architecture, coding, documentation, design, usability, and testing guidelines. In doing so, LogCH defined several common innovation artifacts that shaped meaningful cooperation through clear rules and regulations despite the coopetitive environment.

Based on the created basic structure and procedures, LogCH assumed it had created a functioning innovation ecosystem: *"From the beginning, it was communicated that it should become an [innovation] ecosystem." (Vendor 5—Developer).* However, the innovation ecosystem was still in its infancy and not far progressed in its emergence.

4.2 Phase 2a—Refining Basic Procedures

The involvement of six software vendors after two consecutive kick-off meetings significantly increased the scope of and the coordination effort within the emerging innovation ecosystem. Conducting two consecutive kick-off meetings instead of one became necessary as LogCH initially intended to work with only three of the six software vendors, but shortly after that added the other three for mastering the tasks on time. The aim of the meetings, however, remained the same: to prepare all organizational actors and to involve them actively in order to exploit their expertise, thus to take advantage of the distributed innovation agency:

> In the first part of the kick-off meeting, we provided some background information – how do we intend to proceed, what was the past, and what do we want to do in the future? To convey our vision and emphasize the already made progress. […] In the second part, we conducted various workshops. There we asked them to get involved: "Now that we have provided you with some background information, do you have questions? Do you have any suggestions regarding the architecture? Do you have ideas regarding the test procedure?" etcetera. (Scrum Expert)

Based on their expertise, all organizational actors critically eyed the defined common innovation artifacts. Particularly the six software vendors quickly identified overlooked deficiencies that hindered them in materializing the desired system: *"Well, I have already seen many agile scrum development projects. For us, or at least for me, that was nothing new. My experience certainly allowed me to emphasize things like "this is a definition of ready that will not work in this project—it simply lacks detail."" (Vendors 2—Scrum Master).* However, the distributed innovation agency not only enabled the software vendors to detect deficiencies, but also to propose innovative solutions to remedy them and refine the common innovation artifacts. Vendor 3 was particularly fast and active in this regard: *"Vendor 3, that is something we have noticed, has taken a different approach to the project than we did. That was obvious from the beginning. For example, while we entered into this project and accepted the way LogCH wanted us to work, vendor 3 joined and wanted to work in its own way." (Vendor 2—Scrum Master).* One of the many examples where vendor 3 proposed an innovative solution, related to the guidelines for structuring and testing the use cases:

> We had a certain influence there, and one or the other of my feedbacks on structuring and testing the use cases was adopted and set as standards for all vendors. That was not because we were incredibly innovative, but because we were fast. It is likely that I was the first from whom they [LogCH] received feedback. (Vendor 3 – Scrum Master 2)

Its expertise and speed enabled vendor 3 to help remedy various deficiencies of the defined common innovation artifacts. Its solution for refining the guidelines for structuring and testing use cases, for example, helped to guarantee meaningful cooperation through clear rules and regulations. For vendor 3, doing so had paradox consequences. On the one hand, supporting all organizational actors to materialize the desired coherent system with a refined common innovation artifact was cooperative in itself. On the other hand, enforcing an innovative solution while all other organizational actors had to follow suit was competitive. Since the other organizational actors had innovative solutions to propose themselves, this competitive aspect was particularly problematic:

> Vendor 3 had a certain pioneering role, this certainly because they were among the first to participate. They have suggested some of their procedures early on, which made it difficult to change them – this caused some disagreements. (Vendor 6 – Scrum Master)

For the progress of the innovation ecosystem emergence, such acts had two consequences. First, distributed innovation agency enabled identifying deficiencies in common innovation artifacts earlier and proposing innovative solutions to refine them faster. For all organizational actors, refined common innovation artifacts meant a further homogenization of the basic procedures, which also shaped the coopetitive setting. Second, the fact that most organizational actors themselves had innovative solutions to the same deficiencies steadily increased competition for proposing innovative solutions:

> We had to get involved and show that we are here now. That is why we have attached a poster saying 'deliver or die', which is one of the many sayings at vendor 6. True to this motto – we delivered and performed after a certain ramp-up. Therefore, the other companies have also realized that vendor 6 is here and is ready to work. It is like a new student coming into an existing class. You have to prove yourself and show that you can do something. (Vendor 6 – Scrum Master)

Whether proposed solutions have led to refined common innovation artifacts or not, however, built on LogCH's decision. The example of the design guidelines shows that this was not always the case—although the organizational actors identified deficiencies and proposed innovative solutions for refinements, LogCH decided to keep them unrefined: *"In terms of user interaction design, LogCH had no need… or wanted to have no need." (Vendor 3—Scrum Master 1).*

Even though many organizational actors proposed innovative solutions to the same deficiencies, often only one organizational actor had the necessary expertise in a particular domain. One organizational actor that repeatedly used its superior expertise to propose innovative solutions for refining common innovation artifacts was vendor 4—the first time already shortly after the second kick-off meeting. At that time, vendor 4 noticed that the internet in the shared office space was not working correctly. Access to the internet and the networks was, however, essential for the development of the applications:

> The infrastructure was always an issue, especially in the beginning. We had to work with our hardware, which was a requirement by LogCH, but because they did not allow us access to their network, we had to deal with a malfunctioning guest network. That meant that we had to work with the guest Wi-Fi, which was slow and half the time did not work, making it impossible for us to develop. (Vendor 2 – Scrum Master)

Due to its informal behavior, vendor 4 quickly discovered that the malfunctioning internet also hindered the other organizational actors: *"In the beginning, we were the only ones who always had an open door—such that at least sometimes one dared to look inside. We also stood around the coffee machine, as we were encouraged to exchange information during informal 'coffee talks'." (Vendor 4—Business Analyst).* Since vendor 4, unlike the other organizational actors, already had some experience in providing internet access, it remedied this deficiency: *"They somehow got the internet up and running." (Vendor 3—Scrum Master 1).* In doing so, vendor 4 refined a common innovation artifact, namely the shared office space. For vendor 4, refining this common innovation artifact had paradox consequences. On the one hand, it was cooperative in itself and supported all organizational actors in materializing the desired coherent system. On the other hand, it gave vendor 4 a competitive advantage over all other organizational actors.

A short time later, the flawed automated test infrastructure hindered all organizational actors. This mainly since testing was a vital deliverable to finish a sprint: *"One had imagined the testing differently. We had to redefine it during the project—at the latest when we realized that we needed a higher level of automated testing for the intended development pace. Testing was one part of every sprint—a deliverable." (LogCH—Product Owner 3).* As with the malfunctioning internet access, vendor 4 quickly discovered that the other organizational actors also suffered from the flawed automated test infrastructure. Given its experience in this domain, vendor 4 offered LogCH and IT LogCH its support to refine the common innovation artifact:

> The helpfulness among each other was something! We had issues with the test platform – it just did not work well. At the same time, it was a requirement of the set definition of done. One of the partners [vendor 4] that knew quite a bit about it, eventually agreed to take over: "I do it for you all, so we get going!" I mean, that was something – and it was not that they had to do this free of charge. It was very pleasing that one was able to let the whole crew – I mean all other partners – take advantage of it. (LogCH – Project Manager)

Vendor 4, however, not only left it at refining the common innovation artifact but also actively approached the others to teach them with the set up testing infrastructure and had an open door for their questions and concerns: *"Then, we also had an open door and conducted stand-up Q&A meetings regarding the test infrastructure. That way, we interacted and were able to talk to the developers. Thus, they were standing in our room, and one was able to exchange ideas with them and to see how they solved certain other things." (Vendor 4—Business Analyst).* In doing so, vendor

4 refined a common innovation artifact, namely the automated test infrastructure. Again, refining a common innovation artifact had paradox consequences for vendor 4. On the one hand, it was extremely cooperative in itself and again helped all organizational actors to materialize the desired coherent system. On the other hand, it gave vendor 4 a competitive advantage over all other organizational actors.

The example of vendor 4 demonstrates how superior expertise can help identify hindering deficiencies in common innovation artifacts and to propose innovative solutions to refine them. Such cooperative actions, which gave one organizational actor competitive advantages over all others, had two consequences for the progress of the innovation ecosystem emergence. First, the refinements of the basic common innovation artifacts meant a further homogenization of the basic procedures, which facilitated the orchestration of innovation agency. Second, innovative solutions that have become standards, such as those of vendor 4, have made the proposing vendors indispensable organizational actors within the innovation ecosystem: *"Sure, they cannot let vendor 4 go—I suppose. Without them, the test thing would probably not run as it should." (Vendor 2—Architect).*

4.3 Phase 2b—Stabilizing Basic Structure

The innovation ecosystem progressed in its emergence with the creation of the basic structure and procedures, as well as the refinement of these procedures. In some cases, however, LogCH also needed to stabilize the basic structure of the innovation ecosystem to ensure it progressed in its emergence. One such incident occurred with vendor 6.

Vendor 6 ignored the defaults regularly and caused additional work for LogCH due to its uncooperative behavior: "*There I did a lot more than my role had foreseen and I had to compensate for things that did not exist." (LogCH—Product Owner 3).* Thus, instead of cooperating with the other organizational actors to capitalize on their expertise or bring in its own expertise, vendor 6 isolated itself and focused exclusively on its own tasks. In other words, vendor 6 destabilized the coopetitive setting:

> Their attitude was not that they valued it [the cooperation with other organizational actors] much – regardless of whether it was us or others. Their participation during the joint or vertical meetings [the meetings on a functional level]… well, there they often shone with their absence. (Vendor 2 – Scrum Master)

It did not even help that LogCH repeatedly urged vendor 6 to attend meetings with the other organizational actors: *"He [the scrum master at vendor 6] attended the meetings because he had to—but he rarely rose to speak. That was not motivating at all. […] It quickly became clear that we are not continuing [to work with vendor 6]." (LogCH—Product Owner 3).* In summer 2014, LogCH considered a further engagement of vendor 6 no longer meaningful. Vendor 6 was not actively contributing with its innovation agency to the progress of the emerging innovation ecosystem, and particularly to the materialization of the desired coherent system. LogCH thus stabilized the basic structure of the innovation ecosystem by excluding vendor 6:

> Back then, we foresaw them to complete this first app, and when performing well, this second app. However, since the remaining apps were already assigned, there was no more work left, and it made no sense to assign vendor 6 to something other vendors had already started. (LogCH – Product Owner 3)

Vendor 6's lack of cooperativeness destabilized the coopetitive setting and risked to prevent the innovation ecosystem emergence from progressing. However, since the two apps developed by vendor 6 were of reasonable quality and only used by a particular type of user, its uncooperative behavior hardly compromised the overall success of project REMO.

In the case of vendor 1, however, LogCH faced a much more threatening situation. Vendor 1 stood out for its boastful behavior during the first kick-off meeting. Interestingly, this competitive behavior led to the task of developing the most significant and central application:

> We arrived there, I would say, in quite a fulminant fashion and with quite some people. This because we said that it is important to get to know each other. For that reason, we took all the potential candidates for our team with us and arrived with 7 to 8 people. The entire team presented itself, at which point they [LogCH] assigned us to develop the largest application. (Vendor 1 – Scrum Master)

Right from the beginning, however, vendor 1 had to realize that project REMO was not just about boasting and competing, but also about cooperating with the other organizational actors for the sake of an innovative, high-quality, coherent, and customer-oriented system: *"I mean the largest and most complex application, that's the app used 80% of the time. Everything else is garnishing. Therefore, one was well aware to hinge on the viability of this very app." (Vendor 4—Scrum Master).* Confronted with all other organizational actors, it did not take long until vendor 1 considered itself a victim of this situation and began to isolate itself:

> In the beginning, us – but certainly me – were victims in this regard, in the sense that we said: "Yes, we take us some time for the other vendors." Eventually, however, we noticed that we could isolate ourselves much more, and in doing so, better reach our own goals. (Vendor 1 – Scrum Master)

Instead of cooperating with the other organizational actors and bringing in its innovation agency, vendor 1 thus began to isolate itself and to focus on its own goals. In doing so, vendor 1 destabilized the coopetitive setting, which did not prevent its scrum master from boasting about *"how big its achievements [the achievements of vendor 1] were for LogCH" (LogCH—Product Owner 1).* However, all the boasting and isolating neither hid the fact that vendor 1 was struggling to develop the assigned application nor its multi-week deviation from the schedule: *"Well, there… that's what I sometimes say about vendor 1. They were… I found… They did not start well." (Vendor 3—Scrum Master 1).* To make matters worse, vendor 1 also faced the development of a second application that further limited its capacities for the first one:

> We not only had to develop the largest and most complex application but also a second one. For that reason, we had to split the team in two for putting the requirements for the second application in place. There were [user] stories that we had to put together. Anyway, very demanding! It was not optimal and left much room for optimizations, I have to admit. That's for sure! (Vendor 1 – Scrum Master).

As the other organizational actors increasingly suffered from the deficiencies of the largest and most complex application, LogCH had to act. Since vendor 1 isolated itself and did not seek support from other organizational actors, LogCH stabilized the basic structure of the innovation ecosystem by redistributing the workload of the second application to vendor 4. In doing so, LogCH redistributed innovation agency from vendor 1 to vendor 4. Interestingly enough, while this allowed vendor 1 to focus on the development of the largest and most complex application, it caused neither a behavioral change at vendor 1 nor an improvement of the situation:

> Others have barely managed to reach their sprint targets – one sprint after another [...] but also transparency-wise – the acknowledgment and the transparency of the progress [...] that was also the case with this escalation concerning the largest and most complex application. (LogCH – Project Manager)

This time, however, it took LogCH a moment to find out about the ongoing issues at vendor 1. Once LogCH learned about them, it was therefore already too late to redistribute the application and innovation agency—the already built-up expertise was too large and too irreplaceable. For this simple reason, LogCH chose another approach and urged vendor 1 to seek support from another organizational actor, which vendor 1 refused to do in the first place:

> We then encouraged vendor 1 "You have to reinforce yourself with a partner [a competing software vendor]." In the first place, they [vendor 1] refused, but then we urged them to do so. (LogCH – Project Manager)

After some additional pressure from LogCH, vendor 1 eventually recognized its impasse, relented, and accepted the support from vendor 3: *"[...] one has realized that the largest and most central application is too big and that we would need help" (vendor 1—Scrum Master)*. The helping function, however, whetted the appetite of vendor 3 for more. Thus, once vendor 3 knew enough about the largest and most complex application, it offered to take over the entire development. LogCH willingly accepted, thus stabilized the structure of the innovation ecosystem by redistributing the application form vendor 1 to vendor 3 and thereby the entire innovation agency. Surprisingly, even this second seizure and the factual exclusion from project REMO did not cause vendor 1 to behave uncooperatively:

> I thought I would feel a little offended in their position. During the transition phase, one had to choose its words carefully – one always had to pay attention to its wording. Nevertheless, I believe they have taken it professionally. In the end, they were not so happy with our PO [product owner 1] – the chemistry was not right. So I do not believe that they were sad being forced to surrender. (Vendor 3 – Developer 2).

Given vendor 1's isolating and whitewashing behavior risked the materialization of the desired system, it also risked to hinder the innovation ecosystem emergence from progressing. Its exclusion was thus inevitable for stabilizing the basic structure of the innovation ecosystem.

4.4 Phase 3—Exploiting Stabilized Structure and Refined Procedures

The creation and adaptation (i.e., refinement and stabilization) of both the basic structure and procedures for materializing the desired coherent innovation outcome has led to steady progress in the emergence of the innovation ecosystem. This steady progress continuously facilitated the coherent materialization of the desired system with all its components and features, which is why the organizational actors together became more innovative as an ecosystem. Thus, although all organizational actors were eager to find innovative solutions themselves, they were also more and more willing and able to harness the coopetitive setting for conserving their own resources or taking advantage of existing innovations:

> Yes, there were certain elements where one realized: "Oh, those two apps need the same ingredient, the same building block they [another vendor] already have while we still need it. Let us wait for them to complete it or let us use their most current version to test and improve it." One has definitively done that – at least if one has realized that there were dependencies and similar issues. (Field Expert – PV)

The biggest challenge was the identification of existing innovative solutions, thus the orchestration. This identification of existing innovative solutions was, however, more and more facilitated by the created and adapted basic structure and procedures, such as the scaled agile scrum method with all its meetings and involved organizational actors:

> That is why the scrum meetings were so vital for observing what the others were doing. This was particularly true for the vendors, as they were able to observe what the other vendors did: "We will have to do something where they already have a solution. Let us approach and ask them how they did it and whether we could borrow and adapt." Therefore, the communication took place – especially between the vendors who worked in the same place. (Field Expert – PV).

Interestingly, the organizational actors then harnessed other common innovation artifacts to exploit the identified innovative solutions. In this respect, the freely accessible code and documentation repositories, which made it possible to copy the innovative solutions from each other without cooperating much, became increasingly popular: *"I remember they copied a sequence in one application [the developers of vendor 2]—just copied and adapted. Since we had access to each other's code, they did not have to interact much." (Vendor 2—Business Analyst 2).* Certain common innovation artifacts have therefore made it possible to replace intensive interpersonal cooperation through the simple duplication of digital resources. In many cases, however, the organizational actors did not manage to copy innovative solutions without cooperating with the innovating actor: *"First of all, there was no unpleasant competition. There might have been even some stories where product owner 2 said to us: "Look, they've [vendor 2] already solved that." [..] One just walked over [and asked them]: "Hey, how did you approach that?"" (Vendor 5—Project Leader).*

An example in which an organizational actor found an innovative solution and later shared it with another was vendor 5's geolocation feature. Interestingly, even vendor 5 had little experience in implementing such a feature in the first place. However, vendor 5 was able to leverage the expertise of an expert in this field, who happened to be on a sabbatical:

> During the implementation of the last app, we had to integrate a geolocation function. That is something I have no experience with it. Fortunately, we had access to our own expert [...] he just lived in Geneva for a year, and we were able to involve him. That was just great! The first day he was on site, he looked at the device and knew exactly what is going to work. This spared many issues. (Vendor 5 – Business Analyst)

Vendor 5 thus found an innovative solution on its own. A short time later, vendor 2 was not as fortunate when it faced a similar challenge. However, vendor 2 learned about this innovative solution from product owner 2. In other words, product owner 2 orchestrated the problem and the innovative solution:

> I remember a situation at the beginning where one of our partners [vendor 2] did something in the same area [geolocation]. However, this did not work out, and the first day our expert [for geolocation] was on site, their scrum master and product owner 2 came over to talk to him. Only after two minutes, he was able to tell them why their solution is not working, and he hit bulls' eye! (Vendor 5 – Business Analyst)

Vendor 2 was therefore able to bridge its lack of expertise and to find an innovative solution to a faced challenge thanks to the help of vendor 5. In doing so, vendor 2 harnessed both common innovation artifacts and the coopetitive setting for materializing the desired coherent innovation outcome. At the same time, this increased the competitiveness of vendor 5, as it was able to differentiate itself from the other organizational actors with its superior expertise in a specific domain: *"Well, that was certainly something! We were able to bring in a 'vendor 5 differentiator'. I believe not everyone [the other vendors] would have had the skills to do something similar." (Vendor 5—Project Lead).* Eventually, its cooperativeness that increased its competitiveness paid off, as LogCH assigned vendor 5 to develop two more applications:

> It was pleasing – back then two additional apps were popping up on the horizon. We did not expect that but then we were able to implement them. Back then, I would say, the tough competition was... well, I never had the feeling that another vendor was taking anything from us; let us put it like that. It was clear that one has to deliver; otherwise, there was somebody else that made it better. (Vendor 5 – Business Analyst)

Tables 2, 3, 4, 5, and 6 provide further illustrative evidence for the four stages.

Table 2 Illustrative evidence for phase 1

Stage	Trigger	Innovation agency	Structure/Procedures	Cooperation/Competition
Phase 1 Creating	*Rigid system architecture* Before project REMO, LogCH supported its employees in their day-to-day tasks with mobile computing devices. The architectures of these devices were ***monolithic and did not allow the replacement of individual components or features***	***LogCH*** had to find a system architecture with independent components (i.e., hardware, software platform, framework, features) and features (i.e., applications)	*Modular system architecture* LogCH opted for a ***modular cross-compiler architecture (structure & procedures)*** with independent components and features. LogCH ***involved IT LogCH*** to contribute to the framework ***and the Field Experts*** to revise the processes and define 20 apps	Opting for a flexible cross-compiler architecture enabled the definition of independent components and features. However, to obtain a coherent system, the components and features had to be aligned, which ***assumed the cooperation of the involved organizational actors***
	Limited time horizon Until the approaching end-of-life of the old mobile computing device, ***LogCH had only a little time to implement project REMO*** for supporting its employees with a new mobile computing device in the future	***LogCH*** had to find a solution to develop the twenty predefined applications, evaluate multiple hardware devices, roll out the new mobile computing system, and train its employees in just about two years	*Division of labor* LogCH divided project REMO into subprojects and ***involved six software vendors (structure)***, each with identic capabilities to design, build, test, deploy, and run the twenty predefined applications	Choosing six software vendors with the identic capabilities to design, build, test, deploy, and run the twenty predefined applications ***enabled the competition between these six software vendors***

(continued)

Table 2 (continued)

Stage	Trigger	Innovation agency	Structure/Procedures	Cooperation/Competition
	Methodological hurdles Before project REMO, LogCH relied on the waterfall model for developing software. Given the complexity of project REMO with a variety of organizational actors and a limited time horizon, however, this ***waterfall model was inappropriate and too risk***	***LogCH*** had to find a development method that could cope with both the complexity of the new system with all its components and features and the multitude of development partners. For that purpose, LogCH leveraged the expertise of a hired ***scrum expert***	*Scaled scrum method* LogCH opted for an ***agile scaled scrum development method*** with ***two-weekly sprints and meetings at the team-, functional-, and project level (procedures)***. The scrum expert, who also trained the other organizational actors, specified this method	Opting for an agile scaled agile scrum development method with two-weekly sprints and meetings at the team-, functional-, and project level, made the organizational actors more comparable, which ***facilitated the cooperation and the competition*** between them in project REMO
	Missing guidelines The division of labor among multiple organizational actors and the agile scaled scrum development method confronted LogCH with a ***potential threat of proliferation***	***LogCH*** had to find a solution to coordinate the work of the multiple organizational actors in a way that resulted in a coherent and innovative system for its employees in the field	*Standardized guidelines* LogCH stipulated ***standardized guidelines for the architecture, design, development, documentation, and testing of the applications and other software components (procedures)***	Stipulating guidelines to coordinate the work of the organizational actors in a way that resulted in a coherent and innovative system led to a ***means for ensuring meaningful cooperation despite the coopetitive setting***

(continued)

Table 2 (continued)

Stage	Trigger	Innovation agency	Structure/Procedures	Cooperation/Competition
	Inconsistent development means The division of labor among multiple organizational actors implied the use of different development means, which confronted LogCH with a ***potential threat of proliferation***	***LogCH*** had to find a solution that reduced the potential threat of proliferation caused by the organizational actors using different and potentially inconsistent development means	*Standardized development means* LogCH ***defined and provided a stack of software (procedures)*** and ***made the code- and documentation repositories freely accessible (procedures)*** to all organizational actors	Defining and providing a stack of software and making the code- and documentation repositories freely accessible, further ***facilitated the cooperation and the competition*** between the organizational actors
	Geographical distribution The division of labor among multiple organizational actors implied a geographical distribution, which ***potentially increased the coordination effort*** for LogCH	***LogCH*** had to find a solution that allowed to reduce the possible negative consequences of the geographical distribution of the organizational actors and in particular the own coordination effort	*Co-location* LogCH ***co-located the organizational actors in a shared office space (procedures)*** with individual and shared rooms. Besides, LogCH also provided the vendors with a coffee machine	Co-locating the organizational actors and providing a coffee machine facilitated the interaction between them, which further ***facilitated the cooperation and the competition*** between them

Table 3 Illustrative evidence for phase 2a

Stage	Trigger	Innovation agency	Procedures	Cooperation/Competition
Phase 2a Refining	*Unfinished framework* The framework built by IT LogCH was incomplete when the application development began, which caused overlaps that led to ***incompatibilities and numerous issues and complaints from the six software vendors***	***LogCH*** had to find solutions to reduce the incompatibilities and the related issues and complaints of the ***six software vendors***, as the six software vendors themselves were unable to do much to counter them	*Postponement and co-location* LogCH ***postponed IT LogCH's scrum cycle by one week*** for detecting threatening incompatibilities earlier and ***urged IT LogCH to join the six software vendors in the shared office space*** for remedying issues faster	Facing similar incompatibilities, the ***six software vendors cooperated*** or rather fraternized. Therefore, LogCH postponed IT LogCH's scrum cycle and co-located them, thus ***facilitated the cooperation between IT LogCH and the six software vendors***
	Vague documentation guidelines The stipulated documentation guidelines were ***unknown to all organizational actors and found to be particularly inadequate*** by the six software vendors	The ***six software vendors*** found themselves constrained to make proposals to remedy the identified inadequacies of the documentation guidelines. Ultimately, however, ***LogCH*** decided on these proposals	*Remedied guidelines* ***Vendor 3 and Vendor 2 made proposals*** to remedy the identified inadequacies of the documentation guidelines, and ***LogCH set them as default*** for all organizational actors	Making proposals to remedy identified inadequacies, LogCH set as standards, ***gave vendor 3 and vendor 2 a competitive advantage*** and ***improved the means for ensuring meaningful cooperation despite the coopetitive setting***

(continued)

Table 3 (continued)

Stage	Trigger	Innovation agency	Procedures	Cooperation/Competition
	Vague design guidelines The stipulated usability and design guidelines were found to be ***mostly inadequate to obtain a state-of-the-art, coherent system*** developed by multiple organizational actors, including six competing software vendors	The ***six software vendors*** fund themselves constrained to make proposals to remedy the identified inadequacies of the design and usability guidelines. Ultimately, however, ***LogCH*** decided on these proposals	*Unremedied guidelines* ***The six software vendors made proposals*** to remedy the identified inadequacies of the usability and design guidelines. However, ***LogCH decided to leave the usability and design guidelines mostly unchanged***	Realizing that LogCH left the usability and design guidelines mostly unchanged—this despite the remedies proposed—the ***six software vendors began competing*** for the best, but not necessarily the most consistent designs and usability logics
	Malfunctioning internet access The shared office space provided by LogCH had only poor and error-prone access to the internet, ***hindering all organizational actors in their work***—particularly the six software vendors in developing their applications	***Vendor 4*** noticed that all organizational actors suffered from the same inadequate and error-prone internet access. Against this backdrop, Vendor 4 fund itself constrained to find a solution and therefore approached ***LogCH***	*Functioning internet access* ***Vendor 4*** used its experience to ***provide more reliable internet access to all organizational actors.*** Thus, instead of complaining about the status quo, Vendor 4 took one for the team and helped everybody	Providing more reliable internet access was a ***highly cooperative act of vendor 4*** that helped all organizational actors to proceed with their tasks. At the same time, it ***gave vendor 4 a competitive advantage*** over all other organizational actors

(continued)

Table 3 (continued)

Stage	Trigger	Innovation agency	Procedures	Cooperation/Competition
	Malfunctioning test infrastructure The agile scaled scrum method required the six software vendors and IT LogCH to test their sprint increments with an ***immature and error-prone test infrastructure for complying with the definition of done***	***Vendor 4*** noticed that all organizational actors suffered from the same immature and error-prone test infrastructure. Against this backdrop, Vendor 4 fund itself constrained to find a solution and therefore approached ***LogCH***	*Functioning test infrastructure* ***Vendor 4*** used its experience to support IT LogCH in ***providing a more reliable test infrastructure and training the other organizational actors*** in its use. Thus, Vendor 4 took one for the team and helped everybody	Providing a more reliable test infrastructure and training all organizational actors in its use were ***highly cooperative acts of vendor 4*** that helped all organizational actors. At the same time, these acts ***gave vendor 4 a competitive advantage***
	Neglected feature library IT LogCH increasingly ***neglected the maintenance and enhancement of the feature library*** for which it was responsible. This neglect primarily affected the six software vendors	***LogCH*** had to find solutions for the poorly maintained and barely extended feature library, as the ***six software vendors*** themselves were unable to do much to counter this neglect	*Copy and own policy* ***LogCH introduced a copy and own policy***, which allowed the six software vendors to copy features from each other without having to cooperate and to own these features after that	Facing similar issues, the ***six software vendors cooperated*** or rather fraternized. Therefore, LogCH introduced a ***copy and own policy that fostered cooperation and competition among the six software vendors***

Table 4 Illustrative evidence for phase 2a (continued)

Stage	Trigger	Innovation agency	Procedures	Cooperation/Competition
Stage 2a Refining	*Unknown hardware* The device was initially unknown, which did not ***prevent the six software*** vendors from developing their applications but ***from testing their performance on the device***	***LogCH*** needed to look for bridging solutions for the initially unknown devices. The ***six software vendors*** themselves could not do much about it but still had to test the device performance of their applications	*Bridging solution* ***LogCH bought devices that met the requirements stated in the call for tenders***, which allowed the software vendors to test their applications' performance on a device	Facing similar issues, the ***six software vendors cooperated*** or rather fraternized. Therefore, LogCH bought devices that ***facilitated the cooperation between the software vendors and the Field Experts***
	Immature integration ***IT LogCH struggled to integrate all software components and hardware***, which prevented the software vendors from running their applications on the device	***Vendor 4*** noticed that all organizational actors suffered from immature hardware-software integration. Against this backdrop, Vendor 4 offered to support ***LogCH*** and ***IT LogCH*** in this regard	*Mature integration* ***Vendor 4*** used its experience to ***support IT LogCH with a mature integration and to train everybody else***. Thus, vendor 4 took one for the team and helped everybody	Supporting IT LogCH in integrating all software components and hardware, and training all, were ***highly cooperative acts of Vendor 4.*** At the same time, these acts ***gave Vendor 4 a competitive advantage***

(continued)

Table 4 (continued)

Stage	Trigger	Innovation agency	Procedures	Cooperation/Competition
	Cognitive boundaries The ***Field Experts were not very experienced in software development, which particularly hampered the mutual understanding*** between them and the six software vendors during the joint sprint reviews and planning	***Vendor 2*** noticed the cognitive boundaries between its employees and the Field Experts. Against this backdrop, Vendor 2 fund itself constrained to find solutions for bridging these cognitive boundaries	*Visual mock-ups* ***Vendor 2*** came up with the idea of ***using mock-ups to visualize the visions of their software developers*** for subsequent sprints, allowing field experts to more easily understand and comment on these visions.	Using mock-ups to visualize the software developers' visions for future sprints was adopted by the other software vendors and not only ***facilitated the cooperation with the Field Experts*** but also ***increased the competitiveness of Vendor 2***
	Inadequate backend involvement ***The backend systems maintained by IT LogCH were not included from the beginning.*** However, the six software vendors had to build interfaces to the backend systems early on	The ***six software vendors*** and ***IT LogCH*** needed to find solutions for bridging the previously not included backend systems to ensure a coherent system with functional access to the backend system	*Backend interface emulation* ***The six software vendors began to emulate the interfaces*** of the backend systems, while ***IT LogCH supported them with backend integrators***	Receiving support from backend integrators while emulating the backend interfaces allowed the six software vendors to proceed with their development and ***eased the cooperation with IT LogCH***

Table 5 Illustrative evidence for phase 2b

Stage	Trigger	Innovation Agency	Structure	Cooperation/Competition
Stage 2b Stabilizing	*Issues of vendor 1* Vendor 1 developed the most central and complex application with interdependencies to all organizational actors. However, ***vendor 1 had more and more difficulty dealing with this complexity***, which primarily affected the development of the application	***Vendor 1*** needed to find solutions to address the complexity while ensuring the development of its applications. Dissatisfied with Vendor 1, ***LogCH*** had to intervene and reduce the development backlog of the most central and sophisticated application	*Redistributing application* Vendor 1 initially chose to lie about its difficulties and isolated itself for focusing on the development of the most central and sophisticated application. ***LogCH noticed these ongoing difficulties and reassigned a second application to vendor 4***	Lying over its difficulties ***did not help vendor 1 to maintain its competitiveness***, that initially led to the assignment of the most central and sophisticated application. The opposite was the case, as ***vendor 1 had to cooperate with vendor 4*** for handing over a second application
	Continued issues of vendor 1 After having handed over a second application to vendor 4, vendor 1 continued to develop the most central and sophisticated application. Unsurprisingly, ***vendor 1 continued to face similar difficulties as before***	***Vendor 1*** needed to find solutions to address its difficulties. Dissatisfied with the solutions of Vendor 1, ***LogCH*** had to intervene a second time and find its solutions to reduce the development backlog of the most central and sophisticated application	*Redistributing app and excluding* Vendor 1 chose to lie about its difficulties and isolated itself. ***LogCH noticed these difficulties and urged vendor 1 to seek support***. Vendor 1 did so by approaching ***vendor 3 that eventually took the application over***	Lying over its difficulties ***did not help Vendor 1 to maintain its competitiveness.*** The opposite was the case, as ***Vendor 1 had to cooperate with Vendor 3*** for its support and for handing over a second application

(continued)

Table 5 (continued)

Stage	Trigger	Innovation Agency	Structure	Cooperation/Competition
	Isolation of vendor 6 Vendor 6 early on focused exclusively on its tasks and only reluctantly, if ever, participated during the meetings with the other organizational actors. Although vendor 6 progressed with its applications, ***LogCH had to compensate this uncooperativeness***	***LogCH*** had to find solutions to reduce its efforts, which resulted from the uncooperativeness of Vendor 6. This need to find solutions to reduce its efforts was especially critical, as Vendor 6 was not aware of any misconduct whatsoever	*Excluding organizational actor* LogCH repeatedly urged vendor 6 to participate more actively in the joint scrum meetings. However, this did not help much, and vendor 6 continued to isolate itself. Ultimately, ***LogCH decided to exclude vendor 6*** after the development of two applications	***Urging vendor 6 to cooperate*** more actively in the joint scrum meetings did not work. Given the fact that other software vendors were more cooperative in these joint scrum meetings than vendor 6 and still more competitive, ***LogCH excluded vendor 6*** after just two applications
	Staff fluctuations Vendor 2 and vendor 5 experienced staff fluctuations caused either by internal decisions or by LogCH. Such fluctuations were particularly problematic as ***they led to knowledge losses that hindered development***	***LogCH*** had to find solutions to bridge such staff fluctuations, which is primarily to reduce the loss of knowledge and to obtain a coherent system within the short period before the end-of-life of the previous system	*Compensating staff fluctuations* Since vendor 2 and vendor 5 were unable to compensate for the loss of knowledge on their own, ***LogCH stabilized it with training sessions for the new staff***	Bridging the lost knowledge caused by staff fluctuation with training sessions ***helped to maintain the cooperativeness and competitiveness of Vendor 2 and Vendor 5*** following such staff fluctuations

Table 6 Illustrative Evidence for Phase 3

Stage	Trigger	Innovation Agency	Structure/Procedures	Relation mode
Stage 3 Exploiting	*Development issues* The six ***software vendors and IT LogCH repeatedly faced issues within their development tasks they were unable to remedy on their own***. These inabilities presented risks to the schedule and the coherent system	The ***six software vendors*** and ***IT LogCH*** had to find solutions to faced issues they were unable to remedy on their own. Supporting the vendors in doing so was of interest to LogCH and the Field Experts, as both aimed for a coherent system	*Supporting each other* Since the six software vendors and IT LogCH were not always able to solve faced issues on their own, they ***identified solutions or supporters themselves (procedures) or were pointed to them by LogCH and the Field Experts (structure)***	Identifying potential supporting organizational actors led to either ***cooperation for solving faced issues***, which also ***increased the competitiveness of the supporter***, or to mere replications of solutions found in the open repositories
	GPS functionality issue Both vendor 2 and vendor 5 faced the issue of having to implement a geolocation feature in some of their applications. However, ***while vendor 5 had the necessary expertise to do so, vendor 2 did not***	***Vendor 2*** was not as fortunate as ***vendor 5*** and had to find a solution for implementing the geolocation feature in its application. Since ***LogCH*** aimed for a coherent system, it was in its very interest to support vendor 2 in the quest for a solution	*Supporting vendor 2* ***LogCH became aware of vendor 5's geolocation expertise and informed vendor 2 about this (structure)***. As a result, ***vendor 2 turned to vendor 5 in the shared office (procedure)*** for support in implementing the geolocation feature in its application	Identifying vendor 5's geolocation feature, ***vendor 2 approached vendor 5 and began to cooperate*** for finding a solution to its issue. That way vendor 2 was able to remedy its issue, while ***vendor 5 was able to show its competitiveness to LogCH***

5 Analytical Summary—Developing a Process Model

Our results indicate that an innovation ecosystem progresses to emerge along three different phases, i.e. from the (1) creation, over the (2) adaptation (i.e., (a) refinement and (b) stabilization), toward the (3) exploitation of the underlying structure and procedures. Figure 2 illustrates the resulting process model. In the first phase, one or a few central organizational actors desire a coherent digital innovation. However, they lack the expertise to materialize this coherent digital innovation on their own, which is why they create both the basic structure and procedures of an innovation ecosystem. Concerning the basic structure, the central organizational actors first specify their desires. To materialize these desires, the central organizational actors then determine the necessary capabilities to involve peripheral organizational actors and distribute innovation agency among them. In doing so, the central organizational actors shape the coopetitive setting. In project REMO, LogCH (i.e., the central organizational actor) desired a coherent mobile computing system with a modular cross-compiler architecture. To materialize this system, LogCH needed the capabilities of its independent IT department to develop a framework, field experts to specify applications, and multiple software vendors to build these applications on time. At the same time, LogCH had to distribute innovation agency and to ensure continued cooperation among the peripheral organizational actors. To ensure such continuous cooperation, central organizational actors need to create basic procedures in terms of common innovation artifacts. These common innovation artifacts determine how all involved organizational actors have to materialize a desired digital innovation. In project REMO, these common innovation artifacts included, among others, a standard development method, various development guidelines, and a stack of development tools. Interestingly, not only LogCH defined such common innovation artifacts but also the peripheral organizational actors, for example, IT LogCH that defined the framework.

Once the basic structure and procedures have been created, any hindering or even threatening deficiencies can be identified. In these cases, the basic structure and procedures require adaptations to ensure the materialization of a desired, coherent and customer-oriented digital innovation. More specifically, the procedures require refinements and the structure stabilization. Concerning the basic procedures, distributed innovation agency enables all organizational actors to identify hindering deficiencies of common innovation artifacts and propose innovative solutions to their remedy. The decision whether these innovative solutions lead to refinements of common innovation artifacts, and thus the basic procedures, lies with the central organizational actors. In project REMO, several organizational actors proposed innovative solutions to remedy hindering deficiencies of common innovation artifacts. If LogCH accepted an innovative solution, the refinement of the common innovation artifact was generally very cooperative and made the innovation ecosystem progress in its emergence. At the same time, the refinement of the common innovation artifact also increased the competition between the organizational actors healthily, thereby shaping the coopetitive setting. Besides procedural deficiencies, however, emerging

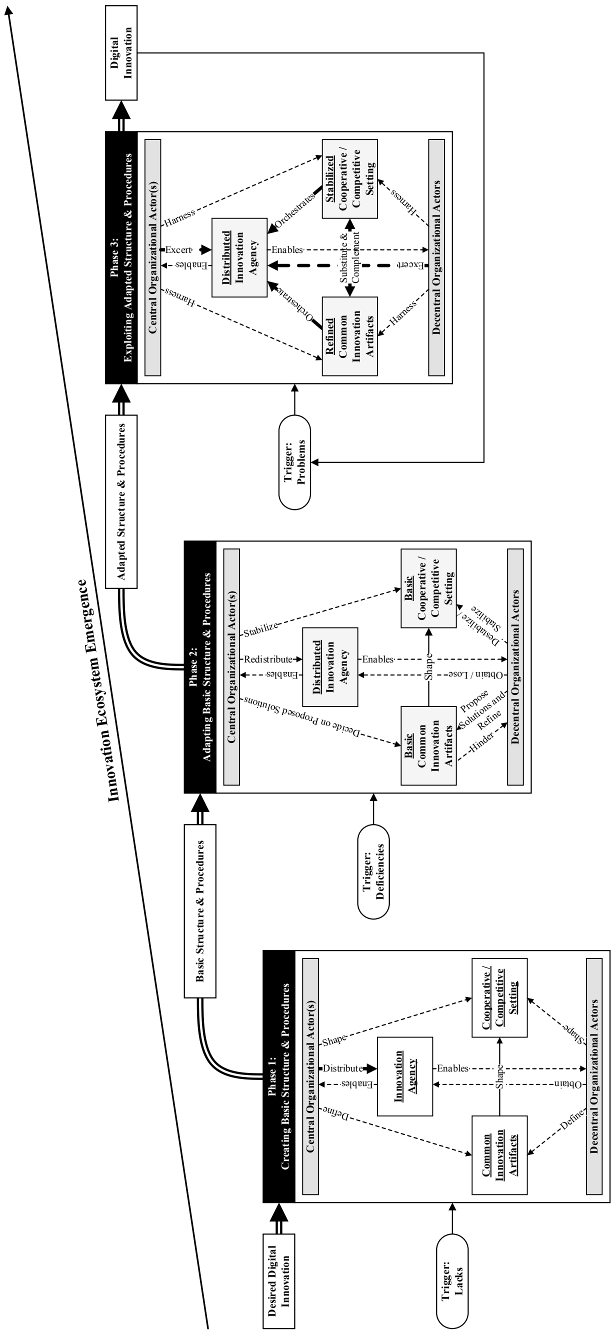

Fig. 2 Process model of innovation ecosystem emergence

innovation ecosystems may also face structural ones. As with procedural deficiencies, all organizational actors can identify them. However, different from procedural deficiencies, peripheral organizational actors can hardly make innovative solutions to remedy structural deficiencies. It is therefore up to the central organizational actors to stabilize them. Thus, while all organizational actors can detect structural deficiencies, only the central organizational actors decide on a possible redistribution of innovation agency to stabilize the coopetitive setting. In project REMO, LogCH needed to stabilize the coopetitive setting multiple times, which eventually led to the exclusion of two peripheral organizational actors and the redistribution of their innovation agency to others. In both illustrated cases, the structural deficiency manifested in the intense competitive thinking of the two peripheral actors and their lack of cooperation, which risked the coherent materialization of the desired digital innovation.

Once the basic structure and procedures of an innovation ecosystem have been adapted, all organizational actors can exploit them to materialize a desired digital innovation. More specifically, all organizational actors can harness both the refined common innovation artifacts and the stabilized coopetitive setting as orchestration entities to match problems with innovative solutions. Concerning the stabilized coopetitive setting, this means that the organizational actors are cooperative enough to share innovative solutions and support others, even if they compete with them. In project REMO, the stabilized coopetitive setting resulted in exactly this—it facilitated and even promoted the cooperation between the vendors despite their own goals. Based on the stabilized structure, for example, LogCH was able to match the innovative solution of vendor 5 to the problem of vendor 3 and to foster the cooperation between them. This example and others illustrate how a stabilized coopetitive setting acts as an orchestration entity that fosters the materialization of a desired and coherent digital innovation. The same applies to refined basic procedures in terms of refined common innovation artifacts. Based on refined basic procedures, all organizational actors can exploit an innovation ecosystem better in terms of matching solutions to problems with the help of refined common innovation artifacts. In project REMO, for example, certain refined common innovation artifacts enabled the organizational actors to identify and borrow or duplicate innovative solutions from others to solve their own problems.

6 Discussion

The goal of this study was to answer the research questions of (1) how central organizational actors create innovation ecosystems and (2) how and why such innovation ecosystems progress in their emergence over time and through the interplay of all involved organizational actors that pursue both common and own goals. To answer these questions, we chose to conduct a longitudinal single-case study about the software development project REMO with nine partially competing organizational actors that had to cooperate for a coherent and customer-oriented mobile computing system to materialize. This constellation has proved to be uniquely suited to answer our research questions for three particular reasons. First, the created basic structure of project REMO was initially competitive and only became cooperative, or rather coopetitive, over time. Such a development from competitive to coopetitive is particularly well suited to understand why and how organizational actors with their own goals begin to join forces for materializing a coherent and customer-oriented digital innovation together. Interestingly enough, such evolutions from competitive to coopetitive structures are rarely found in previous research on inter-organizational collaboration (Majchrzak et al. 2014). Second, project REMO relied on the distribution of innovation agency to all organizational actors in the innovation ecosystem. Such a distribution of innovation agency is a prerequisite for understanding how central organizational actors create innovation ecosystems and distribute innovation agency, as well as how and why all organizational actors subsequently innovate together and materialize a coherent and customer-focused digital innovation. Setting with distributed innovation agency can hardly be found in traditional research on innovation (Nambisan et al. 2017). Third, the desired mobile computing system was both innovation outcome and an essential part of the innovation process. This circumstance is barely known from traditional literature on innovation, but it is increasingly the case with digital innovations (Nambisan et al. 2017). In innovation ecosystems, however, this increasingly blurred boundary between the innovation process and outcome is essential to materialize coherent and customer-oriented digital innovations.

In response to our first research question, our findings highlight the creative role of central organizational actors regarding the basic structure and procedures. In terms of the basic structure, central organizational actors first specify the desired digital innovation. Based on this desired digital innovation, the central organizational actors then derive the necessary competencies, select organizational actors with these competencies, and distribute innovation agency among them. In project REMO, the central organizational actor aimed at materializing a coherent mobile computing device with a modular cross-compiler architecture. To achieve this goal, LogCH required the capabilities of its independent IT department to develop a framework, field experts to specify applications, and multiple software vendors to build these applications on time. In terms of the basic procedures, central organizational actors and selected peripheral organization actors define common innovation artifacts that determine how desired digital innovations are to be materialized by all organizational actors in an innovation ecosystem. In the REMO project, these common innovation artifacts included, for example, a standardized development method, a stack of development tools, and guidelines.

In response to our second research question, we find that innovation ecosystems progress to emerge in three different phases, from the (1) creation, over the (2) adaptation (i.e., (a) refinement and (b) stabilization), toward the (3) exploitation of the underlying structure and procedures. In the first phase, as described above, central organizational actors create the basic structure and procedures of an innovation ecosystem. However, these basic structures and procedures are usually not ideally suited for the materialization of digital innovation by multiple organizational actors with their own goals. For this reason, they are adapted in a second phase. More specifically, the central organizational actors stabilize the basic structure by distributing and redistributing innovation agency, while the peripheral organizational actors help refine the basic procedures. In terms of the basic procedures, the peripheral organizational actors thus assume their innovation agency to refine common innovation artifacts. Once the basic structure and procedures are adapted to the needs of an innovation ecosystem, all organizational actors can exploit them. More precisely, the adapted basic structure and procedures act as orchestration entities for matching solutions with problems during the materialization of desired coherent and customer-oriented digital innovations. Together, these findings theoretically contribute to the literature on innovation ecosystems, digital innovation, and broader research on the coopetition paradox.

6.1 Implications for Theory

Our study has important implications for research on innovation ecosystems, in particular on their emergence. So far, research on innovation ecosystems has investigated how central organizational actors compel other organizational actors to commit to creative efforts in situations where uncertainty is high (Dattée et al. 2018) or move from 'blind dating' toward 'arranged marriages' among themselves (Paquin and Howard-Grenville 2013). While this research has broadened our understanding of the creation of innovation ecosystems, it has focused exclusively on the perspective of central organizational actors (Lumineau and Oliveira 2018). Our results differ from these one-sided insights into innovation ecosystems by looking at several and partially competing organizational actors that, over time, cooperate for a coherent and customer-oriented mobile computing system to materialize. This more sophisticated view on the creation and emergence of an innovation ecosystem allowed us to shed light on the procedural dynamics without limiting ourselves to the potential orchestrating actions of central organizational actors. Notably, this more sophisticated view has enabled us to show that innovation ecosystems emerge in three different phases, from the (1) creation, over the (2) adaptation (i.e., (a) refinement and (b) stabilization), toward the (3) exploitation of the underlying structure and procedures. In contrast to earlier studies, we explicitly emphasize the importance of peripheral organizational actors during these phases and particularly concerning their innovative solutions for refining the basic procedures. Only if the common innovation artifacts effectively support the organizational actors in materializing coherent and customer-oriented digital innovation, an innovation ecosystem progresses to emerge. Besides, however, we also emphasize the importance of stabilizing adaptations of the basic structures of innovation ecosystems by central organizational actors. Coherent and customer-oriented digital innovation can only be materialized if the structure of an innovation ecosystem is stable in terms of the composition of the organizational actors. Thus, the better adapted the structure and the procedures, the better organizational actors can exploit them to materialize coherent and customer-oriented digital innovation.

Our study also has important implications for research on digital innovation. In particular, our findings help to understand how and why innovation agency is distributed and redistributed among organizational actors in innovation ecosystems. So far, traditional research on innovation has focused on predefined sets of focal innovation agents, such as the employees in research and development divisions. In doing so, traditional research on innovation has broadly neglected questions regarding how and under which conditions innovation agency becomes distributed and redistributed among large numbers of organizational actors (Nambisan et al. 2017). Such a general distribution of innovation agency, which makes every single organizational actor a potential innovator, is at the center of innovation ecosystems. However, our results show that it is not just about the distribution of innovation agency, but also about its redistribution over time and its actual use by organizational actors. More specifically, we show that central organizational actors need to stabilize the basic structure of an innovation ecosystem by reassigning innovation agency from organizational actors that refrain from actively contributing during the materialization of a desired digital innovation. The absence of such self-defense mechanism would hinder an innovation ecosystem from progressing in its emergence, which would also prevent the materialization of a desired digital innovation. Besides, our results also shed light on the orchestration of digital innovation in terms of a dynamic matching of innovative solutions to problems through the structure and procedures of an innovation ecosystem. More specifically, we were able to show that organizational actors, as well as common innovation artifacts, can act as orchestration entities for matching innovative solutions to problems. Thus, the better the structure and procedures help to match innovative solutions to problems, the more progressed an innovation ecosystem is in its emergence.

As a final contribution, our study has important implications for the broader literature on coopetition and especially on the aspect of the balancing cooperation and competition (Bengtsson and Kock 2014). So far, the call for exploring the optimal blend between cooperation and competition has remained unaddressed (Ketchen et al. 2004). We contribute to this gap in two ways. First, we show that both cooperation and competition are essential ingredients for the emergence of innovation ecosystems. Cooperation is essential in situations where organizational actors do not have the required knowledge to solve problems on their own. In these situations, cooperation between organizational actors is crucial for identifying innovative solutions fast to continue with the materialization of coherent and customer-oriented digital innovation. Competition is essential for the ongoing search for innovative solutions but must not harm the coherent and customer-oriented materialization of a desired digital innovation. Second, we show that coopetition is not naturally existent in innovation ecosystems, which urges organizational actors to establish coopetitive situations.

6.2 Practical Implications

Established companies are increasingly joining forces with other organizational actors in innovation ecosystems. In doing so, these companies use their collective innovative power to co-create and modify digital innovation for reducing the risk of disruptive technologies and unexpected competitors (Adner 2006, 2017). In this regard, our results provide meaningful guidance for practitioners on how to create innovation ecosystems and how to promote the emergence of an innovation ecosystem over time for materializing coherent customer-oriented digital innovation. First, we show that central organizational actors can create the basic structure and procedures of an innovation ecosystem. To this end, central organizational actors must create the basic structure of an innovation ecosystem by specifying the desired digital innovation, deducting the required capabilities for materializing this digital innovation, and involving organizational actors with these required capabilities. At the same time, central organizational actors must also create the basic procedures in terms of common innovation artifacts that determine how a desired digital innovation is to be materialized by and between the involved organizational actors. Examples for such common innovation artifacts include, for example, guidelines, prescribed development tools, or standardized development methods. Second, we show that for an innovation ecosystem to progress in its emergence, central organizational actors must consider the innovative solutions of peripheral organizational actors for refining the common innovation artifacts, as well as constantly stabilize the basic structure. In other words, while central organizational actors can create the basic structure and procedures of an innovation ecosystem, they should by no means ignore the innovative solutions of the peripheral actors to adapt them. The better adapted the structure and the procedures of an innovation ecosystem, the better organizational actors can exploit them to materialize coherent and customer-oriented digital innovation.

6.3 Future Research

Future research can extend our findings in at least two promising directions. First, our study focused on how central organizational actors create innovation ecosystems and how and why such innovation ecosystems progress in their emergence over time until they are well balanced and running to the satisfaction of all organizational actors. In other words, the organizational actors within an innovation ecosystem can exploit the structure and procedures to materialize coherent and customer-oriented digital innovations. However, we still know little about how such innovation ecosystems continue to emerge once they are well balanced and running, particularly if new organizational actors join. This raises the question of how additional organizational actors should be included in innovation ecosystems with well-adapted structures and procedures. For example, should the existing organizational actors consider the potential agendas and strategies (i.e., Hannah and Eisenhardt 2018) of these newcomers? If so, how could the existing organizational actors identify and respond to these agendas and strategies for guaranteeing a continuous emergence of their innovation ecosystem? Second, our study shows that created and refined procedures are essential drivers for the emergence of an innovation ecosystem. However, we still know little about how the specific natures of the underlying common innovation artifacts contribute to the emergence of an innovation ecosystem. This raises the question of how specific types of common innovation artifacts contribute to the coherent and customer-oriented materialization of digital innovations. For example, are guidelines more efficient common innovation artifacts for achieving coherent and customer-oriented digital innovations than prescribed infrastructures? If so, how could these differences be exploited?

References

Adner, R. (2006). Match your innovation strategy to your innovation ecosystem. *Harvard Business Review, 84*(4), 98–107. https://hbr.org/2006/04/match-your-innovation-strategy-to-your-innovation-ecosystem.

Adner, R. (2017). Ecosystem as structure: An actionable construct for strategy. *Journal of Management, 43*(1), 39–58. https://doi.org/10.1177/0149206316678451.

Adner, R., & Kapoor, R. (2010). Value creation in innovation ecosystems: How the structure of technological interdependence affects firm performance in new technology generations. *Strategic Management Journal, 31*(3), 306–333. https://doi.org/10.1002/smj.821.

Autio, E., & Thomas, L. (2014). Innovation ecosystems: implications for innovation management. In M. Dodgson, D. M. Gann, & N. Phillips (Eds.), *The Oxford handbook of innovation management* (Vol. 1, pp. 204–228). Oxford UK: Oxford University Press.

Bengtsson, M., & Kock, S. (2000). "Coopetition" in business networks—To cooperate and compete simultaneously. *Industrial Marketing Management, 29*(5), 411–426. https://doi.org/10.1016/S0019-8501(99)00067-X.

Bengtsson, M., & Kock, S. (2014). Coopetition—Quo vadis? Past accomplishments and future challenges. *Industrial Marketing Management, 43*(2), 180–188. https://doi.org/10.1016/j.indmarman.2014.02.015.

Block, R. (2007, January 9). Live from Macworld 2007: Steve Jobs keynote. Retrieved from https://www.engadget.com/2007/01/09/live-from-macworld-2007-steve-jobs-keynote/.

Boudreau, K. J., & Haigu, A. (2009). Platform rules: Multi-sided platforms as regulators. In A. Gawer (Ed.), *Platforms, markets and innovation* (pp. 163–191). Cheltenham, UK: Edward Elgar.

Charmaz, K. (2006). *Constructing grounded theory: A practical guide through qualitative analysis*. Thousand Oaks, CA: SAGE Publications.

Dattée, B., Alexy, O., & Autio, E. (2018). Maneuvering in poor visibility: How firms play the ecosystem game when uncertainty is high. *Academy of Management Journal, 62*(2), 466–498. https://doi.org/10.5465/amj.2015.0869.

Davis, J. P. (2016). The group dynamics of interorganizational relationships: Collaborating with multiple partners in innovation ecosystems. *Administrative Science Quarterly, 61*(4), 621–661. https://doi.org/10.1177/0001839216649350.

Dhanaraj, C., & Parkhe, A. (2006). Orchestrating innovation networks. *Academy of Management Review, 31*(3), 659–669. https://doi.org/10.5465/amr.2006.21318923.

Eisenhardt, K. M. (1989). Building theories from case study research. *The Academy of Management Review, 14*(4), 532–550. https://doi.org/10.5465/amr.1989.4308385.

Furr, N., & Shipilov, A. (2018). Building the right ecosystem for innovation. *MIT Sloan Management Review, 59*(4), 59. https://sloanreview.mit.edu/article/building-the-right-ecosystem-for-innovation/.

Garud, R., Tuertscher, P., & Van de Ven, A. H. (2013). Perspectives on innovation processes. *Academy of Management Annals, 7*(1), 775–819. https://doi.org/10.1201/b18841-3.

Gawer, A., & Cusumano, M. A. (2002). *Platform leadership: How intel, microsoft, and cisco drive industry innovation*. Boston, MA: Harvard Business School Press.

Giudici, A., Reinmoeller, P., & Ravasi, D. (2017). Open-system orchestration as a relational source of sensing capabilities: Evidence from a venture association. *Academy of Management Journal, 61,* 1369–1402. https://doi.org/10.5465/amj.2015.0573.

Gulati, R. (1998). Alliances and networks. *Strategic Management Journal, 19*(4), 293–317. https://doi.org/10.1002/(SICI)1097-0266(199804)19:4%3C293:AID-SMJ982%3E3.0.CO;2-M.

Hannah, D. P., & Eisenhardt, K. M. (2018). How firms navigate cooperation and competition in nascent ecosystems. *Strategic Management Journal, 39*(12), 3163–3192. https://doi.org/10.1002/smj.2750.

Ho, H., & Ganesan, S. (2013). Does knowledge base compatibility help or hurt knowledge sharing between suppliers in competition? The role of customer participation. *Journal of Marketing, 77*(6), 91–107. https://doi.org/10.1509/jm.11.0570.

Kale, P., Singh, H., & Perlmutter, H. (2000). Learning and protection of proprietary assets in strategic alliances: Building relational capital. *Strategic Management Journal, 21*(3), 217–237. https://doi.org/10.1002/(sici)1097-0266(200003)21:3%3C217:aid-smj95%3E3.0.co;2-y.

Ketchen, D. J., Jr., Snow, C. C., & Hoover, V. L. (2004). Research on competitive dynamics: Recent accomplishments and future challenges. *Journal of Management, 30*(6), 779–804. https://doi.org/10.1016/j.jm.2004.06.002.

Khanna, T., Gulati, R., & Nohria, N. (1998). The dynamics of learning alliances: Competition, cooperation, and relative scope. *Strategic Management Journal, 19*(3), 193–210. https://doi.org/10.1002/(sici)1097-0266(199803)19:3%3C193:aid-smj949%3E3.0.co;2-c.

Lindgren, R., Eriksson, O., & Lyytinen, K. (2015). Managing identity tensions during mobile ecosystem evolution. *Journal of Information Technology, 30*(3), 229–244. https://doi.org/10.1057/jit.2015.8.

Lumineau, F., & Oliveira, N. (2018). A pluralistic perspective to overcome major blind spots in research on interorganizational relationships. *Academy of Management Annals, 12*(1), 440–465. https://doi.org/10.5465/annals.2016.0033.

Lunnan, R., & Haugland, S. A. (2008). Predicting and measuring alliance performance: A multidimensional analysis. *Strategic Management Journal, 29*(5), 545–556. https://doi.org/10.5465/annals.2016.0033.

Luo, Y. (2007). A coopetition perspective of global competition. *Journal of World Business, 42*(2), 129–144. https://doi.org/10.1016/j.jwb.2006.08.007.

Lyytinen, K., Yoo, Y., & Boland, R. J., Jr. (2016). Digital product innovation within four classes of innovation networks. *Information Systems Journal, 26*(1), 47–75. https://doi.org/10.1111/isj.12093.

Majchrzak, A., Jarvenpaa, S. L., & Bagherzadeh, M. (2014). A review of interorganizational collaboration dynamics. *Journal of Management, 0149206314563399,* xxx. https://doi.org/10.2139/ssrn.2538870.

Mantovani, A., & Ruiz-Aliseda, F. (2016). Equilibrium innovation ecosystems: The dark side of collaborating with complementors. *Management Science, 62*(2), 534–549. https://doi.org/10.1287/mnsc.2014.2140.

Miles, M. B., Huberman, A. M., & Saldaña, J. (2013). *Qualitative data analysis: A methods sourcebook* (3rd ed., Vol. 3). Thousand Oaks, CA: SAGE Publications.

Moore, J. F. (1993). Predators and prey: A new ecology of competition. *Harvard Business Review, 71*(3), 75. https://hbr.org/1993/05/predators-and-prey-a-new-ecology-of-competition.

Myers, M. D., & Newman, M. (2007). The qualitative interview in is research: Examining the craft. *Information and Organization, 17*(1), 2–26. https://doi.org/10.1016/j.infoandorg.2006.11.001.

Nambisan, S., & Sawhney, M. (2011). Orchestration processes in network-centric innovation: Evidence from the field. *Academy of Management Perspectives, 25*(3), 40–57. https://doi.org/10.5465/amp.2011.63886529.

Nambisan, S., Lyytinen, K., Majchrzak, A., & Song, M. (2017). Digital innovation management: Reinventing innovation management research in the digital world. *MIS Quarterly, 41*(1), 223–238. https://doi.org/10.25300/misq/2017/41:1.03.

Paquin, R. L., & Howard-Grenville, J. (2013). Blind dates and arranged marriages: Longitudinal processes of network orchestration. *Organization Studies, 34*(11), 1623–1653. https://doi.org/10.1177/0170840612470230.

Park, S. H., & Ungson, G. R. (2001). Interfirm rivalry and managerial complexity: A conceptual framework of alliance failure. *Organization Science, 12*(1), 37–53. https://doi.org/10.1287/orsc.12.1.37.10118.

Pathak, S. D., Wu, Z., & Johnston, D. (2014). Toward a structural view of co-opetition in supply networks. *Journal of Operations Management, 32*(5), 254–267. https://doi.org/10.1016/j.jom.2014.04.001.

Porter, M. E. (1985). *Competitive advantage: creating and sustaining superior performance*. New York, NY: The Free Press.

Powell, W. W., Koput, K. W., & Smith-Doerr, L. (1996). Interorganizational collaboration and the locus of innovation: Networks of learning in biotechnology. *Administrative Science Quarterly, 41*(1), 116–145. https://doi.org/10.2307/2393988.

Schad, J., Lewis, M. W., Raisch, S., & Smith, W. K. (2016). Paradox research in management science: Looking back to move forward. *The Academy of Management Annals, 10*(1), 5–64. https://doi.org/10.1080/19416520.2016.1162422.

Smith, W. K., & Lewis, M. W. (2011). Toward a theory of paradox: A dynamic equilibrium model of organizing. *Academy of Management Review, 36*(2), 381–403. https://doi.org/10.5465/amr.2011.59330958.

Tiwana, A., Konsynski, B., & Bush, A. A. (2010). Platform evolution: Coevolution of platform architecture, governance, and environmental dynamics. *Information Systems Research, 21*(4), 675–687. https://doi.org/10.1287/isre.1100.0323.

Tjosvold, D., & Choy, W. (1994). Working with customers: Cooperation and competition in relational marketing. *Journal of Marketing Management, 10*(4), 297–310. https://doi.org/10.1080/0267257x.1994.9964276.

Williamson, P. J., & de Meyer, A. (2012). Ecosystem advantage: How to successfully harness the power of partners. *California Management Review, 55*(1), 24–46. https://doi.org/10.1525/cmr.2012.55.1.24.

Yin, R. K. (2009). *Case study research: Design and methods*. Los Angeles, CA: SAGE Publications.

Yin, R. K. (2011). *Qualitative research from start to finish*. New York, NY: The Guilford Press.

Yoo, Y., Henfridsson, O., & Lyytinen, K. (2010). Research commentary—The new organizing logic of digital innovation: An agenda for information systems research. *Information Systems Research, 21*(4), 724–735. https://doi.org/10.1287/isre.1100.0322.

Leveraging the Value of Offshoring

Knowledge Transfer in Software Maintenance Outsourcing: The Key Roles of Software Knowledge and Guided Learning Tasks

Oliver Krancher and Jens Dibbern

Abstract Software maintenance eats up the lion's share of corporate software expenses, and many organizations attempt to reduce these costs through outsourcing and offshoring. A key challenge in these initiatives is to transfer knowledge to the new service delivery unit (a vendor or a captive center). Even though knowledge transfer plays a key role across theoretical perspectives in sourcing research (such as transaction cost economics, knowledge-based perspectives, and social perspectives), we know surprisingly little about what knowledge is most critical and through what mechanisms this knowledge is transferred in software maintenance outsourcing and offshoring. Insights from a multiple-case study of five knowledge transfers at a Swiss bank suggest that the most critical knowledge is software knowledge and that software knowledge is transferred through guided learning tasks. Software knowledge (i.e., knowledge about the application software, including its structure, functionality and behavior) is most critical because it allows engineers to cope with the cognitive burden imposed by enormous amounts of code, data, and documents. While engineers in settings of low knowledge specificity may possess sufficient software knowledge from the beginning, engineers in settings of high knowledge specificity acquire this knowledge through a series of guided learning tasks, i.e., by working on real or realistic maintenance tasks while receiving direction and task-specific information from experts. Our study adds to the emerging literature on transitions and offers important implications for the discourses on transaction cost economics and on knowledge-based perspectives in sourcing research.

O. Krancher (✉)
ITU Copenhagen, Copenhagen, Denmark
e-mail: olik@itu.dk

J. Dibbern
University of Bern, Bern, Switzerland
e-mail: jens.dibbern@iwi.unibe.ch

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_7

1 Introduction

Organizations spend up to 80% of their software budgets on software maintenance, i.e., on modifications of software after its first deployment (Banker et al. 2002; Kemerer and Slaughter 1999). Many organizations aim to reduce these costs by software maintenance outsourcing and offshoring (SMOO), i.e., by handing over the work on an organization's existing software systems to a service delivery unit (SDU), which is an external vendor in the case of outsourcing or a captive center in the case of offshoring. For this hand-over to succeed, it is important that knowledge—defined as the capacity to act in a particular context (Pentland 1992)—is transferred from the incumbent SDU to the new SDU such that the new SDU is able to take over the work. Most SMOO projects include a so-called transition phase (Tiwari 2009), during which the engineers of the incumbent SDU remain in the team to enable knowledge transfer. The viability of outsourcing and offshoring critically depends on this knowledge transfer. Indeed, case studies have shown that the knowledge transfer to the new SDU is often more difficult than expected, resulting in prolonged transition phases (Chua and Pan 2008), limited scope of work delegated (Leonardi and Bailey 2008; Zimmermann et al. 2013), and extra costs that exceed the savings from labor arbitrage (Dibbern et al. 2008). Theoretical perspectives popular in outsourcing and offshoring research also emphasize the critical role of knowledge transfer by pointing to the cognitive and social challenges of knowledge transfer (Conner and Prahalad 1996; Nahapiet and Ghoshal 1998; Rai et al. 2009; Zimmermann and Ravishankar 2014) and to the opportunistic threats that arise from these challenges (Alaghehband et al. 2011; Williamson 1981).

However, despite the widely acknowledged role of knowledge issues in outsourcing and offshoring and despite the economic significance of software maintenance, our understanding of knowledge transfer in SMOO is limited in two important ways. First, it is unclear *what knowledge* is most critical in SMOO. In line with the focus on software development (rather than maintenance) in many outsourcing and offshoring studies (e.g. Gopal and Gosain 2010; Rai et al. 2009; Tiwana and Keil 2009), much work emphasizes knowledge constructs that reflect the coordination-intensive nature of software development. These constructs include technical and business knowledge and their integration (Tiwana 2004a, b; Vlaar et al. 2008), transactive memory systems (Oshri et al. 2008), shared language (Poppo and Zenger 1998), and coordination routines (Whitaker et al. 2010). It is unclear whether these categories capture the essence of software maintenance work, which may be less coordination-intensive than software development but require intimate familiarity with existing software applications (Banker et al. 2002; Von Mayrhauser and Vans 1995). Understanding what knowledge is critical is practically important because such understanding can help managers and engineers set the right priorities in the hand-over of software maintenance work, which is among the most frequently outsourced information systems (IS) functions (Deloitte 2016). Clarity about types of knowledge is also important for research because it is difficult measure knowledge constructs, such as human asset specificity (Williamson 1981), without clarity about what the required knowledge is (Aubert et al. 2004, p. 927).

Second, it is unclear *how* the critical knowledge is transferred to the new SDU. While some case studies on SMOO focus on formal information sharing mechanisms (e.g. documents, presentations, expert directories) and report knowledge transfer durations of a few weeks (e.g. Tiwari 2009), other work emphasizes on-the-job training and reports transition durations of several years (Williams and Durst 2018). The differences between these views are important because transition plans, business cases, and the resulting sourcing decisions will look substantially different depending on which of these views organizations rely on. Greater clarity could come from longitudinal evidence that relates the use of knowledge transfer mechanisms to the performance of the new SDU over time, but such evidence is currently lacking.

In light of these two gaps, we ask: (1) *What knowledge is critical when software maintenance is handed over to a new SDU in SMOO?* (2) *How is the critical knowledge transferred?*

We address these research questions through a multiple-case study of knowledge transfers to vendors at a Swiss bank. We found that software knowledge (i.e., knowledge about the application software, including its structure, functionality and behavior) was the most critical knowledge. Without sufficient software knowledge, engineers were cognitively overburdened by enormous amounts of existing code, data, and documentation. Software knowledge allowed these engineers to make sense of the existing software systems and to solve maintenance problems. In environments of high knowledge specificity, this knowledge was transferred to the engineers primarily through a series of guided learning tasks, in which engineers worked on real software problems under the guidance of experts. Conversely, in environments of low knowledge specificity, the engineers from the new SDU possessed such knowledge from their prior experience.

Our study makes several contributions. First, going beyond on formal information sharing mechanisms in the transition literature, we show the critical role of guided learning tasks for enabling the engineers in the new SDU to take over their work. We suggest that the need to work on a series of guided learning tasks covering the whole software application explains why some transitions can take longer than planned and why clients may need to retain key personnel. Our findings also call for a greater focus on coexistence strategies in addition to information sharing issues. Second, we point out that TCE research may have failed to capture a substantial part of the variance of knowledge specificity because existing measures do not focus on the knowledge most critical for software maintenance—software knowledge—and do not capture variance at the unit of analysis of the particular software application. We also offer implications for knowledge-based perspectives in the sourcing literature.

2 Background Literature

2.1 *The IS Outsourcing and Offshoring Literature and the Key Role of Knowledge*

Over the past 25 years, a substantial body of research on IS outsourcing and offshoring has produced important findings, but also ongoing controversies, about issues such as sourcing decisions, governance of sourcing relationships, and outcomes (Alaghehband et al. 2011; Dibbern et al. 2004; Kotlarsky et al. 2018; Lacity et al. 2010, 2011; Wiener et al. 2010). Among the most influential theoretical perspectives in that research are transaction cost economics (TCE) (Ang and Straub 1998; Aubert et al. 2004), knowledge-based perspectives (Dibbern et al. 2008; Kotlarsky et al. 2014), and social perspectives (Lee and Kim 1999; Rai et al. 2009).

Interestingly, although these perspectives differ in important ways, they share a concern with the critical role of knowledge in the hand-over of work to another SDU. TCE considers knowledge as an asset that requires investment and thus gives rise to opportunistic threats. A key construct in TCE in this regard is knowledge specificity (or human asset specificity), defined as the degree to which knowledge loses its value when used for another client (Aubert et al. 2004; Dibbern et al. 2016; Williamson 1981). Knowledge specificity is high in settings where "skills [are] acquired in a learning-by-doing fashion and imperfectly transferable across [firms]" (Williamson 1981, p. 563). If tasks of high knowledge specificity are outsourced, there is a risk that vendors avoid making investments into knowledge or hold up the client once they have made in the investment. In light of these threats, TCE theorists argue that clients should not outsource work of high knowledge specificity. These ideas have received mixed empirical support (Alaghehband et al. 2011; Carter and Hodgson 2006; Lacity et al. 2011), and controversy has arisen about the reasons for this. According to the review by Lacity et al. (2011), the most frequently given reason for lack of support has been measurement problems, including measurement of the key construct of knowledge specificity (Lacity et al. 2011; Macher and Richman 2008). Empirical studies of TCE may thus benefit from understanding what knowledge is required for a particular type of task (e.g. software maintenance) and whether such knowledge needs to be acquired "in a learning-by-doing fashion" (Williamson 1981, p. 563) when work is handed over to a new SDU.

While TCE problematizes the opportunistic threats arising from difficult knowledge transfer, knowledge-based and social perspectives point to knowledge transfer challenges that arise even in absence of opportunistic behavior (Conner and Prahalad 1996; Grant 1996). A key assumption in knowledge-based perspectives is that there can be "irreducible knowledge differences between individuals" (Conner and Prahalad 1996, p. 477), in particular when individuals lack prior experience in a domain. For instance, engineers may struggle to assimilate unfamiliar information about the client's business when they lack experience with that client (Dibbern et al. 2008) or with the client's institutional environment (Nicholson and Sahay 2004). Drawing on the assumption of irreducible knowledge differences, knowledge-based theorists have made two arguments for why knowledge transfer is more difficult between than within firms. First, through their shared experience, individuals within the same organization develop shared memories, making it easier for them to share information (Blackler 1995; Hodgson 1998; Kogut and Zander 1996). Second, while managers may rely on direction as a substitute for knowledge transfer within an organization, it is argued that direction is not available in outsourcing because clients lack the authority for providing direction in contract-based relationships (Conner and Prahalad 1996, pp. 484–486). To the ideas articulated in knowledge-based perspectives, social perspectives add that knowledge transfer in complex settings requires socially embedded relationships based on shared norms and trust (Levina and Vaast 2008; Rai et al. 2009; Uzzi 1997; Zimmermann and Ravishankar 2014). Empirical evidence points to the difficulties of creating embedded relationships in outsourcing (Zimmermann et al. 2018) and offshoring (Winkler et al. 2008; Zimmermann et al. 2013) but also reveals that some outsourcing and offshoring arrangements succeed in creating them (Winkler et al. 2008; Zimmermann et al. 2013).

Two observations about this selective review of the IS sourcing literature and its underlying theories are noteworthy. First, knowledge and knowledge transfer play important roles in all three theoretical perspectives reviewed. Opportunistic threats, prior knowledge differences, authority issues, and social relationships are argued to affect the outcomes associated with particular sourcing decisions because they complicate or facilitate knowledge transfer in particular ways. A thorough understanding of the knowledge involved in sourcing arrangements and the way of how knowledge is transferred is thus critical for theory development. Second, even though the sourcing literature aims to explain decisions and outcomes at the level of inter-firm relationships, there is an emphasis on knowledge flows at the individual level across theoretical perspectives (Conner and Prahalad 1996; Rai et al. 2009; Williamson 1981). This suggests that important insights for explaining the effect of (macro-level) sourcing decisions on (macro-level) outcomes can be found by understanding (micro-level) individual-level knowledge transfer processes, in line with Coleman's bathtub model for macro-micro-macro relations (Coleman 1990; Distel 2019).

2.2 *Knowledge in SMOO*

Given the key role of knowledge across theoretical perspectives, it is not surprising that outsourcing and offshoring research has examined a variety of knowledge constructs. A substantial body of research has focused on software development (rather than maintenance). This research emphasizes two knowledge domains—business knowledge held by clients and technical knowledge held SDUs—and the structures through which these two are coordinated (Tiwana 2004a, b; Vlaar et al. 2008). In line with such a knowledge coordination perspective, studies have focused on constructs such as knowledge integration (Tiwana 2004b), transactive memory systems (Oshri et al. 2008), shared language (Poppo and Zenger 1998), and coordination routines (Whitaker et al. 2010).

While knowledge in software development outsourcing and offshoring has received substantial attention, our understanding of the knowledge involved in SMOO is far more limited, despite the fact that software maintenance accounts for much if not most of the outsourced and offshored software work (Deloitte 2016). Outsourcing researchers have speculated that knowledge in software maintenance is "generally not company-specific" (Poppo and Zenger 1998, p. 871), that it is "probably not specific" (Aubert et al. 2004, p. 929), and that "maybe [the] variance [in the specificity of this knowledge] is not sufficient to lead significant differences" (Aubert et al. 2004, p. 929). On the other hand, maintenance researchers have argued that software maintenance requires substantial knowledge of the specific business (Shaft and Vessey 1995) and substantial "software-specific knowledge" (Von Mayrhauser and Vans 1995). In line with the latter assertion, experience in maintaining the specific software has been shown to be among the strongest predictors of software maintainer performance in a study based on archival data about in-house software development (Boh et al. 2007).

Although software maintenance research points to a potentially important role of software knowledge (i.e., knowledge about the application software, including its structure, functionality and behavior), the sourcing literature has paid relatively little attention to it. In one of the most comprehensive tests of TCE in the context of IS operation activities (which include software maintenance), Aubert and colleagues correlated the degree of outsourcing of organizations with the relative importance of business and technical knowledge and with human asset specificity (Aubert et al. 2004). Their measures of human asset specificity captured, among others, the degrees to which language and work procedures were specific to the organization (Aubert et al. 1996, 2004). Counter to expectations, the results of their first survey showed a positive and thus "puzzling link between asset specificity and outsourcing" (Aubert et al. 2004, p. 929) and an insignificant link between the importance of business knowledge and the degree of outsourcing. It is noteworthy that their measurement efforts did not explicitly focus on the specificity or importance of software knowledge. Moreover, by choosing the organization as their level of analysis, their data would hardly capture variance in the domains or specificity of knowledge at the level of software systems. Yet if software knowledge is an important knowledge category, then the nature of required knowledge is likely to depend on the particular software to be maintained. For instance, the maintenance of software that has been custom-developed for a particular client may require greater client-specific learning than the maintenance of packaged software implemented at the same client. Like the seminal work by Aubert and colleagues, most other empirical examinations of TCE that included software maintenance work did not include measures specific to software knowledge (Ang and Straub 1998; Barthélemy and Geyer 2005; Diana 2009; Loebbecke and Huyskens 2006; Miranda and Kim 2006; Poppo and Zenger 2002) (Dibbern et al. 2016 is an exception). Moreover, with one exception (Dibbern et al. 2008), existing research has not captured variance in knowledge specificity at the level of software systems but focused on higher levels of analysis. These levels of analysis include the client organizations (Ang and Straub 1998; Diana 2009; Loebbecke and Huyskens 2006; Miranda and Kim 2006), IS functions with the client organization (see the second survey reported in Aubert et al. 2004; Barthélemy and Geyer 2005; Dibbern et al. 2016; Poppo and Zenger 2002), and large-scale contracts (Chen and Bharadwaj 2009). In conclusion, should software knowledge be a critical component in SMOO, this component would hardly be reflected in existing tests of TCE.

Two case studies on transitions in SMOO shed additional light on the knowledge involved in SMOO. In a case study of offshore outsourced software services, Dibbern et al. (2008) examined three cases of software maintenances services. Although their focus was not on exploring the types of knowledge involved, they found it important to account for the specificity of both business knowledge and software knowledge in their assessment of knowledge specificity. Indeed, they found "medium-high" (p. 349) extra costs for knowledge transfer in two cases in which software knowledge was important. In another case study of a large-scale software development and maintenance offshoring program, Chua and Pan (2008) studied the difficulties in the transfer of several domains of knowledge to the offshore SDU. They found that technical knowledge and application domain knowledge (in their case: banking knowledge) posed minor difficulties given that employees can bring prior technical and application domain knowledge from their prior employment relationships. In a similar vein, organizational knowledge (knowledge about the social and economic processes in the bank) was, although specific to the client, relatively easy to acquire for the engineers of the offshore SDU. Software knowledge (or, in their terms, IS application knowledge) was more difficult to transfer given that "[i]nformation overload was expected" (p. 279). The most difficult knowledge to transfer according to their analysis was, however, IS development process knowledge (knowledge about tools, techniques, methods, approaches and principles used in system development). Transferring IS development process knowledge was difficult because software knowledge and organizational knowledge were prerequisites for acquiring IS development process knowledge. While Chua and Pan provide evidence that analysis and design tasks were the most difficult to perform for the offshore engineers, it is unclear how they infer that this was due to lack of IS development process (as opposed to, for instance, software knowledge).

In sum, although existing work points to the potential relevance of various knowledge domains, it remains unclear what knowledge is critical in transitions in SMOO. Our limited understanding of the knowledge critical in SMOO transitions hampers progress in the broader sourcing literature, as our review of the TCE discourse showed. Greater clarity could come from evidence that links the acquisition of knowledge in different domains with the performance of engineers in the new SDU over time, but such evidence is currently lacking.

2.3 Knowledge Transfer in SMOO

As pointed out in our review of the broader sourcing literature, knowledge transfer is a key issue in outsourcing and offshoring. Knowledge transfer is generally defined as the process through which one entity's knowledge affects another entity's knowledge (Argote 2012). In the context of our paper, we define knowledge transfer as the process through which the knowledge held by the incumbent SDU (which may be the client and/or the incumbent vendor) affects the knowledge of the new SDU such that the new SDU is able to perform the outsourced or offshored task. While this definition is consistent with the definition in the knowledge transfer literature (Argote 2012), knowledge transfer in transitions may differ two important ways from the settings typically studied in the knowledge transfer literature. First, the knowledge transfer literature has mostly focused on the transfer of collective knowledge (Lam 2000) such as knowledge about best practices (Jensen and Szulanski 2004; Szulanski 1996). It is not clear whether the key challenge in SMOO transitions lies in the transfer of collective knowledge (e.g. software maintenance practices) or of individual knowledge, such as the maintainers' cognitive schemas of the software. As our review of the outsourcing literature showed, several scholars emphasize individual-level learning processes despite outsourcing and offshoring being firm-level phenomena (Conner and Prahalad 1996; Rai et al. 2009; Williamson 1981). Second, the recipients in the knowledge transfer literature (e.g. units wishing to learn about best practices of other units) are typically already competent in their task. They seek additional knowledge in order to improve their performance, rather than to be able to perform their task in the first place (Jensen and Szulanski 2004; Szulanski 1996). Conversely, the focus in outsourcing and offshoring transitions lies on enabling the new SDU to perform its task in the first place (Chua and Pan 2008; Tiwari 2009). Given these peculiarities, the remainder of this review will focus on the literature specific to outsourcing and offshoring transitions. This literature provides insights into transition durations, into knowledge transfer mechanisms and into the factors that complicate knowledge transfer in outsourcing and offshoring.

Transition Durations. The transition durations reported in the literature vary quite substantially between studies. In a case study of a large offshore outsourcing transition, Tiwari (2009) reported an initial knowledge transfer phase of "about one month" (p. 6) although the vendor was able to take over responsibility for the service only after seven months. Kotlarsky et al. (2014) reported on another large-scale transition program, in which the vendor planned to take over the maintenance of hundreds of applications within three months. In Chua and Pan's (2008) case study, the bank expected transition duration of between 3 and 12 months (depending on the software application) but ultimately chose to retain key onshore personnel, realizing that the new SDU was unable to reach their knowledge level. Barthélemy (2001) examined 50 outsourcing projects and found that the average transition period was one year. Although Dibbern et al. (2008) did not focus on transition durations, they quoted one informant from a maintenance project stating that one "cannot compare the knowledge of a person that has worked on the application … for 10 years with the knowledge one consultant can gain in 6 months or 1 year" (p. 353). This statement points to the long time it may take until the engineers in the new SDU have acquired sufficient knowledge to maintain the software. In line with this statement, William and Durst (2018) reported that, in their case study, it took 25 months until the vendor was able to take over more than 50% of the tasks and, within these tasks, to take over specification work rather than only coding. The differences between these durations are economically substantial given that transitions involve the presence of both the incumbent SDU and the new SDU and are, hence, substantial costs (Barthélemy 2001).

Knowledge Transfer Mechanisms. Closely related to the question of how long knowledge transfer takes is the question of through what mechanisms knowledge is transferred. Some case studies emphasize the use of *formal information sharing mechanisms*, including formal presentations and knowledge codification (Feng et al. 2010; Kotlarsky et al. 2014; Schott 2011; Tiwari 2009). Engineers in the new SDU often create and use documents to codify knowledge about software applications and about expertise location (Kotlarsky et al. 2014). Some vendors advertise their formal information sharing methods as a means for ensuring that knowledge is transferred in short time and thus not lost during transition (Cognizant 2002). On the other hand case studies show that engineers often struggle to assimilate the information presented through formal methods (Chua and Pan 2008; Dibbern et al. 2008) and that knowledge codified in documents may not help when problems are complex (Kotlarsky et al. 2014). Despite these limitations, formal information sharing mechanisms are often a key element of transition plans, in particular during the first few weeks (Chua and Pan 2008; Kotlarsky et al. 2014; Tiwari 2009).

Other studies focus on *socialization mechanisms* such as site visits, teleconferencing, instant messaging, and tight embedment of SDU engineers with the client organization (Gregory et al. 2013; Levina and Vaast 2008; Oshri et al. 2008; Wende et al. 2010; Williams 2011; Zimmermann and Ravishankar 2014). These studies reveal benefits from socialization such as allowing to rapidly clarify queries, promoting transactive memory, and helping improve the client's and the new SDU's understanding of each other. It is noteworthy that socialization mechanisms are primarily described in studies of software development (rather than software maintenance), indicating that they may be particularly useful for addressing the high need for ongoing coordination between client and SDU in software development.

Case studies of SMOO also describe *practice activities*, such as quizzes, support simulation (i.e., engineers of the new SDU work on problems that occurred in the past), and on-the job training (i.e., engineers of the new SDU work on actual tasks or observe experts working on actual tasks) (Chua and Pan 2008; Feng et al. 2010; Schott 2011; Williams and Durst 2018). In Chua and Pan's case study, these practice activities helped, in particular, to transfer IS development process knowledge such as problem-solving methods and communication about open maintenance issues. Williams and Durst (2018) describe a case study where the vendor was able to take over the majority of work only after working on tasks for more than two years, pointing thus to the long time that practice activities may take until performance levels are satisfactory.

Although the literature reveals a variety of knowledge transfer mechanisms used in transitions, it remains unclear which of these mechanisms enable engineers in the new SDU to take over software maintenance work. Greater clarity could come from longitudinal evidence that relates the use of particular knowledge transfer mechanisms to the performance of the new SDU over time, but such evidence is currently lacking.

Factors Complicating Knowledge Transfer in Outsourcing and Offshoring. A number of studies provide insights into the factors that complicate knowledge transfer in outsourcing and offshoring settings. Understanding these factors is important because they help understand how knowledge transfer in transitions can be different from knowledge transfer in other settings such as internal employee succession. *Low absorptive capacity* of the engineers in the new SDU (i.e., low ability to assimilate outside information) often complicates knowledge transfer because the engineers in the new SDU often lack prior experience with the client and therefore struggle to process the information presented to them during knowledge transfer (Chua and Pan 2008; Dibbern et al. 2008). *Limited availability of experts* from the incumbent SDU is often a problem given that planned transition durations are often short and experts have the double role of supporting knowledge transfer and providing the maintenance service during this short time (Beulen et al. 2011; Tiwari 2009; Williams and Durst 2018). *Distance* of various types (cultural, geographic, temporal, status) also hampers knowledge transfer because it makes it more difficult for the engineers in the new SDU to interact with the engineers of the incumbent SDU (Dibbern et al. 2008; Imsland and Sahay 2005; Levina and Vaast 2008; Wende et al. 2013). In a similar vein, weak social relationships may hamper interaction and thus knowledge transfer (Williams 2011; Zimmermann and Ravishankar 2014). It is interesting to note that these factors are reported both in captive (i.e., in-house) and in outsourced settings, suggesting that they emanate primarily from the nature of transitions (i.e., handing over work in short time despite complicating factors such as low absorptive capacity, distance, and weak social relationships) rather than from inter-firm boundaries.

Taken together, existing research on knowledge transfer in SMOO focusses on a variety of knowledge domains, knowledge transfer mechanisms, and challenges. Yet, it remains unclear which of these knowledge domains are critical in SMOO and how knowledge in these domains is transferred to the new SDU such that the new SDU can take over the service.

3 Methods

We conducted a multiple-case study of software maintenance outsourcing transitions. The case study method promised insight into the importance of knowledge domains in real-life settings across a set of projects (Yin 2009). It was also well suited for collecting and analyzing longitudinal data (Langley 1999; Yin 2009), which allowed capturing the use of knowledge transfer mechanisms and the resulting outcomes over time.

3.1 Sampling

Our unit of analysis was the knowledge transfer for a particular software application. We selected five such knowledge transfers. All five cases stemmed from the same globally operating bank located in Switzerland. At the time of the study, the bank held assets of over $1 trillion, had considerable experience in SMOO to Indian vendors, and relied on offshore outsourcing to reduce maintenance costs. The bank collaborated with three major Indian vendors, which we refer to as A, B, and C. The bank granted us permission to accompany transitions, which presented a unique opportunity for collecting real-time data (Langley 1999) that would grant rich insights into the knowledge transfer process. In all five cases, the bank outsourced software maintenance work to one of the three Indian vendors although, in each case, at least one on-site coordinator was permanently present at the client site. Our analysis focused on the knowledge transfer to on-site coordinators because on-site coordinators play a key role in sourcing arrangements (Tiwari 2009; Williams and Durst 2018).

Table 1 provides an overview of the cases. Although our case selection was not guided by a priori hypotheses, we made sure to include both cases of high and of low knowledge specificity, given the key role of this construct in TCE and knowledge-based research (Aubert et al. 2004; Dibbern et al. 2008). The transitions of high knowledge specificity referred to the maintenance of custom-developed software used exclusively by the client, whereas the transitions of low knowledge specificity referred to the maintenance of software packages installed at the client and at other firms. The cases also differed in the nature of the incumbent SDU. Case 1 and case 4 involved knowledge transfer from the client (i.e., the incumbent SDU) to a vendor. Case 2 and case 3 involved knowledge transfer from one vendor (vendor C) to another vendor (vendor A), who was supposed to take over maintenance. Case 5 was a special case in that it involved the knowledge transfer within a vendor (vendor B), where the on-site coordinator needed to be replaced by another on-site coordinator because his work permit was about to expire after six years in Switzerland. We included case 5 because it promised insights into the question of whether critical knowledge is SMOO is primarily located at the individual or at an organizational level. Although the incumbent SDU was a vendor in the cases 2, 3, and 5, the engineers from the new SDU also interacted with retained experts from the client organization (see Table 1).

Table 1 Cases

Case	Knowledge specificity	New SDU Engineer(s)	Experts from incumbent SDU and/or client	Process duration captured by data
1	High: custom-developed executive information system producing profitability key figures	"Anand": 4 years of experience in PL-SQL and in software projects, vendor A	Four Swiss or Germans (client)	5 months
2	High: custom-developed data marts supporting internal accounting	6 years of experience in data marts and 11 years in software projects, vendor A	One Indian (main contact, vendor C), one Swiss (client)	3 months
3	High: custom-developed data-warehousing systems predicting customer credit risk	5 years of experience in data warehousing, 11 years in software projects, vendor A	Two Indians (main contacts, vendor C), one Swiss (client)	5 months
4	Low: Software package SunGard Intellimatch, allowing reconciliation of financial transactions	Two engineers: – "Raj": 4 years of experience in Intellimatch and in software projects, vendor B – "Satish": 1 year of experience in Intellimatch, 5 years in software projects, vendor B	Three Swiss (client)	5 months for engineer 1 and 3 months for engineer 2
5	High: custom-developed application for tax rate administration	"Sai": 10 years of experience in mainframe technology and software projects, vendor B	One Indian ("Pratap", vendor B, initial main contact), two Swiss (client)	6 months

As can be seen from the third column in Table 1 all engineers from the new SDU had substantial experience in the technologies relevant for their task. This can be expected given that clients often choose vendors due to the vendors' technical knowledge in the task domain (Aubert et al. 2004; Tiwana 2004a)

3.2 Data Collection

We collected data through semi-structured interviews, observation, and document analysis based on a case-study protocol (Yin 2009, p. 79). Table 2 gives an overview of the data sources. We conducted 38 interviews with client managers and with engineers from the new vendors, from the incumbent vendor, and from the client. The interviewees were asked to exhaustively report on transition activities, such as the work on particular tasks and the exchange of information, and on the knowledge domains involved in these activities. Moreover, we elicited information on task performance and task complexity from experts and managers. We also asked engineers about their assessment of the relative importance of different knowledge domains, using the IS body of knowledge taxonomy (Iivari et al. 2004), which is also used in Chua and Pan's (2008) case study. Appendix 1 shows the interview guideline. To reduce memory effects, we conducted several interviews with the same informants at different points in time. We triangulated information from different sources and followed up by email or subsequent interviews. All interviews were recorded and transcribed. We also observed formal sessions that were intended to provide the vendor engineers with introductory information about the software. The sessions helped us to familiarize ourselves with knowledge domains, which was useful to code knowledge domains. Moreover, we studied documents, such as requirements and design specifications, transition plans, and software documentation. The documents allowed for triangulation but also offered complementary information on tasks and the related knowledge domains. For instance, design documents revealed information about the amount of guidance provided through documents.

Table 2 Data sources

Case	No. of interviews/No. of interviewees[a]			No. of observed sessions	No. of documents
	Vendor engineer	Subject matter experts	Manager		
1	5/1	2/2	2/1	4	20
2	2/1	2/2	4/4[b]	2	8
3	3/1	2/2		2	16
4	4/2	2/2	1/1	2	2
5	2/1	4/3	3/3	2	4

[a] 5/1 denotes that five interviews were conducted with the same one interviewee. 2/2 denotes that a total of two interviews were conducted with two interviewees, i.e. each interviewee was interviewed once

[b] Although case 2 and 3 referred to different software applications and involved the knowledge transfer to different vendor engineers, they involved the same managers

3.3 Data Analysis

We began our analysis by coding knowledge domains and knowledge transfer activities in NVivo. To capture knowledge domains, we coded knowledge subdomains such as knowledge on a particular software component or a particular technical skill. We coded a statement to a particular knowledge subdomain whenever a statement indicated a need for this knowledge. We then aggregated these subdomains to the five areas of the IS body of knowledge (Chua and Pan 2008; Iivari et al. 2004). Since we found it difficult to separate organizational knowledge from application domain knowledge, we aggregated these two to the category business knowledge, resulting in overall four knowledge domains. This coding process resulted in 1320 coding instances, referring to 357 knowledge subdomains. We also coded knowledge transfer activities, which yielded a variety of inductive codes describing the activities during knowledge transfer such as knowledge elicitation sessions and the work on learning tasks (i.e., real or realistic maintenance tasks). As our analysis proceeded, learning tasks turned out to play an important role. We therefore systematically coded all learning tasks and the knowledge subdomains involved in each of the learning tasks, using a matrix that related learning tasks to knowledge domains. Extracts of the matrix were validated by participants. We also graphically depicted all knowledge transfer activities by creating one visual map (or time-order display) (Langley 1999; Miles et al. 2014) per case, showing the knowledge transfer activities over a timeline.

In line with the principle of triangulation of data analysis (Yin 2009), we then relied on three analysis strategies to answer our first research question. First, we explored why the vendor engineers were, at some point, able to take over tasks (i.e., independently and successfully work on a learning task). We compared those learning tasks which vendor engineers were able to perform to those which vendor engineers were unable to perform and examined the involved knowledge domains. In this analysis, we also considered the explanations provided by the engineers about why they were or were not able to perform a particular learning task. Second, we counted the number of coding statements for each knowledge domain, assuming that the more important a knowledge domain is, the more frequently it would be coded. Third, we built a regression model that regressed task performance on the experience that vendor engineers have gained within the four knowledge domains (software, technical, business, IS development process) and on a number of further variables that resulted from the analysis below. We coded task performance as a dichotomous variable that was 1 when a vendor engineer was able to perform a task and otherwise 0. Consistent with learning curve research (Boh et al. 2007), we used logarithms of the numbers of prior learning tasks in a subdomain and averaged these values for all subdomains that were relevant for the focal learning task.

To answer our second research question of what knowledge transfer activities were critical, we compared those learning tasks which vendor engineers were able to perform to those which they were not able to perform, focusing this time on the type of knowledge transfer mechanisms. For instance, we examined whether particular formal information sharing mechanisms or learning tasks were able to explain changes in performance. This analysis led us to discover the key concept of guided learning tasks, i.e., learning tasks accompanied by direction and by task-specific information. To corroborate our contention that guided learning tasks were the critical mechanism, we added direction, task-specific information, and generic information as control variables to the regression model that we also used for the first research question. We coded task-specific information and generic information based on the amount of time that vendor engineers spent in particular information sharing mechanisms in relation to a particular learning task. We also added the control variable of task complexity to this analysis, coded in a three-point scale that captured the extent to which the information involved in the task was mutually related. To assess the reliability of the coding procedure, a second coder coded task complexity, task performance, and information sharing activities, yielding satisfactory inter-coder reliability.

4 Results

Before we present our analyses regarding what knowledge was critical and how critical knowledge was transferred, we provide narratives of three cases (case 1, case 4, case 5). We focus, in our narratives, on these three cases because the events in case 2 and case 3 were relatively similar to case 1.

4.1 Case Narratives

Case 1. In case 1, the bank aimed to augment their internal maintenance team responsible for a custom-developed executive information system with personnel from vendor A. The bank requested CVs of suitable candidates from vendor A, conducted phone interviews, and selected Anand, who had four years of experience in PL-SQL, the programming language in which the system was built. During the first two weeks, the client manager arranged three knowledge elicitation sessions that involved Anand, a knowledge transfer coach, and one of the client's experts. These sessions served to draw a conceptual map that should provide Anand with an overview of the software application, its components, the dataflow between these components, and people in the organization responsible for the components. The conversation largely followed the pattern of the coach asking questions, the expert answering the questions, and the coach drawing the map based on the information provided. Although the coach encouraged the Anand several times to ask questions, Anand participated in the discourse very sporadically. Outside these sessions, Anand primarily studied documents and code. He found it difficult to understand the information in the documents and in the source code:

> I went through the document and I did not understand anything. Most of the things I could not understand. Whichever is easy to understand, I did not do that.
>
> As such, just code study is pretty tough. Especially here where implementation is quite complex… It is complex because you don't know what the code does and because you lack an overview. And if you go into the code without an overview, that's very difficult.

After two weeks, Anand began working on a series of maintenance tasks. When working on these tasks, Anand felt like a novice, being confronted with the unfamiliar software application and unknown business concepts:

> At first, you do not have any knowledge. It is like a layman.

The manager and the client's experts were not surprised that Anand—the carefully selected, experienced engineer from a respected vendor—depended on their help:

> It is quite rare to see any work results within the first three months. (Client expert 1, case 1)
>
> He has been here for 2.5 or 3 months or so. You cannot expect that he is able to do anything at that point. (Client expert 2, case 1)

Anticipating Anand's need for help, they initially gave him rather simple tasks, provided detailed solution steps, and answered Anand's questions such as:

> What data is stored in these tables?… Why are we doing these queries? … How is the data loaded?

As Anand continued to work on maintenance tasks, he became increasingly competent within those areas of the software application in which he has worked:

> This kind of task, I have already done several times because it is a [XYZ] task. I have worked on [XYZ] tasks so far. So I was sure about what steps need to be done. (Anand)

After three months, Anand took a more active role and asked his colleagues to delegate tasks to him when he had spare capacity, which allowed him to *"learn these database views, which are quite important" (Anand)*. Meanwhile, he was also able to understand those documents that referred to familiar areas of the software application. After having worked on 14 maintenance tasks during the first five months, the client's manager was *"very satisfied with his level of knowledge"* and assessed that he was now able to independently work on maintenance requests in several areas of the software application, notably those areas on which he had worked thus far.

Case 4. Like in case 1, the bank aimed to augment their in-house software maintenance team with personnel from a vendor, in this case vendor B. Raj, the first vendor engineer that joined the client's team, experienced a transition that was very different from Anand's case (case 1). Like Anand, he had four years of specialized experience. But in contrast to Anand, he was the protagonist in a case of packaged software maintenance. All informants viewed Raj as an expert from the beginning:

> I was already knowledgeable about the product and everything. (Raj)
>
> He knew everything. This is why he got started so quickly. We just had to present the application to him. … Then everything went fast. (Expert 1, case 4)
>
> He knows the tool very well, almost better than we do. (Expert 2, case 4)
>
> He is highly skilled. (Manager, case 4)

When the client provided him with information on client-specific aspects, the participants found this helpful but not necessary for him to solve tasks:

> Even before I got the knowledge transfer, I started working. (Raj)
>
> "Why did [Raj] need information on these scripts?" (Interviewer) – "The scripts are somewhat hidden on the server. You simply need to know where the stuff is. He would have discovered this on his own, but it would have taken more time. With his know-how, he would discover virtually anything on his own." (Expert 2, case 4)

2.5 months after Raj joined the team, Satish (another engineer from vendor B) joined the same team as an on-site resource. Like Raj, he found it easy to assimilate information about client-specific aspects of the set-up of the application.

Given their expertise in the software application, both Raj and Satish were able to successfully perform all 14 maintenance tasks that were assigned to them within their first five (Raj) and three (Satish) months, respectively. However, a primary challenge for them was that the client engineers initially hesitated to delegate maintenance task to vendor engineers. They started delegating more work only after observed the outcomes of the vendor engineers' work:

> Slowly you noticed that you can give him tasks. After you saw that [Raj] did his first task well and that this did not consume much of [the client's experts'] time, they noted: Ah, there is someone to whom you can give work. (Client manager, case 4)
>
> Admittedly, [Raj and Satish] are doing a good job. (Expert 2, case 4)

Case 5. Case 5 is different from the other cases in that it was a within-vendor knowledge transfer. Vendor B had been responsible for a tax application of the bank for several years. The team of vendor B comprised the on-site coordinator Pratap and an offshore team. The client staff's perceptions of the cooperation with the vendor were at best mixed:

> *Whenever there were problems, people said [vendor B] is incapable. But I doubt that this is true. (Client manager 1, case 5)*

Before Pratap's work permit was about to expire, a manager of the bank arranged two knowledge elicitation sessions to improve Pratap's business understanding of the tax application. The sessions involved the same coach as in case 1, Pratap, and an engineer from a local service provider, who had in-depth knowledge about the tax application due to his involvement during the development of the system. The key outcome of the sessions was a conceptual map showing the data flow between application and between departments and organizations.

Two months after the knowledge elicitation sessions, the vendor presented Pratap's successor Sai to the client. Sai had 10 years of experience with mainframe systems, the technology domain of case 5. During the month that followed, Sai was still in

India and had three weekly phone conferences with Pratap. After that month, Sai had six weeks during which both he and Pratap were at the client site. During that period, Sai had almost daily knowledge sharing sessions with Pratap based on a knowledge transfer plan created by Pratap, which comprised different areas of the tax application as well as IS development process information such as the handling of maintenance requests. Pratap also handed over the work on one maintenance request to Sai and answered Sai's questions around that request.

After Pratap left Switzerland, a client engineer, who had been involved in the development of the tax application and who was responsible for several related applications, became the primary point of contact for Rai. When that client engineer noted problems in an important data load and asked the Sai to solve them, "*it took a while and nothing happened" (Client expert).* What followed was an iterative process in which the Sai presented an incomplete solution, the client engineer added elements to the solution, and Sai implemented the solution. In Sai's view, the problem was *"not that much of simple because a lot of business rules"*, *"many tables"* and *"other factors"* needed to be considered. In contrast, the client engineer said he *"would have solved the task in half an hour"*. Frustrated by this experience, several client engineers decided to solve maintenance requests in the tax application themselves, rather having them processed by Sai:

> I am not aware that Sai is actively working on tickets. Our internal employees assess for each ticket whether [vendor B] would be able to solve it. Most tickets are solved by us because the problems are too complex… This is reasonable given that the software requires very good knowledge in order to perform analyses and corrections. Moreover, the tickets need to be solved quickly. And this is currently not possible with [vendor B]. (Client manager 2, case 5)

While several client engineers and managers were dissatisfied with Sai's knowledge level, they did not see it in their responsibility to provide Sai with help:

> In the cases where we asked Sai to do the analysis, this was by no means successful. At the end, he stood next to our expert every five minutes and he had to explain each and everything. (Client manager 2)

> We have the agreement with [vendor B] that they deliver irrespective of the people they are sending. (Client manager 2)

Even though several client employees were dissatisfied with Sai's knowledge level, Sai was subsequently able to successfully deliver a relatively comprehensive software enhancement in which he collaborated with another client expert:

> With this one I was more comfortable because I already worked on the application for a couple of months. Moreover, I was involved [in that software enhancement] from the beginning, from requirements gathering to solution design. I have been part of all these meetings. (Sai)

Despite this success, client management decided not to extend the purchase order for Sai, suggesting that the client saw the knowledge transfer to Sai as a failure.

4.2 What Knowledge Was Critical

Software Knowledge. Informants from all five cases emphasize the key role of *software knowledge* (or application knowledge). In particular the four cases of custom-developed software (i.e., high knowledge specificity; cases 1, 2, 3, and 5) abound with statements of difficult software knowledge acquisition. The vendor engineers in these cases often depended on help on application issues even later during transition, when knowledge gaps in other domains were not very salient any more. For instance, when Anand (case 1) was assigned a maintenance request during his third month at the client, he struggled with the request given his lack of experience in this particular software component:

> This task is very low-level. It is about a particular table and how it gets loaded." (Anand) – "If you had plenty of time and nobody was available to help you, could you solve the task by looking up information in the source code?" (Interviewer) – "That would take a lot more time because coding is very complex. Doing that alone is very tough – almost impossible I would say without any help." (Anand)

The statement shows why software knowledge was so central for the task. Although Anand did not require significant business knowledge for the task ("*very low-level*"), although he impressed clients experts with his technical knowledge, and although he had access to the source code of the software, he found it *"almost impossible"* to comprehend the software behavior that results from the source code. Having little experience in the part of the software relevant for request, he apparently lacked the cognitive schemas that would allow him make sense of the large amounts of source code. Like in this case, informants from other cases of custom-developed software maintenance agreed that it was most often the lack of software knowledge that most constrained the vendor engineers' performance:

> [When you are experienced in a system and something happens,] you know where this can happen because you know the application. Suppose you get data here and you know these data are coming from that table, and that table is also used in another application. […] For [the vendor engineer] it will be a little bit difficult in the beginning. (Expert, case 2)

> Application knowledge is vast. (Expert, case 3)

> [Software knowledge] is the main thing to gain (Vendor engineer, case 3)

> [When I worked on this task,] I had a lot of questions: Why something is there? Why do we get the data through this interface? Why are we feeding the data to this interface? (Sai, case 6)

Code comprehension was less an issue for the two vendor engineers in case 4, which referred to packaged software (low knowledge specificity). In this case, Raj and Satish could use their software knowledge from prior projects to comprehend the client-specific aspects of the software:

> I know how [the software package] works, but then set-ups can slightly change like pre-processing and post-processing [of data]. […] That knowledge is what you have to gather. (Raj, case 4)

> My mind could easily map what the difference [to other implementations of the same software package] is. [… When I had first worked on this software package,] it had been like working on a blank sheet of paper. […] But if I have been through something, it always stays in the memory. My subconscious always has some images which never get lost. (Satish, case 4)

The statements suggest that these engineers could use their cognitive schemas from prior projects with the same software package to make sense of information about client-specific aspects of the software implementation. Unlike in custom-developed software maintenance, engineers may thus draw on their prior software knowledge to understand more easily the client-specific aspects of the set-up. The concept of software knowledge is thus key for explaining the difference between cases of custom-developed and of packaged software. In the cases of packaged software, engineers could make much greater use of their knowledge from prior projects, indicating that they were lower in knowledge specificity than cases of custom-developed software.

Technical Knowledge. All vendor engineers had substantial prior experience in the technology domains of their tasks such as data warehousing or mainframe programming. They were considered as technically highly proficient by the client's or incumbent vendor's experts such as in this statement:

> [The vendor engineer] has a very firm grasp of all the technologies involved. (Incumbent vendor expert, case 2)

Very few statements indicated situations where vendor engineers were dependent on help on technological issues. Hence, increases in technical knowledge cannot explain why vendor engineers are, at some point, able to take over work.

IS Development Process Knowledge. In all cases, the vendor engineers needed to learn about the client's software maintenance processes, such as document templates, tools, and deployment procedures:

> [The VE had to learn] what to do to properly process a business request. There is a realization specification. There are templates. Where are they? What to write where? […] Then testing. How to fill the tool with life? What is the process? (Expert 2, case 1)

Although vendor engineers needed to learn about processes, they did not perceive this knowledge as difficult. Moreover, the experts soon attested that the vendor engineers had acquired sufficient process knowledge when vendor engineers still struggled with software knowledge. For instance, while Anand (case 1) was still cognitively overstrained by the complexity of the software in month 3, he commented:

> Process things are now clear to me.

The vendor engineers and experts in the other cases also perceived IS development process knowledge as less difficult:

> [The process is] clear, there is no confusion. Now I know where to go, what to do. (Vendor engineer, case 2)

> I don't have much difficulty in grasping these tools and processes. (Vendor engineer, case 3)

> Process knowledge is always a little bit simpler. (Expert, case 3)

Even the client expert that was very critical about Sai's level knowledge (case 5), attested that the problem does not lie in IS development process knowledge:

> He needs to know where to test what and that there are requirements documents and solution descriptions. […] He now knows all this. (Expert, case 5)

Business knowledge. Because we found it difficult to separate application domain knowledge from organizational knowledge (Chua and Pan 2008; Iivari et al. 2004), we subsume these two under the construct of business knowledge (i.e., knowledge about the application domain and the social and economic processes that the software supports at the client). Although business knowledge concepts (e.g. securities, tax rules, account reconciliation processes at the bank) appeared in all cases, the engineers reported that business knowledge was not among the most critical knowledge domains for their work. It appears that the demands for business knowledge were limited because requirements engineers of the client were present in four of the five cases, as explained by the vendor engineer in case 3:

> Here, we have a requirements engineer. He is facilitating with the business requirements. He is the translator from business terms to technical terms. Therefore we don't need any specific banking knowledge. (VE, case 3)

Counts of Coding Instances. While the interview statements suggest that software knowledge was most critical for the engineers to acquire during transition, we triangulated this finding through two further analyses: counts of coding instances and regression models predicting task performance (see the regression models in the subsequent section). Table 3 shows the counts of coding instances. A statement was coded to a particular knowledge category whenever it suggested the need for a particular knowledge domain. The figures reflect thus the degree to which different knowledge categories were salient in the work of the vendor engineers. In all cases, software knowledge was the most frequently coded knowledge domain, with percentages ranging from 48 to 76%. Business knowledge was the second most frequently coded knowledge domain in all cases with percentages ranging from 13 to 30%. Technical knowledge and IS development process knowledge were coded least frequently, with percentages ranging from 4 to 13% for technical knowledge and 5 to 14% for IS development process knowledge. These figures are in line with our informants' perception that software knowledge was the most critical knowledge for

Table 3 Counts of coding instances

Case	Software knowledge	Technical knowledge	IS development process knowledge	Business Knowledge
1	300 (76%)	16 (4%)	28 (7%)	50 (13%)
2	106 (48%)	26 (12%)	32 (14%)	59 (26%)
3	207 (64%)	17 (5%)	45 (14%)	53 (16%)
4	75 (55%)	11 (8%)	10 (7%)	41 (30%)
5	133 (55%)	31 (13%)	13 (5%)	67 (27%)

them to acquire during transitions. We will further triangulate this assertion in the regression models predicting performance in the next section.

5 How Critical Knowledge Was Transferred

Although the vendor engineers participated in a variety of activities aimed at knowledge transfer and although all these activities may have contributed to their learning process in some way, our interest was to examine which activities were most important for enabling vendor engineers to take over. We define *taking over* as the independent and successful completion of the work that the client has delegated to the vendor through outsourcing. It is clear from the case narratives that the prerequisites for taking over differed substantially based on knowledge specificity. The two engineers in case of packaged-software maintenance (case 4, *low knowledge specificity*) were able to take over work right from the start by drawing on their software knowledge from prior projects. Although these engineers also engaged in knowledge transfer activities such as formal presentations and document study, these activities were helpful but not necessary for them to take over (*"Even before I got the knowledge transfer, I started working"*, Raj). Hence, it does not appear that any particular knowledge transfer activities were critical for allowing these engineers to take over.

Limits of Formal Information Sharing Mechanisms. Conversely, the engineers in the cases of custom-developed software maintenance (cases 1, 2, 3, 5, *high knowledge specificity*) were not able to take over at the beginning of transitions. In all these cases, the engineers engaged in substantial *information sharing* mechanisms such as formal presentations, knowledge elicitation sessions, and document study. However, they were not able to independently work on maintenance tasks even after they had spent substantial time in information sharing mechanisms. For instance, at the time when Anand (case 1) struggled to independently work on the maintenance request at the beginning of his third month, he had previously participated in three induction sessions and in three knowledge elicitation sessions and he had spent *"about two hours"* (Anand) a day reading documents during the first six weeks. Although the induction sessions, the knowledge elication sessions, and the documents had covered the area of the application that this particular maintenance request was about, it was the first time Anand had worked on a task in this particular area:

> [I need help on implementation and testing for this task] because it is a totally new thing which I had not worked on before. (Anand)

According to one of the client's experts, one reason for Anand's need for help was that the task required more detailed knowledge that can be covered in formal sessions:

> "What knowledge would Anand have required for this task?" (Interviewer) – "We are very much metadata-based and these metadata are sometimes quite complex… We have over 100 metadata tables with sometimes more than 1,000,000 entries. This particular task involved

> perhaps 20 metadata tables. Just explain 20 metadata tables with 10,000 entries each in a knowledge transfer session (laughing). There must be gaps. That's hard to avoid. Even if you explain: 'You have to know these 10 tables'" (Expert 1, case 1)

Another important limitation of formal information sharing mechanisms was that when information was presented to the new vendor engineer without being linked to a particular task, it was difficult for the engineers to understand the information. This is indicated by Anand's struggles to make sense of documents during the first weeks (see the quote in the case narrative) and by the expert's advice not to delve too deep into documents without working on a particular task:

> If you read this [document] without a specific topic [i.e., a concrete task], you won't understand the document. It is impossible to convey the content of this document without a specific topic. (Expert 1, case 1)

Guided Learning Tasks. While formal information sharing mechanisms did not enable vendor engineers to take over, our analysis points to a mechanism that can explain how and when vendor engineers are able to take over work: guided learning tasks. *Learning tasks* are real or realistic tasks, such as particular software enhancements or defect corrections. *Guided learning tasks* are learning tasks that are accompanied by direction and by task-specific information. Direction means that an expert provides the engineer with solution steps or by points to a similar task that the vendor engineer can imitate. Task-specific information means that an expert or a document provides information that has been created or tailored for this specific task. We uncovered the critical role of guided learning tasks by identifying all tasks on which the vendor engineers worked on during the transitions and by examining under which conditions vendor engineers were able to independently complete them. Within the four cases of custom-developed software, we identified 51 learning tasks. Among these 51 learning tasks, we identified 11 learning tasks in which no direction was provided to the vendor engineers by experts. These learning tasks provided the opportunity to observe whether the vendor engineers were able to solve them independently (i.e., without direction). The vendor engineers were able to successfully complete 5 of these 11 learning tasks. In each of these 5 learning tasks, the vendor engineer had previously worked on a guided learning task in the same area of the software as the focal task. In other words, vendor engineers were able to take over work only when they had previously worked on a learning task in the same area of the software with help from an expert. This logic is also echoed in several interview statements such as:

> He is now able to do what he has done before. (Client expert 1, case 1)

It appears that guided learning tasks were effective because of two of their qualities. First, given that they referred to real or realistic tasks, they involved practice opportunities and were less abstract than documents that are not tied to a particular maintenance request. Second, although they presented engineers with real or realistic and thus complex tasks, the cognitive burden on maintainers was reduced by providing direction and task-specific information. To illustrate, consider the first learning task for the vendor engineer in case 3. In this case, the expert from the incumbent vendor provided direction through a detailed design document, a document that normally the vendor engineers would be responsible for creating themselves. The solution steps pointed out in the design helped reduce the cognitive burden on the vendor engineers while giving him the opportunity learn based on a real task:

> For that release, I kept two change requests open for [the vendor engineer]…. We from [vendor C] had already done the design because I didn't want to give [the vendor engineer] the full design the first go. I told him to review it, so he reviewed it, things were fine. Then he started to implement the design… Once he started implementing, the entire process comes into place like where to get the design document, the review document … how he should prepare, how he should deploy. (Expert, case 3)

> For this change request, the design was already given to you, right? Did that simplify the work in comparison to a situation in which you would have to do the design on your own?" (Interviewer) – "Yes, definitively. (Vendor engineer, case 3)

Although the direction provided through the design document was helpful (the direction component of guided learning tasks), the vendor engineer also required additional explanations from the expert (the task-specific information component of guided learning tasks):

> When you were working on this change request, did you trigger some explanations from [the expert]?" (Interviewer) – "Yes, I was open to ask any queries to him." (Vendor engineer, case 3) … – "How much input did you require?" (Interviewer) – "In the beginning, there was much input required. … I asked many doubts in the design … how the design was to be read and how implement this code. (Vendor engineer, case 3)

A Regression Model Predicting Task Performance. Our analyses presented in the previous sections suggest that software knowledge was the most critical knowledge domain and that guided learning tasks were the key mechanisms for transferring this knowledge in the cases of high knowledge specificity. To corroborate these two assertions, we estimated a probit regression model that predicted task performance, coded as a binary variable (1 if the engineer was able to perform the task successfully, otherwise 0). The level of analysis of the model was the learning task. The sample includes all learning tasks from the cases of low knowledge specificity (i.e., case 1, 2, 3, and 5) because task performance did not vary in case 4 (i.e., the engineers were able to perform all tasks). We present the results of two models, one with dichotomous control variables for the cases (model 2) and one without (model 1). Table 4 shows the results. The model includes predictors for the vendor engineers' experience within the domains relevant for the focal learning task. It distinguishes experience in four distinct domains (software, technical, IS development process, business). Experience stems from the work on learning tasks prior to the focal task and from prior projects. Moreover, the models include the two components of guided learning tasks (direction and task-specific information) and the two control variables task complexity and generic information (i.e., information that has not been created or tailored for the focal task).

Table 4 Regression results

Parameter	Model 1	Model 2
Intercept	−0.73 (0.65)	−1.07 (0.80)
Software experience	1.38** (0.53)	1.20* (0.61)
Technical experience	0.64 (0.39)	0.79 (0.64)
IS development experience	0.87 (0.55)	0.68 (0.51)
Business experience	−0.29 (0.32)	0.06 (0.58)
Task complexity	$-0.52^{\dagger}$ (0.30)	$-0.56^{\dagger}$ (0.31)
Direction	$1.51^{\dagger}$ (0.79)	1.37 (0.84)
Task-specific information	$0.94^{\dagger}$ (0.48)	$1.22^{\dagger}$ (0.69)
Generic information	−0.17 (0.43)	-0.37 (0.56)
Case 2	–	1.13 (1.53)
Case 3	–	1.14 (0.1.46)
Case 5	–	0.11 (1.19)
Likelihood ratio Chi-Square test (df)	34.69*** (8)	35.60*** (11)

$^{\dagger}p < 0.10$, $*p < 0.05$, $**p < 0.01$, $***p < 0.001$, $n = 51$, standard error in parentheses, dependent variable: task performance, all variables have been standardized expect for the dichotomous variables direction and the case control variables

In both models, software experience was the most significant predictor of task performance ($p < 0.01$ in model 1, $p < 0.05$ in model 2). Conversely, the coefficients related to technical experience, IS development experience, and business experience were insignificant in both models. These findings highlight the critical role of software knowledge. The findings on software experience also shed light on the role of learning tasks. Given that software experience captures the number of prior learning tasks within the domain of the focal task, the strong positive coefficient related to software experience also show that task performance strongly depends on the number of prior learning tasks that are related to the focal task. This is consistent with our assertion that vendor engineers were only able to take over work when they had previously worked on learning tasks in domains related to the focal task.

The coefficients related to direction and information shed light on the "guided" aspect of learning tasks. Model 1 shows marginally significant ($p < 0.10$), positive coefficients related to direction and task-specific information. In model 2, the coefficient related to task-specific information is marginally significant ($p < 0.10$), too, while the coefficient related to direction becomes insignificant (although at $p = 0.104$). On contrast to these, the coefficients related to generic information are clearly insignificant ($p = 0.70$ in model 1 and $p = 0.51$ in model 2). These results indicate that task-specific information and direction can help decrease the cognitive burden that learning tasks impose on vendor engineers. As a result vendor engineers may be able to perform such a guided learning task even when they lack experience in the relevant area of the software, just like the vendor engineer in case 3 when he received a detailed design document and further clarifying information on the task from the expert. Conversely, it does not seem that the cognitive burden on vendor engineers is relieved by providing generic information, such as through induction sessions or the study of software documentation, highlighting thus the limitations of formal information sharing mechanisms.

6 Discussion

This study was motivated by the lack of clarity about what knowledge is critical in SMOO and about how critical knowledge is transferred in SMOO transitions. Our key findings are that (1) software knowledge is most critical and that (2) guided learning tasks are the key mechanism for transferring this knowledge to the new SDU. These findings have important implications for the transition literature and for TCE-based and knowledge-based perspectives on IS outsourcing and offshoring.

6.1 Implications for the Transition Literature

A key insight flowing from our study is that the key knowledge transfer mechanism for enabling the new SDU to take over are *guided learning tasks*. While existing work describes the use of a variety of mechanisms, such as formal information sharing, socialization, and practice-based mechanisms (Chua and Pan 2008; Kotlarsky et al. 2014; Schott 2011; Tiwari 2009; Williams 2011), these studies have not aimed at examining which mechanisms enable engineers in the new SDU to take over work. Our study leveraged a longitudinal data collection approach that captures in detail the activities of on-site vendor engineers during knowledge transfer, allowing thus more fine-grained analyses than in previous work. This analysis has led us to discover a surprisingly simple regularity: Vendor engineers were able to independently and successfully complete a given maintenance task only if they had previously worked on a guided learning task in the same area of the software application. Guided learning tasks are authentic maintenance tasks on which experts provide direction and task-specific information. While some vendors emphasize their methods for formal information sharing (e.g. knowledge codification in documents) (Cognizant 2002; Tiwari 2009), our results suggest a rather pessimistic view on the effects of formal information sharing. The engineers in the cases of high knowledge specificity struggled to make sense of documents before they started working on particular tasks. In a similar vein, our regression results indicate that generic information (i.e., information that is not created or tailored for a particular task) hardly enables engineers to take over work. While existing research has also pointed to limitations of formal information sharing mechanisms such as cognitive overload (Chua and Pan 2008) and low fit for complex work (Kotlarsky et al. 2014), our study goes beyond these studies by demonstrating that formal information sharing is not what enables vendor engineers to take work.

Our study also offers insights into the somewhat puzzling question of why *knowledge transfer durations* vary between a few weeks and a few years in existing case studies (Chua and Pan 2008; Kotlarsky et al. 2014; Tiwari 2009; Williams and Durst 2018). Studies reporting shorter knowledge transfer (or transition) durations tend to focus on formal information sharing mechanisms (Kotlarsky et al. 2014; Tiwari 2009). While it is likely that a series of formal information sharing activities (e.g. presentations, capturing presentation content in documents) can be concluded within a few weeks, these activities are unlikely to enable vendor engineers to take over work in environments of high knowledge specificity. In light of our findings on guided learning tasks and the key role of software knowledge, the engineers of the new SDU will be able to independently and successfully complete software maintenance tasks only after they worked, under the guidance of experts, on a series of maintenance tasks covering the whole software application. Depending on the size of software, this process may take several months or even years. This need for knowledge transfer through a series of guided learning tasks may explain why some studies report rather long transition durations (Barthélemy 2001; Williams and Durst 2018) and why several studies report that they needed to retain people from the incumbent SDU or make much greater use of them than planned (Chua and Pan 2008; Dibbern et al. 2008). Given that it appears unrealistic to hand over software maintenance work within a few weeks through formal information sharing mechanisms in settings of high knowledge specificity, an important question becomes how and how long the former SDU (an incumbent vendor and/or the client) and the new SDU should co-exist in an outsourcing or offshoring arrangement. The bank in our case studies addressed this challenge by what has been called concurrent sourcing (Tiwana and Kim 2016), i.e., by retaining their in-house staff such that they were able to help vendor engineers even several months after the start of the transition and even after incumbent vendors have left. Future research on transitions may thus look more systematically at different coexistence strategies and their effects on knowledge transfer.

Our study offers a third important insight around the question of whether knowledge transfer in SMOO concerns primarily *individual or collective knowledge* (Chua and Pan 2008; Lam 2000). Even though the decision to outsource or offshore is an organizational-level decision, our findings highlight the key role of individual-level learning processes in capturing the consequences of this organizational-level decision. The software knowledge that the new SDU aimed to acquire was clearly individual-level knowledge (rather than collective knowledge such as documents, shared memories, or organizational routines; Lam 2000). Only by engaging in learning tasks did engineers acquire the cognitive schemas of the software (i.e., software knowledge) that subsequently helped them make sense of thousands of lines of code and millions of database entries. Capturing this knowledge in documents and making it available to all engineers within the new SDU would hardly have substituted that learning process, given our findings on the limitations of generic information sharing. Software knowledge is thus, to a large extent, what Polanyi (1962) has called “personal knowledge”. The case of Pratap and Sai (case 5) also illustrates this. Even though they worked for the same vendor, Sai needed to learn the custom-developed software from scratch, a fact that the short parallel presence of Pratap and Sai did not accommodate for, leading thus to unsatisfactory service quality for the client. More broadly, these findings highlight that macro-micro-macro perspectives can be helpful for explaining the effects of sourcing decisions (Coleman 1990).

6.2 Implications for TCE-Based Research

In line with the assertion in TCE that asset specificity is a critical construct, knowledge specificity was a useful construct for capturing the differences between cases of custom-developed software maintenance and cases of packaged-software maintenance. The cases of custom-developed software maintenance were clearly of high knowledge specificity given that "skills [needed to be] acquired in a learning-by-doing fashion and were imperfectly transferrable across [clients]" (Williamson 1981, p. 563). The vendor engineers needed to acquire software knowledge through guided learning tasks (a type of learning-by-doing) and the software knowledge was hardly transferable across clients given that the knowledge domain—the custom-developed software—was unique to the client. In such settings, even highly experienced vendor engineers felt "like a layman". In contrast, the two engineers in the case of packaged software maintenance were in an environment of much lower knowledge specificity given that they could leverage their learning-by-doing from prior projects to quickly understand the custom-specific aspects of the software implementation at that client. There is thus an important difference between custom-developed software and packaged-software maintenance in the extent to which the most critical knowledge—software knowledge—loses or keeps its value when the engineers possessing the knowledge are deployed to another client. While prior TCE-based research has speculated that knowledge in software maintenance is "generally not company-specific" (Poppo and Zenger 1998, p. 871), that it is "probably not specific" (Aubert et al. 2004, p. 929), and that "maybe [the] variance [in the specificity of this knowledge] is not sufficient to lead significant differences" (Aubert et al. 2004, p. 929), our multiple-case study shows that these concerns are probably not warranted. In contrast, knowledge in SMOO can be highly specific, notably in settings of custom-developed software maintenance where engineers may need to work on a series of guided learning tasks over months or years until they are fully understand a client-specific software system. As our comparison of custom-developed software (cases 1, 2, 3, 5) and packaged software maintenance (case 4) show, the variance in knowledge specificity may also be substantial, such that some software applications may require months or years of learning while others do not.

Our study points to one potential reason for mixed support when existing studies tested TCE in arrangements that included software maintenance. In line with the awareness of potential measurement problems in TCE research (Alaghehband et al. 2011; Lacity et al. 2011), our study suggests that existing research may not capture sufficient variance in knowledge specificity because of the measures and the levels of analysis used. As our review of the TCE literature shows, existing studies rarely measure the specificity of software knowledge (Ang and Straub 1998; Barthélemy and Geyer 2005; Diana 2009; Loebbecke and Huyskens 2006; Miranda and Kim 2006; Poppo and Zenger 2002). Yet software knowledge was the most critical knowledge in the five cases that we studied. Although most IS outsourcing studies have an empirical scope that is somewhat broader than software maintenance (e.g. by including new software development), software maintenance is likely to be a substantial portion of the outsourcing business given that the majority of software costs accrue during maintenance (Banker et al. 2002; Kemerer and Slaughter 1999) and that software maintenance is particularly popular for outsourcing (Deloitte 2016). The negligence of the specifics of software maintenance is particularly troublesome for tests of TCE given that TCE emphasizes the hold-up risks that accrue over time from client-specific learning-by-doing. Since software maintenance naturally follows software development, software maintenance is a particularly important context for capturing the essence of such learning accruing over time. Moreover, it is possible that software knowledge is also valuable for tasks other than maintenance, such as software reengineering projects (Dibbern et al. 2008; Gregory et al. 2009) and software development projects in which interfaces to existing software need to be built. Future TCE research should thus put greater emphasis on software knowledge.

Another potential issue in existing TCE research concerns the levels of analysis. Only one study chose the level of the software application (Dibbern et al. 2008) whereas most work chose the higher levels of analysis such as client organizations (Ang and Straub 1998; Diana 2009; Loebbecke and Huyskens 2006; Miranda and Kim 2006) and IS functions (see the second survey reported in Aubert et al. 2004; Barthélemy and Geyer 2005; Dibbern et al. 2016; Poppo and Zenger 2002). We would have been unable to capture the substantive difference in knowledge specificity between the cases of custom-developed software maintenance and of packaged-software maintenance if we had aggregated our data to the level of the client organizations or of the IS function. Future TCE research should thus consider testing TCE at the level of the particular software application. It is important to note that, while our study points to measurement issues in TCE, our study is not a test of TCE.

6.3 Implications for Knowledge-Based Perspectives

Our study also offers implications for the discourse around knowledge-based perspectives in outsourcing and offshoring. Like others did before us (Dibbern et al. 2008), our study shows that there can be “irreducible knowledge differences” (Conner and Prahalad 1996) between parties. In the cases of high knowledge specificity, the differences in knowledge between engineers from the incumbent SDU and engineers from the new SDU was clearly irreducible. Even after the engineers from the new SDU had worked on guided learning tasks over a period of several months, they were still dependent on the guidance by experts when they worked in areas of the software in which they had not worked before. This long time needed to acquire software knowledge may explain why Dibbern et al. (2016) found particularly high switching costs in outsourcing of software maintenance work of high knowledge specificity.

While our study supports the idea of irreducible knowledge differences, it provides no support for the idea that direction is infeasible in outsourced settings given the lack of authority of the client (Conner and Prahalad 1996). Indeed, the engineers in the new SDU gladly welcomed any direction from client engineers and from engineers working for the former vendor. Being cognitively overburdened by unfamiliar software code, data, and documents, they were clear that the direction provided by these experts helped them lower their cognitive load. Moreover, while knowledge-based perspectives have considered direction as a substitute for knowledge transfer (Conner and Prahalad 1996; Dibbern et al. 2008), our study shows that direction can also play an important role for enabling knowledge transfer. Direction helps reduce the cognitive burden on engineers, making a knowledge transfer approach based on real or realistic tasks (learning tasks) feasible.

6.4 Limitations

Our study is not without limitations. Our focus on cases from one particular client may have led us to overlook issues that were not salient at that client. Moreover, although we examined the learning processes of two vendor engineers under low knowledge specificity, these engineers were actors in the same case of packaged software maintenance. Examining a greater breadth of packaged-software settings might yield new insights. Furthermore, although the dynamics in the social relationships between vendor engineers and client staff differed quite substantially between cases (e.g. between case 1 and 5), we did not focus the analysis reported in this paper on these differences for the sake of parsimony. Finally, although our regression analysis shall serve primarily the function of corroborating insights form qualitative analyses, one may criticize that the analysis suffers from a relatively low sample size of 51 learning tasks.

Appendix: Interview Guideline

See Table 5.

Table 5 Interview guideline

Interviewee	Questions
Vendor engineer (first interview only)	1. What is your understanding of the task you are about to take over? 2. What are your expectations towards the transition? 3. Describe your prior experiences that might be of value for this project. 4. What happened so far?
Vendor engineer (all interviews)	5. What happened since our last interview? Please describe all activities in detail. • *Probe for formal presentations, document study, informal discussions, work on tasks, studying examples, code study.* • *For each activity, probe for time, duration, involved participants, and the knowledge domains to which they were related.* • *For work on tasks and example study, probe:* - What was the task about? - How did you work on the task from the beginning to the end? - What documents were involved in the task? - Were you able to resolve the task successfully? If not, what were the problems? - How difficult did you find the task at that time? - Did you receive any guidance or information on the task? If so, what guidance or information? Why, how, when, by whom, and how long?
Vendor engineer (last interview only)	6. What experiences or knowledge from prior projects have been useful to you in this project? How much prior experience did you have in each of these areas? • *Probe for technical knowledge, business knowledge, software knowledge, process knowledge, and organizational knowledge.* 7. How satisfied are you with the transition? What would you do differently next time?
Experts and Client Managers	8. Please explain your current role. 9. How does the transition relate to larger programs or initiatives of the client? 10. What will be the role of the vendor engineer after transition? 11. Did you select the vendor engineer? If so, how? 12. *Experts only*: What knowledge or skills does the vendor engineer need to take over the maintenance role? • *Probe for technical knowledge, business knowledge, software knowledge, process knowledge, and organizational knowledge.* 13. Please describe what happened so far during the transition phase. • *Probe for formal presentations, document study, informal discussions, work on tasks, studying examples, code study.* 14. Let's talk about the tasks on which the vendor engineer has worked so far. For each task, please report: • How complex was the task? What made it complex or simple? • Did you provide any help on this task? If so, how, why, when and how long? • Has the vendor engineer successfully solved the task? If not, what was the problem? *Probe whether time schedules were met and whether defects were raised.* 15. What makes this transition particularly difficult or smooth? 16. How satisfied are you with the transition? What would you do differently next time?

Note We asked further questions on how the transitions were managed and how relationships evolved. We do not report these questions here because they have not been used for this paper. We also asked closed questions on constructs from the knowledge management literature, such as source credibility, expert's motivation to share knowledge, and the vendor engineer's motivation to learn. During data analysis, these constructs did not turn out to play central roles in the cases examined here

References

Alaghehband, F. K., Rivard, S., Wu, S., & Goyette, S. (2011). An assessment of the use of transaction Cost Theory in Information Technology Outsourcing. *The Journal of Strategic Information Systems, 20*(2), 125–138.

Ang, S., & Straub, D. W. (1998). Production and Transaction Economies and IS Outsourcing: A Study of the US Banking Industry. *MIS Quarterly, 22*(4), 535–552.

Argote, L. (2012). *Organizational Learning: Creating*. Retaining and Transferring Knowledge, Heidelberg, Germany: Springer.

Aubert, B. A., Rivard, S., & Patry, M. (1996). Development of Measures to Assess Dimensions of IS Operation Transactions. *Omega, 24*(6), 661–680.

Aubert, B. A., Rivard, S., & Patry, M. (2004). A Transaction Cost Model of IT Outsourcing. *Information & Management, 41*(7), 921–932.

Banker, R. D., Datar, S. M., Kemerer, C. F., & Zweig, D. (2002). Software Errors and Software Maintenance Management. *Information Technology and Management, 3*(1–2), 25–41.

Barthélemy, J. (2001). The Hidden Costs of IT Outsourcing: Lessons from 50 IT-Outsourcing Efforts Show That Unforeseen Costs Can Undercut Anticipated Benefits. Understanding the Issues Can Lead to Better Outsourcing Decisions. *MIT Sloan Management Review, 42*(3), 60–72.

Barthélemy, J., & Geyer, D. (2005). An Empirical Investigation of IT Outsourcing versus Quasi-Outsourcing in France and Germany. *Information & Management, 42*(4), 533–542.

Beulen, E., Tiwari, V., & van Heck, E. (2011). Understanding Transition Performance during Offshore IT Outsourcing. *Strategic Outsourcing: An International Journal, 4*(3), 204–227.

Blackler, F. 1995. “Knowledge, Knowledge Work and Organizations: An Overview and Interpretation,” *Organization Studies* (16:6), p. 1021.

Boh, W. F., Slaughter, S. A., & Espinosa, J. A. (2007). Learning from Experience in Software Development: A Multilevel Analysis. *Management Science, 53*(8), 1315–1331.

Carter, R., & Hodgson, G. M. (2006). The Impact of Empirical Tests of Transaction Cost Economics on the Debate on the Nature of the Firm. *Strategic Management Journal, 27*(5), 461–476.

Chen, Y., & Bharadwaj, A. (2009). An Empirical Analysis of Contract Structures in IT Outsourcing. *Information Systems Research, 20*(4), 484–506.

Chua, A. L., & Pan, S. L. (2008). Knowledge Transfer and Organizational Learning in IS Offshore Sourcing. *Omega, 36*(2), 267–281.

Cognizant. 2002. “How to Conduct Knowledge Transfer Efficiently.” (http://www.cognizant.com/InsightsWhitepapers/amseries-kt.pdf).

Coleman, J. S. (1990). *Foundations of Social Theory*. Cambridge, MA: Belknap Press of Harvard University.

Conner, K., & Prahalad, C. (1996). A Resource-Based Theory of the Firm: Knowledge versus Opportunism. *Organization Science, 7*(5), 477–501.

Deloitte. 2016. “Current and Planned Information Technology Outsourcing (ITO) by Service in 2014.” (http://www.statista.com/statistics/496625/outsourcing-information-technology/).

Diana, M. L. 2009. “Exploring Information Systems Outsourcing in US Hospital-Based Health Care Delivery Systems,” *Health Care Management Science* (12:4), p. 434.

Dibbern, J., Chin, W. W., & Kude, T. (2016). The Sourcing of Software Services: Knowledge Specificity and the Role of Trust. *ACM SIGMIS Database, 47*(2), 36–57.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2004). Information Systems Outsourcing: A Survey and Analysis of the Literature. *ACM SIGMIS Database, 35*(4), 6–102.

Dibbern, J., Winkler, J., & Heinzl, A. (2008). Explaining Variations in Client Extra Costs between Software Projects Offshored to India. *MIS Quarterly, 32*(2), 333–366.

Distel, A. P. 2019. “Unveiling the Microfoundations of Absorptive Capacity: A Study of Coleman’s Bathtub Model,” *Journal of Management* (45:5), pp. 2014–2044. (https://doi.org/10.1177/0149206317741963).

Feng, Y., Ye, H., and Pan, S. L. 2010. *Delivering Knowledge across Boundaries: A Process Model of Knowledge Delivery in Offshoring Projects*, presented at the Pacific Asia Conference on Information Systems.

Gopal, A., & Gosain, S. (2010). The Role of Organizational Controls and Boundary Spanning in Software Development Outsourcing: Implications for Project Performance. *Information Systems Research, 21*(4), 1–23.

Grant, R. M. (1996). Toward a Knowledge-Based Theory of the Firm. *Strategic Management Journal, 17*(10), 109–122.

Gregory, R., Beck, R., & Keil, M. (2013). Control Balancing in Information Systems Development Offshoring Projects. *MIS Quarterly, 37*(4), 1211–1232.

Gregory, R., Beck, R., and Prifling, M. 2009. *Breaching the Knowledge Transfer Blockade in IT Offshore Outsourcing Projects - A Case from the Financial Services Industry*, presented at the The 43th Hawaii International Conference on Systems Sciences.

Hodgson, G. M. (1998). Competence and Contract in the Theory of the Firm. *Journal of Economic Behavior & Organization, 35*(2), 179–201.

Iivari, J., Hirschheim, R., & Klein, H. (2004). Towards a Distinctive Body of Knowledge for Information Systems Experts: Coding ISD Process Knowledge in Two IS Journals. *Information Systems Journal, 14*(4), 313–342.

Imsland, V., and Sahay, S. 2005. "'Negotiating Knowledge': The Case of a Russian–Norwegian Software Outsourcing Project," *Scandinavian Journal of Information Systems* (17:1), p. 3.

Jensen, R., & Szulanski, G. (2004). Stickiness and the Adaptation of Organizational Practices in Cross-Border Knowledge Transfers. *Journal of International Business Studies, 35*(6), 508–523.

Kemerer, C. F., & Slaughter, S. A. (1999). An Empirical Approach to Studying Software Evolution. *Software Engineering, IEEE Transactions On, 25*(4), 493–509.

Kogut, B., & Zander, U. (1996). What Firms Do? Coordination, Identity, and Learning. *Organization Science, 7*(5), 502–518.

Kotlarsky, J., Oshri, I., Dibbern, J., and Mani, D. 2018. "IS Sourcing," in *MIS Quarterly Research Curations*, A. Bush and A. Rai (eds.). (http://misq.org/research-curations).

Kotlarsky, J., Scarbrough, H., & Oshri, I. (2014). Coordinating Expertise across Knowledge Boundaries in Offshore-Outsourcing Projects: The Role of Codification. *MIS Quarterly, 38*(2), 607–627.

Lacity, M. C., Khan, S., Yan, A., & Willcocks, L. P. (2010). A Review of the IT Outsourcing Empirical Literature and Future Research Directions. *Journal of Information Technology, 25*(4), 395–433.

Lacity, M. C., Willcocks, L. P., & Khan, S. (2011). Beyond Transaction Cost Economics: Towards an Endogenous Theory of Information Technology Outsourcing. *The Journal of Strategic Information Systems, 20*(2), 139–157.

Lam, A. 2000. "Tacit Knowledge, Organizational Learning and Societal Institutions: An Integrated Framework," *Organization Studies* (21:3), p. 487.

Langley, A. (1999). Strategies for Theorizing from Process Data. *The Academy of Management Review, 24*(4), 691–710.

Lee, J., & Kim, Y. (1999). Effect of Partnership Quality on IS Outsourcing Success: Conceptual Framework and Empirical Validation. *Journal of Management Information Systems, 15*(4), 29–61.

Leonardi, P. M., & Bailey, D. (2008). Transformational Technologies and the Creation of New Work Practices: Making Implicit Knowledge Explicit in Task-Based Offshoring. *MIS Quarterly, 32*(2), 159–176.

Levina, N., & Vaast, E. (2008). Innovating or Doing as Told? Status Differences and Overlapping Boundaries in Offshore Collaboration. *MIS Quarterly, 32*(2), 307–332.

Loebbecke, C., & Huyskens, C. (2006). What Drives Netsourcing Decisions? An Empirical Analysis. *European Journal of Information Systems, 15*(4), 415–423.

Macher, J. T., & Richman, B. D. (2008). Transaction Cost Economics: An Assessment of Empirical Research in the Social Sciences. *Business and Politics, 10*(1), 1–63.

Miles, M. B., Huberman, A. M., & Saldaña, J. (2014). *Qualitative Data Analysis: A Methods Sourcebook*. Thousand Oaks, CA: SAGE.
Miranda, S. M., and Kim, Y.-M. 2006. "Professional versus Political Contexts: Institutional Mitigation and the Transaction Cost Heuristic in Information Systems Outsourcing," *Mis Quarterly*, pp. 725–753.
Nahapiet, J., and Ghoshal, S. 1998. "Social Capital, Intellectual Capital, and the Organizational Advantage," *Academy of Management Review*, pp. 242–266.
Nicholson, B., & Sahay, S. (2004). Embedded Knowledge and Offshore Software Development. *Information and Organization, 14*(4), 329–365.
Oshri, I., Van Fenema, P., & Kotlarsky, J. (2008). Knowledge Transfer in Globally Distributed Teams: The Role of Transactive Memory. *Information Systems Journal, 18*(6), 593–616.
Pentland, B. T. 1992. "Organizing Moves in Software Support Hot Lines," *Administrative Science Quarterly* (37:4), pp. 527–548. (https://doi.org/10.2307/2393471).
Polanyi, M. (1962). *Personal Knowledge*. Towards a Post-Critical Philosophy, Chicago, IL: The University of Chicago Press.
Poppo, L., & Zenger, T. (1998). Testing Alternative Theories of the Firm: Transaction Cost, Knowledge-based, and Measurement Explanations for Make-or-buy Decisions in Information Services. *Strategic Management Journal, 19*(9), 853–877.
Poppo, L., & Zenger, T. (2002). Do Formal Contracts and Relational Governance Function as Substitutes or Complements? *Strategic Management Journal, 23*(8), 707–725.
Rai, A., Maruping, L. M., & Venkatesh, V. (2009). Offshore Information Systems Project Success: The Role of Social Embeddedness and Cultural Characteristics. *MIS Quarterly, 33*(3), 617–641.
Schott, K. 2011. *Vendor-Vendor Knowledge Transfer In Global ISD Outsourcing Projects: Insights From A German Case Study*, presented at the PACIS, p. 170.
Shaft, T. M., and Vessey, I. 1995. "The Relevance of Application Domain Knowledge: The Case of Computer Program Comprehension," *Information Systems Research* (6:3), pp. 286–299. (https://doi.org/10.1287/isre.6.3.286).
Szulanski, G. (1996). Exploring Internal Stickiness: Impediments to the Transfer of Best Practice within the Firm. *Strategic Management Journal, 17*(1), 27–43.
Tiwana, A. 2004a. "Beyond the Black-Box: Knowledge Overlaps in Software Outsourcing," *IEEE Software* (21:5).
Tiwana, A. (2004b). An Empirical Study of the Effect of Knowledge Integration on Software Development Performance. *Information and Software Technology, 46*(13), 899–906.
Tiwana, A., & Keil, M. (2009). Control in Internal and Outsourced Software Projects. *Journal of Management Information Systems, 26*(3), 9–44.
Tiwana, A., & Kim, S. K. (2016). Concurrent IT Sourcing: Mechanisms and Contingent Advantages. *Journal of Management Information Systems, 33*(1), 101–138.
Tiwari, V. 2009. *Transition During Offshore Outsourcing: A Process Model*, presented at the The 30th International Conference of Information Systems.
Uzzi, B. 1997. "Social Structure and Competition in Interfirm Networks: The Paradox of Embeddedness," *Administrative Science Quarterly*, pp. 35–67.
Vlaar, P., van Fenema, P., and Tiwari, V. 2008. "Cocreating Understanding and Value in Distributed Work: How Members of Onsite and Offshore Vendor Teams Give, Make, Demand, and Break Sense," *MIS Quarterly* (32:2), p. 5.
Von Mayrhauser, A., & Vans, A. M. (1995). Program Comprehension during Software Maintenance and Evolution. *Computer, 28*(8), 44–55.
Wende, E., Schwabe, G., and Philip, T. 2010. *Exploring the Media Mix During IT-Offshore Transition Phase – an Opportunity for a Better Knowledge Transfer*, presented at the The Fourth Global Sourcing Workshop,, March 22.
Wende, E., Schwabe, G., Philip, T., and King, G. 2013. *Why They Do Not Understand–A Model of Knowledge Discourse in the Transition Phase of Globally Distributed Projects*, presented at the System Sciences (HICSS), 2013 46th Hawaii International Conference on, IEEE, pp. 396–405.

Whitaker, J., Mithas, S., & Krishnan, M. S. (2010). Organizational Learning and Capabilities for Onshore and Offshore Business Process Outsourcing. *Journal of Management Information Systems, 27*(3), 11–42.

Wiener, M., Vogel, B., and Amberg, M. 2010. "Information Systems Offshoring—A Literature Review and Analysis," *Communications of the Association for Information Systems* (27:1), p. 25.

Williams, C. (2011). Client–Vendor Knowledge Transfer in IS Offshore Outsourcing: Insights from a Survey of Indian Software Engineers. *Information Systems Journal, 21*(4), 335–356.

Williams, C., and Durst, S. 2018. "Exploring the Transition Phase in Offshore Outsourcing: Decision Making amidst Knowledge at Risk," *Journal of Business Research*.

Williamson, O. E. (1981). The Economics of Organization: The Transaction Cost Approach. *American Journal of Sociology, 87*(3), 548–577.

Winkler, J., Dibbern, J., & Heinzl, A. (2008). The Impact of Cultural Differences in Offshore Outsourcing—Case Study Results from German-Indian Application Development Projects. *Information Systems Frontiers, 10*(2), 243–258.

Yin, R. K. (2009). *Case Study Research: Design and Methods*. Thousand Oaks, CA: Sage.

Zimmermann, A., Oshri, I., Lioliou, E., and Gerbasi, A. 2018. "Sourcing in or out: Implications for Social Capital and Knowledge Sharing," *The Journal of Strategic Information Systems* (27:1).

Zimmermann, A., Raab, K., & Zanotelli, L. (2013). Vicious and Virtuous Circles of Offshoring Attitudes and Relational Behaviours. A Configurational Study of German IT Developers. *Information Systems Journal, 23*(1), 65–88.

Zimmermann, A., & Ravishankar, M. (2014). Knowledge Transfer in IT Offshoring Relationships: The Roles of Social Capital, Efficacy and Outcome Expectations. *Information Systems Journal, 24*(2), 167–202.

Cultural Intelligence of Offshore IT Suppliers

Ning Su

Abstract Today's IT service suppliers increasingly need to acquire "cultural intelligence", or the ability to operate in a culturally-diverse environment. Based on an interview-based, qualitative case study of one of the largest IT service suppliers in China, this study explores how the offshore IT service supplier develops cultural intelligence through interaction with a portfolio of clients with diverse cultural backgrounds. Drawing on the dynamic constructivist view of culture, the study adapts the concept of cultural frame to define a set of shared interpretive schemes and practices that enable the supplier to make sense of and respond to clients from different cultures. During repeated client-supplier interaction, supplier employees' cultural frames are continuously enacted, aligned and realigned, and eventually institutionalized into a set of cultural structures and artifacts. This emergence and embedding process is both facilitated by, and in turn enhances, the supplier's firm-level cultural intelligence.

1 Introduction

Today's information technology (IT) service suppliers increasingly need to acquire "cultural intelligence" (Ang and Inkpen 2008), or the ability to operate effectively in culturally-diverse environments (e.g., Dibbern et al. 2008). Such intercultural capabilities can be challenging to develop for offshore suppliers with cultural backgrounds significantly different from those of their clients'. During client-supplier interaction, different parties need to continuously adjust their behaviors and expectations (e.g., Krishna et al. 2004). Such adaption tends to be a challenging and complex process (e.g., Aubert et al. 2005; Rai et al. 2009).

To alleviate cultural barriers, client firms can adopt a set of best practices prescribed in both information systems (IS) and management literatures. For example, clients can leverage cultural mentors and liaisons (Osland and Bird 2000; Carmel and Agarwal 2001) and implement appropriate routines and methodologies (Krishna

N. Su (✉)
Ivey Business School, Western University, London, Canada
e-mail: nsu@ivey.ca

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_8

et al. 2004; Leonardi and Bailey 2008). The literature, meanwhile, provides limited insights into the specific cultural processes and capabilities from the perspective of the supplier (e.g., Vlaar et al. 2008), although research has shown that suppliers' capabilities gradually evolve and mature as the suppliers gain increasing experience with clients from different markets (e.g., Carmel et al. 2008).

The goal of this study is to explore at a micro level (Eisenhardt et al. 2010), how offshore IT service suppliers acquire cultural intelligence during their interaction with a portfolio of clients with different cultural backgrounds. China's IT service industry provides a suitable setting for exploring this question, for several reasons. On one hand, compared to major multinational and India-based suppliers with mature capabilities, such as IBM and TCS (e.g., Oshri et al. 2007), even China's largest IT service firms are still in a comparatively early phase of development. On the other hand, since the mid 2000s, major Chinese suppliers have been actively expanding their business in international markets, and have achieved some success (e.g., Rottman and Hao 2008). Today, Chinese suppliers have secured contracts with major clients from the world's three largest economies: U.S., China, and Japan. It is crucial for the suppliers to be able to effectively interact with these distinct cultures (e.g., Tiwana et al. 2008). In addition, China itself represents an important and yet understudied location in both IT outsourcing research (e.g., Jarvenpaa and Mao 2008) and the overall IS literature (e.g., Zhang et al. 2008; Levina et al. 2011).

To explore the research question, a multi-year, in-depth, interview-based case study was conducted at one of the largest and most globally recognized Chinese IT service suppliers (e.g., IAOP 2008, 2009, 2010, 2011, 2012). The firm had been working for major clients from China's three largest IT service markets: Japan, U.S., and the Chinese domestic market. Based on the interview data and drawing on research in cultural psychology and sociology, this study adapts the concept of cultural frame and elaborates the process by which the supplier effectively interacts with clients from different national cultural backgrounds.

2 Literature Review

This section first provides an overview of the research on culture-related issues in both IS and international business (IB) literatures. This section then reviews different conceptualizations of culture, and highlights the dynamic constructivist view and the notion of cultural frame. Finally, the context of this study, China, is briefly discussed based on the existing IB literature.

2.1 Culture in IT Outsourcing

Culture is an increasingly important element in the development, use, and management of today's information systems (Ford et al. 2003; Leidner and Kayworth 2006;

Kappos and Rivard 2008). The IS literature has demonstrated that culture has a significant impact on both the processes and outcomes of a broad range of IT initiatives, including: business process reengineering (BPR) (Martinsons and Hempel 1998); virtual organizations (Jarvenpaa and Leidner 1999); enterprise resource planning (ERP) solution implementation (Soh et al. 2000); online programming marketplace (Gefen and Carmel 2008); e-Commerce (Sia et al. 2009); IT offshoring (Sarker and Sarker 2009); and outsourcing (Lacity et al. 2010). As culture shapes these IT initiatives, these initiatives, in turn, also contribute to the change and transformation of culture, both at national (Madon 1992) and organizational levels (Doherty and Doig 2003).

Global IT outsourcing, in particular, brings compelling needs and unique opportunities to further understand culture (Nicholson and Sahay 2001; Sahay et al. 2003). Global outsourcing requires the client and the supplier to collaborate across national and organizational boundaries (Levina and Vaast 2005; 2008; Du and Pan 2010). In such boundary-spanning activities, culture plays an important role (Salk and Brannen 2000; Cramton and Hinds 2007). With different national cultural origins, different client firms (e.g., Tiwana and Bush 2007; Dibbern et al. 2012) and IT suppliers (e.g., Jarvenpaa and Mao 2008; Carmel et al. 2008) exhibit different practices and decisions. Various stakeholders need to continuously modify their behaviors, beliefs, and expectations (e.g., Koh et al. 2004) until a "negotiated culture" is formed (Brannen and Salk 2000; Krishna et al. 2004). This process of cross-cultural adaptation can be expensive and risky (Hahn et al. 2009), as research shows that national cultural difference tends to increase cost (Dibbern et al. 2008; Rai et al. 2009) and reduce client satisfaction (ibid).

To improve cross-cultural collaboration, client firms have developed a set of best practices. These practices include: providing cultural training to foster mutual understanding of national cultural norms of the client and the supplier (Kaiser and Hawk 2004); employing "cultural mentors", who are usually long-term expatriates or members of foreign cultural origins (Osland and Bird 2000); hiring "cultural liaisons" (Carmel and Agarwal 2001) or "straddlers" (Heeks et al. 2001), who are knowledgeable of multiple cultures, to mitigate cultural barriers; assigning client managers to the supplier's offshore locations to establish relationships and provide training (Levina and Su 2008); building one-to-one collaboration between individual client and supplier developers to facilitate the transferring of informal information and tacit knowledge (Heeks et al. 2001); and creating formal coordination and control mechanisms (Leonardi and Bailey 2008), such as sharing common development methodologies and compatible technologies with the supplier (Krishna et al. 2004).

The goal of these best practices is to enable the client to operate effectively in cross-cultural environments. At the individual level, this intercultural capability is termed "cultural intelligence" (CQ) (Earley and Ang 2003; Ang and Van Dyne 2008). Individual-level CQ consists of four dimensions: meta-cognitive, cognitive, motivational, and behavioral (e.g., Ang et al. 2007). Elevating this individual-level construct to the collective level, Ang and Inkpen (2008) define a firm's intercultural capability as firm-level or organizational cultural intelligence (e.g., Hong 2010; Chen et al. 2012). In the context of offshore IT outsourcing, firm-level CQ comprises three dimensions: first, managerial, which refers to the client firm's individual managers' cultural intelligence; second, competitive, which refers to a strategic approach toward leveraging global outsourcing to create a competitive advantage; third, structural, which refers to the firm's processes and routines for facilitating cross-cultural interaction (Ang and Inkpen 2008).

Among the diverse set of research on culture-related issues in global outsourcing, most studies adopt the "pan-cultural" (Hong and Chiu 2001) or "trait" (Morris and Fu 2001) approach. This approach reduces national culture into a set of general dispositions. Hofstede's (1980) theory is the most prominent example of the pan-cultural approach. The pan-cultural approach has provided a parsimonious, heuristic generalization of national culture. On the other hand, however, this approach has been criticized as static and overly simplistic for understanding cultural issues in both IS (e.g., Avison and Myers 1995; Myers and Tan 2002; Straub et al. 2002) and management in general (e.g., Redding 1994; Goodall 2002; Tung and Verbeke 2010). In more recent research, culture is increasingly characterized as situated, contested, heterogeneous, and dynamic (e.g., Walsham 2002; Srite and Karahanna 2006; Avison and Banks 2008).

Given the IS literature's focus on the relatively static and reductionist national culture models, there is limited research on how a firm as a collectivity manages intercultural interaction and how a firm develops intercultural capabilities (e.g., Wiener et al. 2010; Chen et al. 2012). Moreover, in the IT outsourcing literature, most cultural studies adopt the client's perspective, offering limited insights into the micro-level interactions within the suppliers (Vlaar et al. 2008). Among the studies of offshore suppliers, most focus on major Indian firms, which have already developed mature capabilities for Western markets and no longer face major cultural barriers (e.g., Oshri et al. 2007; Olsson et al. 2008). Research on suppliers from other emerging markets is needed to expand the knowledge of outsourcing (Lacity et al. 2010). To address these gaps, the following sections first review related international business and strategy literatures to obtain insights on how firms manage inter-firm relationships, and then synthesize related research in cultural psychology and sociology in order to develop an alternative conceptualization of culture for elaborating the social processes of cultural interactions in global IT outsourcing.

2.2 Culture in International Business

Suppliers' provisioning of IT services to offshore clients is a form of international business (Capar and Kotabe 2003). The IB literature has incorporated cultural factors since early development of the discipline. A fundamental assumption of the literature is the "liability of foreignness" (Hymer 1976; Zaheer 1995). According to this concept, firms need to overcome the "psychic distance" between their home countries and foreign markets, and cultural difference is a key component of psychic distance (Johanson and Wiedersheim-Paul 1975; Johanson and Vahlne 1977, 2009). National cultural difference can increase transaction costs and information processing needs, and therefore has been traditionally viewed as a barrier that hinders firms' performance. Recent research, however, shows that cultural difference can also give firms access to diverse knowledge and practices embedded in foreign cultures and form a source of competitive advantage (e.g., Morosini et al. 1998; Brannen 2004).

In the specific area of international alliances (e.g., Inkpen and Beamish 1997; Steensma et al. 2005) including outsourcing alliances, cultural difference can inhibit inter-firm learning and become a key obstacle to alliance success (e.g., Simonin 1999; Lane et al. 2001). On the other hand, if well-managed, cultural diversity brings valuable knowledge and capabilities, and can improve alliance performance (e.g., Lavie and Miller 2008). Firms need to develop the ability to manage such cross-cultural inter-firm relationships. Alliance management capability in general comprises three dimensions: coordination, communication, and bonding (Schreiner et al. 2009). These capabilities can be accumulated through the firm's repeated interaction with a portfolio of alliance partners (e.g., Ethiraj et al. 2005; Kale and Singh 2007). During client-supplier interaction, trust, information sharing, and joint problem-solving are the key mechanisms by which firms develop alliance management capabilities (McEvily and Marcus 2005).

In inter-firm relationships such as alliances, trust, in particular, plays an important role (e.g., ibid; Ring and Van de Ven 1994; King and Torkzadeh 2008). Interorganizational trust can facilitate client-supplier negotiation and improve supplier performance (e.g., Zaheer et al. 1998; Katsikeas et al. 2009). Trust can be based on both relationship, that is, social interactions between exchange partners, and process, that is, institutionalized procedures and routines between and within the partners (Dyer and Chu 2000). Trust is a culturally-conditioned construct (e.g., Doney et al. 1998; Currall and Inkpen 2002; Schoorman et al. 2007). In client-supplier relationships, the determinants and impact of interorganizational trust vary across cultures (e.g., Bensaou and Anderson 1999; Dyer and Chu 2000; Zaheer and Kamal 2011; MacDuffie 2011).

To characterize national cultures, the IB literature has predominantly adopted Hofstede's (1980) framework. More recent work in IB, however, highlights the significant limitations of the exclusive focus on the pan-cultural approach. This traditional approach inaccurately assumes a homogenous impact of cultural difference on firm performance (e.g., Tung and Verbeke 2010), neglects intra-national cultural diversity (e.g., Tung 2008), and masks the dynamic adaptation and change of culture that is especially pronounced in the era of globalization (e.g., Gelfand et al. 2006; Leung et al. 2005). Furthermore, even the validity of the notion of "national culture" becomes questionable (Gould and Grein 2009; Leung et al. 2011). To address these weaknesses, recent IB literature has been searching for a more fine-grained, dynamic, heterogeneous view of culture (e.g., Leung et al. 2005; Luo and Shenkar 2011). Qualitative methods can be especially valuable for acquiring such a view (Birkinshaw et al. 2011). However, to date, a theoretical lens that can realistically conceptualize the micro-level cultural interactions in practice has yet to emerge (Tung and Verbeke 2010). To assist in developing such an alternative lens, the following sections review related research in both cultural psychology and sociology.

2.3 Culture

Culture, broadly defined as the human-made part of the environment (Herskovitz 1955), includes both subjective aspects, such as values, beliefs and attitudes (Triandis et al. 1972; Triandis 1994), and objective aspects, such as tools, artifacts, social networks and institutions (Gelfand et al. 2006). In management research, the pan-cultural approach is still adopted as the dominant framework for modeling culture. This approach was pioneered by Hofstede (1980, 1991), who characterizes the cultural values of individuals from different nations along five dimensions: individualism-collectivism, power distance, uncertainty avoidance, masculinity-femininity, and long-term orientation. Virtually all subsequent pan-cultural models have incorporated Hofstede's dimensions (Taras et al. 2010). One widely-cited such study is the GLOBE (Global Leadership and Organizational Behavior Effectiveness) project, which identified nine cultural dimensions (House et al. 2004).

The pan-cultural approach assumes that culture is internalized as a set of coherent, tightly-integrated themes, manifested in the form of overall mentality, worldviews and values (DiMaggio 1997) that have a pervasive and continuous influence on individuals (Morris and Fu 2001). More recent development in cognitive and social psychology, however, shows that culture is in fact internalized in fragmented smaller pieces, in the form of loosely-linked and context-specific knowledge structures or representations (ibid; Bruner 1990; Gilbert 1991). Individuals can acquire and simultaneously possess multiple such cultural knowledge structures, including structures that conflict or contradict with each other. Such cultural knowledge is shared, maintained, and changed through social processes such as interpersonal communication (e.g., Lau et al. 2001). In specific contexts, a subset of knowledge is activated to guide individuals' interpretation of external stimuli (Hong et al. 2000; Morris and Fu 2001; Hong and Chiu 2001).

This above conceptualization of cultures as "dynamic open systems that spread across geographical boundaries and evolve through time" (Hong and Chiu 2001, p. 181) is termed the "dynamic constructivist approach" (ibid). This approach emphasizes that culture is temporal, emergent, domain-specific, and is constantly produced and reproduced, interpreted and reinterpreted in social interaction (Morris and Fu 2001; Oyserman et al. 2002). The construction of shared meanings among group members is deemed central to the definition of culture (e.g. Fischer 2006; Wan et al. 2007; Zou et al. 2009). As an emerging paradigm for describing and explaining culture, the dynamic constructivist approach has propelled extensive research on individual-level cognition and behaviors, such as bicultural individuals' cultural frame switching (Chen et al. 2009). Recently, this approach has been used to explore firm-level phenomena, such as globally-distributed work processes (e.g., Cramton and Hinds 2007; Hinds et al. 2011).

This dynamic, heterogeneous view of culture has also been increasingly adopted in sociology as the fields of cultural psychology and sociology converge (DiMaggio 1997). Sociology research has emphasized the strategic value of culture. In particular, culture can be conceptualized as "a grab-bag of odds and ends: a pastiche of mediated representations, a repertoire of techniques, or a toolkit of strategies" (DiMaggio 1997, p. 267). By focusing on how actors leverage culture to produce observable outcomes, this practice-oriented view of culture (Wedeen 2002) is especially valuable for analyzing social and organizational change. Specifically, in the social and political realm, culture is used as a set of rhetorical, interactional, and material tools to mobilize and coordinate collective action (e.g., Pattillo-McCoy 1998). In the organizational realm, culture provides a tool kit, or menu, from which entrepreneurial actors, acting as "skilled cultural operators" (Lounsbury and Glynn 2001), strategically choose different options and resources to achieve their goals (e.g., Rao 1994; Rindova et al. 2011).

2.4 Cultural Frame

A key concept in the dynamic constructivist view of culture is "frame". The notion of frame was initially introduced by Goffman (1974) to refer to "schemata of interpretation" that "allows individuals to locate, perceive, identify, and label" occurrences within their daily life (ibid, p. 21). Frame has been incorporated as a foundational concept into a broad range of theories. For example, frame is a key element of structuration theory, according to which, frames are "clusters of rules which help to constitute and regulate activities, defining them as activities of a certain sort and subject to a given range of sanctions" (Giddens 1984, p. 87), and frames allow "individuals to categorize an indefinite plurality of circumstances or situations as to be able to respond in an appropriate fashion to whatever is 'going on'" (ibid, p. 88).

The concept of "cultural frames" has been introduced in psychology to describe individuals' culturally-rooted meaning systems (Hong et al. 2000). Cultural frames enable individuals to interpret and respond to cultural cues. In the definition of cultural frame, culture is conceptualized as a dynamically-constructed "network of discrete, specific constructs that guide cognition" (ibid, p. 709). Individuals may possess multiple cultural frames and can switch from one frame to another in response to social stimuli (e.g., LaFromboise et al. 1993; Benet-Martinez et al. 2002; Fu et al. 2007). This process of cultural frame switching is central to individuals' "bicultural competence", that is, the ability to leverage cultural knowledge and engage in cross-cultural activities (Hong 2010). People with multiple cultural frames play important roles in today's businesses (Brannen and Thomas 2010). For example, bicultural individuals can enhance team performance (Hong 2010). It should be noted that although the research on cultural frame in psychology developed independently from Goffman (1974), the meaning of "frame" in psychology is consistent with Goffman's definition (Agar 1994; Stolte and Fender 2007).

Cultural frames provide individuals with the knowledge structures for "making sense of" various stimuli (Hong et al. 2000). Sensemaking has been extensively studied in organizational behavior (e.g., Starbuck and Milliken 1988; Harris 1994). It refers to the process by which "people develop some sort of sense regarding what they are up against, what their own position is relative to what they sense, and what they need to do" (Weick 1999, p. 42). This process involves an ongoing construction of coherent account of reality (Berger and Luckmann 1966; Strang and Meyer 1993). In cross-cultural settings, individuals engage in cultural sensemaking, a process through which individuals respond to social cues by making attributions and enacting appropriate behavior (Bird and Osland 2006; Osland et al. 2007). The process of cultural sensemaking plays the important role of linking cultural knowledge to contexts (Osland and Bird 2000), and serves as "a springboard into action" (Weick et al. 2005, p. 409).

The concept of "cultural frame" has also been extensively used in sociology to refer to a repertoire of cultural and ideological symbols and rhetoric that is often employed as tactics by individuals or organizations to initiate collective social actions (Swidler 1986). The process of "framing", including frame transformation, amplification, extension, and bridging (Snow et al. 1986), in particular, is a focus of recent research on social movements. The framing process is both deliberately planned and dynamically constructed (Benford and Snow 2000), and plays the critical role of connecting individual psychological factors with structural or organizational elements. The cultural frame perspective provides a valuable lens for analyzing many social, political, and economic phenomena (e.g., Hunt 1984; Rao et al. 2003; Fligstein and McAdam 2011). Although this definition of cultural frame does not focus on national culture, it also emphasizes the situation-specific, socially-constructed nature of culture (DiMaggio 1997).

Recently, the notion of frame has been increasingly applied to study organizations, especially organizations in transformation (Davis et al. 2005). This emerging stream of research conceptualizes organizational dynamics as social movements (Fligstein and McAdam 2011). In particular, at interorganizational level, changes in organizational fields can oftentimes be viewed as the result of institutional entrepreneurs' construction of cultural frames by leveraging and recombining diverse cultural resources (e.g., Rao 1998; Amburgey and Singh 2005; Perretti et al. 2008). At intraorganizational level, organizational changes can be viewed as emerging from organizational actors' strategic framing practices (e.g., Kaplan 2008; Rindova et al. 2011; Howard-Grenville et al. 2011). In these studies, culture is conceptualized as inherently strategic and contested, which is consistent with the aforementioned dynamic constructivist view.

2.5 China

To explore the development of intercultural capabilities in supplier organizations, this study focuses on Chinese IT service firms. Chinese IT service suppliers have several unique characteristics. First, compared to established Indian suppliers (e.g., Garud et al. 2006; Oshri et al. 2007), even the largest Chinese suppliers are significantly smaller and less mature. Limited "cultural compatibility" with Western clients remains a competitive disadvantage for Chinese suppliers (e.g., Gartner 2010). On the other hand, China is rapidly growing into a top global outsourcing destination outside of India (e.g., KPMG 2009). Major Chinese technology firms, including a small set of IT service firms, are rapidly moving up the global value chain and even becoming new multinationals (Dedrick and Kraemer 2006; Dedrick et al. 2010; IAOP 2012). In addition, with major clients from the world's three largest markets: U.S., China, and Japan, Chinese suppliers have accumulated significant technical and managerial experience in the Japanese and domestic markets and are actively improving their capabilities for the U.S. market.

China's overall business environment also has its idiosyncrasies. In the past three decades, as the country underwent a fundamental transformation toward a market economy (e.g., Peng 2003; Guthrie 2005), even cultural norms that were traditionally viewed as deeply embedded in the Chinese society, such as the importance of guanxi, have changed dramatically (e.g., Guthrie 1998). China also has high intranational diversity, with business environments and practices varying significantly across regions (e.g., Fan et al. 2009; McSweeney 2009). In China's economic transition, many firms served as outsourcing manufacturers for foreign firms (Child and Rodrigues 2005). In the recent several years, as the focus of the country's growth shifted from manufacturing to service, which tends to be more culturally-sensitive than manufacturing (Knight 1999), and as major Chinese IT service firms increasingly internationalized into new markets, acquiring the ability to work with multiple cultures became critical to Chinese suppliers.

To summarize, culture is a critical component of global outsourcing relationships. In such interorganizational relationships, the IT service supplier needs to possess the ability to interact with a portfolio of clients with diverse cultural backgrounds. The IS and related management literatures have identified a rich set of best practices for mitigating cultural barriers, and have prescribed the types of capabilities firms should possess in offshore outsourcing, but have not adequately elaborated how such firm-level intercultural capability is formed and evolves in practice, especially from the perspective of IT service suppliers. The IB and related strategy literatures have also examined the impact of national culture on firms' behavior and performance, and have provided frameworks for conceptualizing organizations' overall ability to manage inter-firm relationships, but offer limited specific insights into firms' intercultural capability.

Recent development in IS, IB, and management research suggests that understanding the micro-level cultural processes in offshore IT outsourcing requires moving beyond the traditional, predominant pan-cultural approach, and adopting a more situated, fine-grained view of culture. Related research in cultural psychology and sociology increasingly emphasizes a dynamic constructivist view of culture. Based on this perspective, the concept of cultural frame, in particular, provides a potential valuable theoretical lens for understanding cultural processes within and between organizations. Finally, Chinese IT service suppliers, with their broad client portfolios, limited but rapidly-improving intercultural capability, coupled with the country's diverse and dynamically-changing intra-national environment, provide an especially suitable context for exploring the formation and evolution of firms' intercultural capability. The goal of this study focuses on conceptualizing and elaborating the micro-level processes by which offshore IT service suppliers accumulate intercultural capability through their interaction with clients with different cultural backgrounds.

3 Research Methods

Since this research focuses on answering "how" and exploratory "what" questions regarding a contemporary phenomenon embedded in organizational practices, the case study methodology was selected (e.g., Benbasat et al. 1987; Yin 2003). Specifically, an in-depth, multiple-year, interview-based field study of SoftCo (pseudonym), one of the largest Chinese IT service firms, was conducted. The following section first introduces SoftCo's overall history, and then explains the process of data collection and analysis.

SoftCo was one of the largest and oldest IT service firms in China. From 2006 to 2011, the field researcher paid five visits to SoftCo's headquarters and largest subsidiary. Altogether sixty in-depth, face-to-face interviews with the firm's top and middle level managers were conducted and recorded. After the initial several interviews, SoftCo allowed the field researcher to systematically select interviewees. The selected informants spanned multiple organizational levels, from top executives, such as CEOs of the North American and Japanese subsidiaries, to middle-level managers such as project leaders. The informants covered all major business lines of the firm, including divisions servicing Japanese, U.S., and domestic markets, as well as corporate-wide functions such as human resources and business development. Some informants oversaw multiple divisions such as both Japanese and U.S. business lines. Snowball and opportunistic sampling was also utilized by the field researcher in some cases.

Each interview lasted between 45 min and 3 h, with the average length of approximately 1.5 h. The interviews were semi-structured. The field researcher mostly asked open-ended questions and let the informants describe and explain stories related to the firm's experience in different markets. An interview guide was used to ensure the completeness of data. The interview guide focused on the characteristics of the supplier's major markets and the supplier's perceptions and practices in different markets. The field researcher also collected secondary, archival data, including the firm's cultural training materials, marketing and sales presentations, press releases, annual reports, published news articles and books, and reports from industry and government entities. In summer 2009, the field researcher spent over a month living in SoftCo's "software park". The park was home to the firm's global headquarters, several software labs, employee training centers, as well as dorms for entry-level employees and residence of some top executives. At the park, the field researcher observed some of the firm's daily activities, such as manager training sessions, speeches of the firm's founder and CEO, product launch events, and international marketing meetings. In 2011, the field researcher attended SoftCo's largest anniversary gala in its history. During the celebration, SoftCo's earliest and most important clients were invited to review their experience with the firm. Another important source of information was the annual Chinese International Software and Information Service Fair (e.g., CISIS 2011), the leading IT outsourcing conference in China. The field researcher attended the fair in 2009, 2010, and 2011 to observe presentations and roundtable discussions of the top decision-makers of SoftCo and other major Chinese IT service suppliers. The immersion in the supplier's business activities allowed the field researcher to not only supplement and triangulate the interview data, but also acquire an intuitive, contextualized view of the supplier's internal and external environments.

Throughout data collection and analysis, the interview data were triangulated whenever possible. Including informants from different hierarchical levels, business lines, and functional areas helped reduce retrospective bias. Interview data were also triangulated with secondary data which provided valuable factual information on the firm's history. The interviews, including 56 conducted in Chinese, 3 in English, and 1 in Japanese through an interpreter, were recorded and transcribed by the field researcher, who was bilingual in English and Chinese. Data analysis consisted of two iterative phases: first, the development of the concept of cultural frame; second, the conceptualization of the process by which such cultural frames emerge and become embedded in the supplier. In the first phase, the concept of cultural frame was primarily constructed based on existing theories and informed by the data. Based on this concept, data analysis synthesized information obtained from different stakeholders across the firm to elaborate the content of cultural frames in the context of client-supplier relationships. In the second phase, the concept of cultural frames were applied to interpret the data and model the process of emergence and embedding of cultural frames. In the data analysis process, inductive techniques were applied to generate theoretical insights (Miles and Huberman 1994; Strauss and Corbin 1997; Patton 2002). Key principles of interpretive field studies (Klein and Myers 1999; Myers 2004; Gasson 2004) were followed. In particular, hermeneutic cycle as the fundamental principle of interpretive IS research was enforced through continuously linking findings to their contexts, generalizing findings by drawing on a wider theoretical base, demonstrating dialogical reasoning processes, considering multiple interpretations, and exercising suspicion by combining views from multiple perspectives (Klein and Myers 1999). The specific data analysis process will be demonstrated together with the presentation of findings in the following sections.

4 Supplier's Cultural Frames

In describing the characteristics of their clients, the most salient dimension the informants used to categorize their client bases was the clients' country of origin. All suppliers grouped their clients into three major markets: Japan, U.S., and the Chinese domestic market. A key reason for this grouping was that the supplier tended to apply a different "toolkit" of interpretive schemes and managerial practices to effectively manage interaction with clients from each of these markets. To conceptualize different aspects of the toolkit, Schreiner et al. (2009) provides a suitable framework which elaborates a firm's ability to manage inter-firm relationships into three dimensions;

each dimension has a direct impact on the outcomes of such relationships. These dimensions are coordination, communication, and bonding. This framework is consistent with the focus of firm-level cultural intelligence in the context of outsourcing and offshoring, which is on the "micro-interoganizational interface" between clients and suppliers (Ang and Inkpen 2008).

Building on prior research, Schreiner et al. (2009) define coordination as management of interdependence between partners by specifying and adapting each partner's roles and responsibilities in task execution; communication as formal and informal sharing of information between partners; bonding as development of a reciprocal trusting relationship through social integration between partners. These definitions are aligned with McEvily and Marcus' (2005) framework which identifies joint problem-solving, information sharing, and trust as the key factors of client-supplier interaction. McEvily and Marcus (2005) complement Schreiner et al. (2009) by showing that these three factors are also the mechanisms by which firms acquire capabilities through alliances. Interorganizational trust, in particular, is critical to client-supplier alliances. Such trust includes process-based trust resulting from procedures and routines, and relationship-based trust emerging from cross-firm social interaction (Dyer and Chu 2000, 2011).

Based on the synthesis of the above frameworks, this study compares the supplier's perceptions and practices during its interaction with clients from the three major markets, along the three key dimensions: coordination, communication, and bonding. Meanwhile, data analysis was conducted to refine each of the three dimensions by identifying the key areas in which the supplier's behavior differed across markets. Specifically, coordination encompasses two aspects: the level of autonomy assumed by the supplier in the joint execution of outsourced tasks, and the degree of interaction between the supplier and the client in these tasks; communication includes two aspects; the level of specificity of the information that the client shared with the supplier, and the scope, or breadth, of the content of the shared information; bonding focuses on establishing trust between the client and the supplier, which consists of trust based on rigorous, routinized interorganizational and intraorganizational processes, and trust based on the social relationship between the two parties' key stakeholders.

The comparison suggests that the type of capability required for the supplier to complete the outsourced tasks is conditioned on the cultural background of the client. For clients from each of the three markets, the supplier employees developed a shared set of interpretive schemes and practices in order to collectively make sense of and respond to the stimuli from the clients. Based on the dynamic constructivist view, which conceptualizes culture as a toolkit of habits, styles and practices for constructing strategies and actions, and drawing on the specific concept of individual-level cultural frame, this study defines the toolkit of interpretive schemes and practices shared by the supplier's employees for a particular market as the supplier's cultural frame. This toolkit enabled the supplier to effectively manage inter-firm interaction with clients with a certain cultural background.

4.1 Coordination

Coordination refers to the management of dependence between the client's and the supplier's activities (e.g., Malone and Crowston 1994) in the joint execution of outsourced tasks. Data analysis shows that when coordinating with clients from Japan, U.S., and China, the supplier employees assumed significantly different levels of autonomy and engaged in varied degrees of interaction with clients. After the two emergent aspects: supplier autonomy and supplier-client interaction were identified from data analysis, in subsequent interviews, the informants were asked to compare the three markets based on these two aspects. The results were highly consistent among informants across different levels and divisions, from top executives to middle-level project managers.

Specifically, when working with Japanese clients, the supplier tended to undertake a relatively passive role and was given a low level of autonomy in the outsourcing relationship. Meanwhile, the supplier mostly focused on specific tasks with clearly-defined boundaries and only engaged in limited interaction with the client. When working with U.S. clients, in contrast, the supplier played a more active role and was given a higher level of autonomy. The supplier was also expected to more proactively engage clients in joint problem-solving to interactively address clients' business needs. The difference between coordination with Japanese and U.S. clients was illustrated by the following quote.

> With U.S. clients, you need to have the ability to propose your own solutions…; otherwise, they won't recognize you. Sometimes you need to have the ability to plan new projects. You need to be more autonomous in management… whereas Japanese clients emphasize: 'if I want you to do this, just do it; if you follow your own ideas then you are wrong, you violated my rule.' [Director of a Division]

When working with domestic clients, the supplier had a highly active, autonomous role. Many domestic clients outsourced tasks to suppliers in order to access technical and managerial capabilities that the clients lacked internally. Moreover, in China's emerging economy, many domestic clients did not possess adequate capability to systematically manage interaction with suppliers. Therefore, the supplier often assumed full responsibility for independently designing and implementing solutions to address the clients' business needs.

4.2 Communication

Communication refers to the sharing of information between the client and the supplier. When communicating with clients from the three markets, the supplier developed significantly different perceptions and practices. Such difference concentrated in the level of specificity and the scope of information shared by the client. When communicating with Japanese clients, the supplier received instructions that were highly specific, detailed, and precise. The supplier had to strictly follow the specifications when performing the outsourced tasks. Adopting a highly cautious approach toward quality and intellectual property, Japanese clients tended to divide a task into several components and outsource each fragmented component to one supplier. As a result, the supplier only acquired a limited view of the overall outsourced product or process. When communicating with U.S. clients, the supplier also anticipated specific and detailed instructions, but could oftentimes obtain a more holistic view of the outsourced product.

> The Japanese have very formal, clear specifications, with elegant drawings and diagrams. Every detail has been thought through for you. U.S. clients sometimes give you specifications that are just a few sentences. [Director of a Division]

When communicating with domestic clients, the specifications and instructions from clients tended to be high-level, abstract, and oftentimes ambiguous. The supplier had to interpret and implement such requirements. Meanwhile, in contrast with Japanese and U.S. firms, which mostly outsourced to China to reduce cost and increase capacity, domestic clients outsourced IT-related tasks in order to access the supplier's specialized capabilities. These clients tended to delegate a broad portfolio of tasks to a single supplier. As a result, the supplier was able to not only obtain a holistic view of the outsourced products, but also acquire valuable information about the clients' overall business. For example, SoftCo provided services to a major Chinese telecommunications company. This client was one the largest telecommunications firms in the world, and had adopted a set of world-leading technologies and practices. Through the outsourcing relationship, SoftCo was gained access to such information and knowledge.

4.3 Bonding

Bonding in the context of offshore outsourcing centers on the development of a trusting relationship between the client and the supplier. Such interorganizational trust can be based on consistent processes that represent the supplier's commitment, and social or personal relationship between the two parties' key stakeholders (e.g., Dyer and Chu 2000, 2011). When working with clients from the three markets, the supplier relied on different approaches to develop trust. Specifically, when working with Japanese clients, in order to obtain and maintain trust, the supplier needed to adhere to a set of highly rigorous and stable processes throughout the projects. Meanwhile, it was also important for the supplier to develop personal ties with, and demonstrate commitment to, the client. For example, in many cases, the supplier needed to accommodate requirements that were beyond the scope of the contract, such as an unexpected, urgent order. In contrast, when working with U.S. clients, while the supplier also needed to comply with a set of standardized processes, relationship-based trust was not as pronounced.

> (Compared to U.S. clients) Japanese are a little similar to the Chinese culture. People bring a lot of personal feelings into projects. If you pursue a project in a manner that is confrontational or detached from personal bonds, the project will probably go wrong… [Marketing Director of an International Business Division]

When working with domestic clients, the supplier's focus was on achieving the desired end results for the clients and flexibly responding to the clients' rapidly changing business needs. Process was not emphasized in domestic projects, whereas building "guanxi", or affect-based, family-like social relationship (e.g., Yang 1994; Chua et al. 2009) with the clients' key stakeholders was critical to the development of inter-organizational trust in many cases.

To summarize, when interacting with clients from Japan, U.S., and China, the supplier developed different cultural frames, each comprising of a set of interpretive schemes and practices. These frames provided the supplier's employees with a toolkit to interpret and respond to client needs. Generally speaking, with Japanese clients, the supplier approached outsourcing as a passive process of adhering to specific instructions and complying with highly structured procedures. With U.S. clients, the supplier approached outsourcing as a more interactive process of actively engaging the client while still following process standards. With Chinese clients, the supplier approached outsourcing as an autonomous, result-driven, adaptive process in which social relationships played a critical role. It should be noted that in this study, nation is still used as a boundary for characterizing culture. In practice, the supplier's employees continuously adjusted and refined their cultural frames as they encountered new clients.

5 Emergence and Embedding of Cultural Frames

The supplier's portfolio of cultural frames emerged and became embedded in the organization during the supplier's interaction with clients from multiple markets. Based on the interview data and drawing on the research on framing, this section identifies and elaborates three specific sub-processes: frame enactment, alignment, and institutionalization.

5.1 Frame Enactment

The supplier possessed a repertoire of cultural elements that were leveraged to respond to different clients. The "cultural repertoire" was based on the prior cultural background, training, and experience of the supplier's individual members. When the supplier interacted with a client from a certain culture, elements from the cultural repertoire that resonated with the client's behavior were enacted. In particular, employees with bicultural or multicultural background played a critical role by engaging in "frame switching", which enabled these individuals to respond to clients' social cues in culturally consistent ways (Hong et al. 2000, 2001). In the Chinese economy, an important group of such people was the "sea turtles", that is, Chinese natives who had acquired significant education and working experience overseas and returned to China (Newsweek 2008). At SoftCo, many senior and middle level managers belonged to this category. For example, the Director of the International Software and Services Division was a Chinese native with over a decade's experience working in the U.S., first as a post-doctoral researcher at a university and then as a technologist at a U.S. multinational software company. After returning to China to join SoftCo, based on his prior experience, he was able to switch cultural frame when engaging U.S. clients. For example, he emphasized active and timely communication with clients, a practice critical to U.S. projects but less common in domestic projects. This cultural frame helped him successfully manage several large U.S. clients.

He also noticed the different cultural frame of his Chinese employees in communication.

> There is a cultural difference. Chinese engineers are relatively 'modest', not aggressive. They know things, but they don't speak. This creates problems in communication… They want to make sure that everything is 100% correct before speaking… But from the client's perspective… if there are some errors, not a problem… but you have to speak. If you don't speak, then that is your problem.

If an individual did not possess sufficient experience with the client's culture, the individual drew on elements from his or her existing cultural frames to make sense of Weick (1999) and respond to cues from clients. For example, a manager from the above division worked exclusively with Japanese clients for several years, before being transferred to U.S. projects. At the beginning of U.S. projects, the manager relied on habits and styles acquired in the Japanese market. For example, when receiving a request from a client, instead of actively communicating with the client to clarify the client's expectations, he immediately pursued intensive, in-depth research to create a set of high detailed deliverables for the client. While the client was impressed by his team's commitment, such efforts brought unnecessary burden and created inefficiency. Such examples occurred frequently at the beginning of the relationship.

According to the framework of firm-level intercultural capability in offshore outsourcing (Ang and Inkpen 2008), a firm's cultural intelligence is embedded in its individual managers and the firm's structural arrangements. The data suggested that the cultural frame enactment process was facilitated if the firm possessed high cultural intelligence. Specifically, if the firm could leverage managers with adequate cultural knowledge and skills, the firm could more effectively enact cultural frames that resonated with the client. Culturally intelligent individual managers, such as "biculturals", in particular, played a pivotal role in the cross-cultural interaction.

5.2 Frame Alignment

As can be seen in the previous examples, the enacted cultural frames “resonated” (Swidler 1986) to varied degrees with the supplier’s internal members and external clients. Internally, employees with different cultural background potentially constituted a source of frame misalignment. In the previous example, the Director’s cultural frame for the U.S. market, which emphasized client interaction and communication, was misaligned with the Project Manager’s initial cultural frame, which was mostly based on the manager’s prior experience in the Japanese market. Externally, clients with new or unique cultural background often generated frame misalignment. For example, the aforementioned Project Manager’s initial cultural frame which valued overinvestment in details was misaligned with the U.S. client. During the social interaction within the supplier and between the supplier and the client, if either or both of the internal and external resonance were low, difficulty and conflict arose, and the process of frame alignment took place. In this process, individuals’ existing cultural frames were modified until alignment was achieved both within the supplier and between the supplier and the client. Drawing on Snow et al.’s (1986) taxonomy, the frame alignment activities can be categorized into four processes: transformation, amplification, extension, and bridging.

Frame Transformation: This refers to changing initial understanding and meanings of a frame, and is also called “keying” by Goffman (1974). In this process, some elements of a cultural frame are deleted or replaced with new ones. This process often requires “a systematic alteration” of an existing, established frame so that individuals can interpret a phenomenon into “something quite else” (ibid, p. 45). At SoftCo, the supplier’s interaction with new foreign clients and recruitment of employees with different cultural background had both functioned as catalyst for frame transformation.

For example, in the aforementioned International Software and Services Division, the Director facilitated the transformation of the supplier team’s cultural frame, as illustrated by the quote below. Repeated client-supplier interaction eventually modified team members’ frames and led to internal and external resonance. In the following years, the division recorded high performance in the U.S. market. The aforementioned Project Manager, who was transferred from the Japanese to the U.S. market, had also grown into a stellar manager in the U.S. market.

> (The client) told me that they (the Chinese project team) don’t talk much, and are not very open. Then he said he discovered that they were too precise, too serious, wanting to make sure 100% correctness… So, (in subsequent client meetings) I said to my team: ‘if you have any questions or ideas, just raise them. Just tell me in Chinese… The team ended up raising many sharp questions. [Director of International Software and Services Division]

Frame Amplification: This refers to “the clarification and invigoration of an interpretive frame” (Snow et al. 1986, p. 469). In this process, some elements of an existing cultural frame are strengthened and become more salient. In contrast to frame transformation, in which the original cultural frame is significantly reconstructed, this process only involves the elevation of certain components of the original cultural frame.

For example, in another division of SoftCo, before starting a relationship with a Japanese client, the supplier's engineers already learned about the detail-oriented style of Japanese clients. However, when the supplier actually started working with the Japanese client, the employees discovered that the client was even more demanding in terms of quality of details than expected. The initial frame was further amplified as the supplier accumulated more experience.

> You have to make no mistake in any word of any document; no mistake in any punctuation mark; no mistake in the color; no mistake in the type of paper for printing, size, and font. Any mistake is a mistake. The client pays attention to every detail. [Project Director of a Division]

Frame Extension: This refers to the broadening of the boundaries of an existing frame. In this process, some auxiliary elements are incorporated into an individual's cultural frame. Different from frame amplification, in which some elements of the existing cultural frame are elevated, this process integrates elements that are not present in the original frame.

For example, during its interaction with several long-term, strategic Japanese clients, SoftCo's managers' perception of Japanese firms' tendency to micromanage and tightly control the supplier started to resonate less with the external situation. The initial frame was then extended by the knowledge that more self-management would be expected from the Chinese supplier once a highly stable, trusting relationship emerged between the two parties.

> Japanese clients are more detail-driven, especially in the early phase of the collaboration. But after long-term collaboration with Japanese clients, you don't need to spend too much effort (in dealing with the clients' micromanagement). You can take actions on your own … they will only directly contact your client representative. [Human Resource General Manager]

Frame Bridging: This refers to the linkage of two or more unconnected frames. In this process, one cultural frame incorporates elements from another frame. This process can potentially generate innovative knowledge and practices. At SoftCo, certain organizational arrangements, including rotating personnel between multiple markets, setting up cross-market business units such as firm-wide research and development centers, and sharing knowledge and practices across the firm, all facilitated the frame bridging process.

For example, a division director who had significant experience managing both Japanese and U.S. projects explained that some individuals who had worked with both Japanese and U.S. clients had learned to creatively integrate different styles, knowledge, and practices across markets. Such cross-pollination of cultural frames enhanced the supplier's resonance with clients from both markets, and created a competitive advantage for the firm.

> We borrow things (from one market) and apply to another. Sometimes this can generate competitive advantages for us. For example, in U.S. business we are able to strengthen our management of details… On the other hand, with Japanese clients, we are strengthening our

ability to propose new plans (by drawing on experience from the U.S. market)… [Director of a Division]

The above four "framing" processes characterize the pattern of evolution of individuals' cultural frames within the supplier. The framing processes are enabled by the process of interpretive sensemaking (Fiss and Hirsch 2005). Sensemaking involves an "internal, self-conscious process of developing a coherent account of what is going on" (ibid, p. 31; Weick 1999; Weick et al. 2005). In cross-cultural environments, "cultural sensemaking" links cultural knowledge with contexts (Osland and Bird 2000). A fine-grained view of a "culturally-guided sensemaking" process has been developed by Harris (1994). According to Harris (1994), "in the social setting of organizations, individuals make sense out of their experiences based in large part on the outcomes of contrived mental dialogues between themselves" … "and other contextually-relevant (past or present; real or imagined) individuals or groups." (ibid, p. 309). In this process, the cultural context shapes the common sensemaking across a set of organizational members, and becomes manifested within the organization.

At SoftCo, this self-conscious sensemaking process, including both individual sensemaking and group-based sensemaking, facilitated frame alignment across the organization. For example, many project managers emphasized the importance of the practice of "introspection", or self-examination, which was routinely employed during the supplier's interaction with clients. This practice was especially valuable at the beginning of the relationship, when there was significant frame misalignment between the client and the supplier.

Based on the definition of cultural intelligence (Ang and Inkpen 2008), this study shows that the process of cultural frame alignment was facilitated by firm-level cultural intelligence. Specifically, individual managers with high intercultural capability tended to engage in active cultural sensemaking, which contributed to the alignment and resonance of cultural frames. Culturally-intelligent structural arrangements, such as organizational units devoted to cross-market knowledge sharing, also assisted in effective frame alignment. It is worth noting that the process of frame alignment, in turn, enabled individuals within the supplier to improve their cultural knowledge and therefore enhanced the supplier's overall cultural intelligence.

5.3 Frame Institutionalization

The cultural frames that emerged through interaction between the supplier and the clients and among the supplier's employees became embedded and manifested in a set of relatively durable artifacts and stable structures (e.g., Lanzara and Patriotta 2007; Vaast and Levina 2006) within the supplier organization. These elements, in turn, enriched the cultural repertoire which the supplier drew on for future client interaction. Data analysis highlighted three sets of such elements, including: the implementation of processes, the design of roles and jobs, and the creation of artifacts.

Each of the three major markets: Japan, U.S., and China, led to significantly different characteristics in these three aspects within the supplier.

Process: Process encompasses routines that accomplish the execution and delivery of outsourced tasks (Levina and Ross 2003; Jarvenpaa and Mao 2008). The supplier adopted different process standards for the three markets. For Japanese clients, the supplier implemented and followed a set of highly structured and repeatable processes. These processes included not only industry-wide standards such as various CMM (Capability Maturity Model) and ISO (International Organization for Standardization) models, but also the clients' firm-specific process standards. Through working with Japanese clients, a set of highly detail-oriented, quality-driven practices, such as meticulously examining and continuously improving the quality of deliverables, was also formed within the supplier.

For U.S. clients, standardized processes were also often required to be implemented, although the supplier was given significantly more flexibility in the execution of these processes. Chinese clients, in contrast, paid little attention to process management. The focus was on the outcome of outsourcing. The actual process of executing the outsourced tasks was decided by the supplier and was highly flexible. Many domestic clients did not possess sufficient capability to rigorously manage outsourced IT projects. Some clients even intentionally avoided process standards. In response, sometimes the supplier needed to creatively adapt and improvise processes to meet the changing needs of domestic clients.

Role: Role includes the responsibility and the required skill set of the supplier's employees. For the three major markets, the supplier designed and specified different roles for its employees. For Japanese clients, most of the supplier employees' role focused on following a set of standard processes to perfect a task of limited scope, allowing little autonomy among the employees. This role usually required a relatively narrow set of technical skills. One special role in Japanese projects was a small group of "bridge engineers", who were responsible for communicating with the clients. For U.S. clients, the supplier's employees were expected to play an autonomous role by proactively interacting with clients to address their business needs. This role required a broader set of capabilities including both technical and client-facing skills.

For Chinese clients, the supplier's personnel needed to assume an active, leadership role by simultaneously responding to and shaping the client's expectations and behaviors. This role required a broad, all-rounded skill set, including not only technical skills and domain knowledge, but also the ability to build interpersonal and business relationship with clients. The ability to develop and leverage guanxi, or social relationships, was critical in some domestic projects.

Artifact: Artifacts provide organizational members with a set of cognitive tools to make sense of and respond to their environment. The use of two types of artifact was especially salient in supplier-client interaction: documentation and prototype. For Japanese clients, the supplier developed highly detailed documentation of both the processes and outcomes of the outsourced task. In order to achieve the continuous learning and improvement, the supplier also created its own internal documentation as a knowledge repository. Such documentation included a set of shared, comprehensive checklists recording best practices for dealing with different clients.

For U.S. clients, the supplier also maintained documentation but the level of details and specificity tended to be lower compared to that in Japanese projects. Since U.S. clients expected the supplier to be able to independently propose new solutions, the supplier sometimes prepared prototypes in advance in order to demonstrate to the clients. For Chinese domestic clients, the supplier's focus was on clients' satisfaction with the final output, whereas documentation of the development process was not emphasized. Given domestic clients' oftentimes ambiguous requirements and focus on visible output, prototypes were strategically used throughout different phases of projects to negotiate project details with the clients and shape the clients' expectations of the final outcome.

The institutionalized cultural elements reflected the degree of "taken for grantedness" of the supplier's idiosyncratic cultural knowledge (Tsoukas and Vladimirou 2001). The process of institutionalization centered on individuals' and collectivities' retrospective rationalization of their experience with different clients. This process, which involved extracting cues and constructing plausible images, constituted a process of sensemaking (e.g., Weick et al. 2005). In order to facilitate sensemaking activities, the supplier created various structures and routines. In particular, at the firm level, SoftCo established a division specialized in identifying and synthesizing best practices from different business lines and disseminating these practices across the organization. These resulting cultural elements: processes, roles, and artifacts, propagated the institutionalized cultural frames to the firm level.

The process of institutionalization occurred at multiple organizational levels, from project team to business unit and eventually the firm level. According to the definition of cultural intelligence (Ang and Inkpen 2008), the above organizational mechanisms that had facilitated the process of frame institutionalization reflected the firm's structural cultural intelligence. The overall process of the emergence and embedding of cultural frames is illustrated in Fig. 1. In this model, cultural frame bridges the high-level conceptualization of cultural intelligence with the underlying process of cultural sensemaking. The model suggests that the supplier's cultural intelligence contributed to the emergence and embedding of cultural frames. These institutionalized cultural frames, in turn, enhanced the supplier's firm-level cultural intelligence.

6 Conclusions and Discussion

Drawing on recent development in cultural psychology and sociology, this study seeks to unpack the cognitive process underlying offshore IT service suppliers' development of cultural intelligence. Based on the dynamic constructivist view of culture, especially the construct of cultural frame as a key mechanism in cultural cognition, the study develops the concept of cultural frame to refer to a portfolio of shared interpretive schemes and practices which enable the supplier to make sense of and respond to expectations and actions of clients with different cultural backgrounds. The supplier's cultural frames originated from the supplier employees' individual cultural knowledge. During repeated interorganizational and intra-organizational interaction,

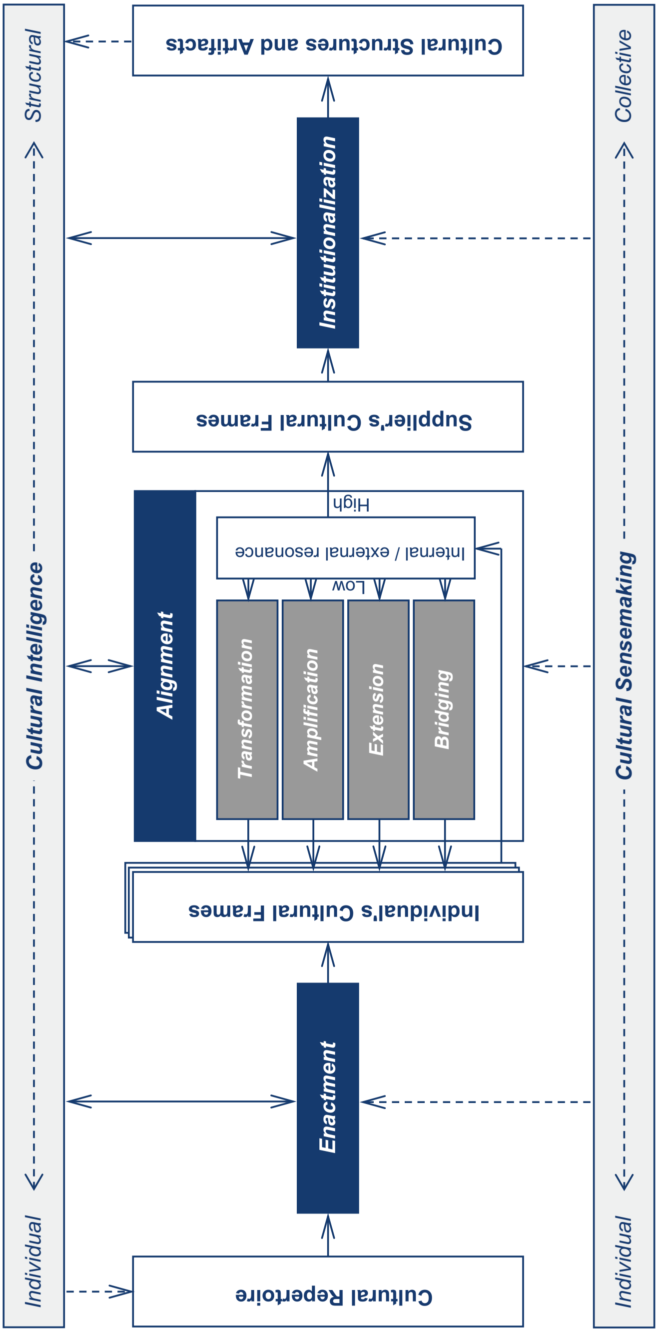

Fig. 1 Emergence and embedding of cultural frames within the supplier

individual cultural frames were continuously enacted, aligned and realigned, and eventually institutionalized into a set of cultural structures and artifacts in the supplier. This ongoing, iterative process was enabled by individual and collective sensemaking among the supplier's employees. The effectiveness of this emergence and embedding process was facilitated by, and in turn increased, the supplier's firm-level cultural intelligence.

This study contributes to the IS and management literatures in three main ways. First, this study elaborates how firm-level cultural intelligence emerges and evolves within offshore IT service suppliers, especially those in an early and expansionary phase of their lifecycle. Second, this study complements the dominant pan-cultural approach by incorporating the alternative, dynamic constructivist view of culture into the research on IS outsourcing. Third, this study builds on the convergence of cultural psychology and sociology, and integrates the concept of "cultural frame" with the process of "framing". The resulting model connects micro-level individual cognition with macro-level organizational capability, and assists to understand the microfoundations (Abell et al. 2008; Eisenhardt et al. 2010) of cultural intelligence.

This study has its limitations. The case study focused on one China-based supplier; some of the supplier's practices might be specific to this firm and the Chinese IT service industry. However, the concept of cultural frame and the process model are applicable to any supplier that seeks to acquire cultural knowledge through interacting with diverse clients. Moreover, the characterization of the supplier's cultural frames was mostly based on national boundaries. More fine-grained analysis of suppliers' interaction with clients from different regions within these markets and from other countries should be conducted in the future.

Future research can also explore several other directions. First, the study can include established suppliers from other markets that have already developed sophisticated intercultural capability; these firms can provide a comparison with the Chinese suppliers. Second, future research can apply quantitative measurement of the supplier's cultural intelligence to test some effects derived in this study, such as the interaction between the supplier's cultural intelligence and the emergence and embedding process of cultural frames. Third, future studies can explore the emergence and evolution of cultural intelligence from the client's perspective.

References

Abell, P., Felin, T., & Foss, N. (2008). Building micro-foundations for the routines, capabilities, and performance links. *Managerial and Decision Economics, 29*(6), 489–502.

Agar, M. (1994). The intercultural frame. *International Journal of Intercultural Relations, 18*(2), 221–237.

Amburgey, T. L., & Singh, J. V. (2005). Organizational evolution. In J. A. C. Baum (Ed.), *Companion to organizations* (pp. 327–343). Malden, MA: Blackwell Publishing.

Ang, S., & Inkpen, A. C. (2008). Cultural intelligence and offshore outsourcing success: A framework of firm-level intercultural capability. *Decision Sciences, 39*(3), 337–358.

Ang, S., & Van Dyne, L. (Eds.). (2008). *Handbook of cultural intelligence: Theory, measurement, and applications*. Armonk, NY: M.E. Sharpe.
Ang, S., Van Dyne, L., Koh, C., Ng, K. Y., Templer, K. J., Tay-Lee, S. L. C., et al. (2007). Cultural intelligence: Its measurement and effects on cultural judgment and decision making, cultural adaptation and task performance. *Management and Organization Review, 3*(3), 331–357.
Aubert, B. A., Patry, M., & Rivard, S. (2005). A framework for information technology outsourcing risk management. *The DATA BASE for Advances in Information Systems, 36*(4), 9–28.
Avison, D. E., & Myers, M. D. (1995). Information systems and anthropology: An anthropological perspective on IT and organizational culture. *Information Technology & People, 8*(3), 43–56.
Avison, D. E., & Banks, P. (2008). Cross-cultural (mis)communication in IS offshoring: Understanding through conversation analysis. *Journal of Information Technology, 23*(4), 249–268.
Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly, 11*(3), 369–386.
Benet-Martínez, V., Leu, J., Lee, F., & Morris, M. W. (2002). Negotiating biculturalism: Cultural frame switching in biculturals with oppositional versus compatible cultural identities. *Journal of Cross-Cultural Psychology, 33*(5), 492–516.
Benford, R. D., & Snow, D. A. (2000). Framing processes and social movements: An overview and assessment. *Annual Review of Sociology, 26*(1), 611–639.
Bensaou, M., & Anderson, E. (1999). Buyer-supplier relations in industrial markets: When do buyers risk making idiosyncratic investments? *Organization Science, 10*(4), 460–481.
Berger, P. L., & Luckmann, T. (1966). *The social construction of reality*. Garden City, NY: Doubleday.
Bird, A., & Osland, J. S. (2006). Making sense of intercultural collaboration. *International Journal of Management and Organizations, 3*(4), 115–132.
Birkinshaw, J., Brannen, M. Y., & Tung, R. L. (2011). From a distance and generalizable to up close and grounded: Reclaiming a place for qualitative methods in international business research. *Journal of International Business Studies, 42,* 573–581.
Brannen, M. Y. (2004). When Mickey loses face: Recontextualization, semantic fit, and the semiotics of foreignness. *Academy of Management Review, 29*(4), 593–616.
Brannen, M. Y., & Salk, J. E. (2000). Partnering across borders: Negotiating organizational culture in a German-Japanese joint venture. *Human Relations, 53*(4), 451–487.
Brannen, M. Y., & Thomas, D. C. (2010). Bicultural individuals in organizations: Implications and opportunity. *International Journal of Cross Cultural Management, 10*(1), 5–16.
Bruner, J. S. (1990). *Acts of meaning*. Cambridge, MA: Harvard University Press.
Capar, N., & Kotabe, M. (2003). The relationship between international diversification and performance in service firms. *Journal of International Business Studies, 34*(4), 345–355.
Carmel, E., & Agarwal, R. (2001). Tactical approaches for alleviating distance in global software development. *IEEE Software, 18*(2), 22–29.
Carmel, E., Gao, G., & Zhang, N. (2008). The maturing Chinese offshore IT services industry: It takes 10 years to sharpen a sword. *MIS Quarterly Executive, 7*(4), 157–170.
Chen, Y. R., Leung, K., & Chen, C. C. (2009). Bringing national culture to the table: Making a difference with cross-cultural differences and perspectives. *Academy of Management Annals, 3*(1), 217–249.
Chen, X., Liu, D., & Portnoy, R. (2012). A multilevel investigation of motivational cultural intelligence, organizational diversity climate, and cultural sales: Evidence from U.S. real estate firms. *Journal of Applied Psychology, 97*(1), 93–106.
Child, J., & Rodrigues, S. B. (2005). The internationalization of Chinese firms: A case for theoretical extension? *Management and Organization Review, 1*(3), 381–410.
Chua, R. Y. J., Morris, M. W., & Ingram, P. (2009). Guanxi vs networking: Distinctive configurations of affect- and cognition-based trust in the networks of Chinese vs American managers. *Journal of International Business Studies, 40,* 490–508.
CISIS (China International Software and Information Service Fair). (2011). About CISIS. http://www.cisis.com.cn/News/735/2298/Newsdis2967.aspx.

Cramton, C., & Hinds, P. (2007). intercultural interaction in distributed teams: Salience of and adaptations to cultural differences. In *Academy of Management Best Paper Proceedings*, Philadelphia, PA.

Currall, S. C., & Inkpen, A. C. (2002). A multilevel approach to trust in joint ventures. *Journal of International Business Studies, 33*(3), 479–495.

Davis, G. F., McAdam, D., Scott, W. R., & Zald, M. N. (2005). *Social movements and organization theory*. Cambridge, UK: Cambridge University Press.

Dedrick, J., & Kraemer, K. L. (2006). Is production pulling knowledge work to China? A study of the notebook PC industry. *IEEE Computer, 39*(7), 36–42.

Dedrick, J., Kraemer, K. L., & Linden, G. (2010). Who profits from innovation in global value chains?: A study of the iPod and notebook PCs. *Industrial and Corporate Change, 19*(1), 81–116.

Dibbern, J., Winkler, J., & Heinzl, A. (2008). Explaining variations in client extra costs between software projects offshored to India. *MIS Quarterly, 32*(2), 333–366.

Dibbern, J., Chin, W. W., & Heinzl, A. (2012). Systemic determinants of the information systems outsourcing decision: A comparative study of German and United States firms. *Journal of the Association for Information Systems, 13*(6), 466–497.

DiMaggio, P. (1997). Culture and cognition. *Annual Review of Sociology, 23,* 263–287.

Doherty, N. F., & Doig, G. (2003). An analysis of the anticipated cultural impacts of the implementation of data warehouses. *IEEE Transactions on Engineering Management, 50*(1), 78–88.

Doney, P. M., Cannon, J. P., & Mullen, M. R. (1998). Understanding the influence of national culture on the development of trust. *Academy of Management Review, 23*(3), 601–620.

Du, W., & Pan, S. L. (2010). Boundary spanning by design: Insights from a vendor perspective. In *Proceedings of the 31st International Conference on Information Systems (ICIS)*, Paper 222.

Dyer, J. H., & Chu, W. (2000). The determinants of trust in supplier-automaker relationships in the U.S., Japan, and Korea. *Journal of International Business Studies, 31*(2), 259–285.

Dyer, J. H., & Chu, W. (2011). The determinants of trust in supplier-automaker relationships in the U.S., Japan, and Korea: A retrospective. *Journal of International Business Studies, 42,* 28–34.

Earley, P. C., & Ang, S. (2003). *Cultural intelligence: Individual interactions across cultures*. Palo Alto, CA: Stanford University Press.

Eisenhardt, K. M., Furr, N. R., & Bingham, C. B. (2010). Microfoundations of performance: balancing efficiency and flexibility in dynamic environments. *Organization Science, 21*(6), 1263–1273.

Ethiraj, S., Kale, P., Krishnan, M., & Singh, J. (2005). Where do capabilities come from and how do they matter? A study in the software services industry. *Strategic Management Journal, 26*(1), 25–45.

Fan, J., Huang, J., Morck, R., & Yeung, B. (2009). Vertical integration, institutional determinants and impact: Evidence from China. *NBER Working Paper*, No. 14650.

Fischer, R. (2006). Congruence and functions of personal and cultural values: Do my values reflect my culture's values? *Personality and Social Psychology Bulletin, 32,* 1419–1431.

Fiss, P. C., & Hirsch, P. M. (2005). The discourse of globalization: Framing and sensemaking of an emerging concept. *American Sociological Review, 70*(1), 29–52.

Fligstein, N., & McAdam, D. (2011). Toward a general theory of strategic action fields. *Sociological Theory, 29*(1), 1–26.

Ford, D. P., Connelly, C. E., & Meister, D. B. (2003). Information systems research and Hofstede's culture's consequences: An uneasy and incomplete partnership. *IEEE Transactions on Engineering Management, 50*(1), 8–25.

Fu, H. J., Chiu, C., Morris, M. W., & Young, J. M. (2007). Spontaneous inferences from cultural cues: Varying responses of cultural insiders and outsiders. *Journal of Cross-Cultural Psychology, 38*(1), 58–75.

Gartner. (2010). Gartner says India still no. 1 destination for offshore services. http://www.gartner.com/it/page.jsp?id=1502714.

Garud, R., Kumaraswamy, A., & Sambamurthy, V. (2006). Emergent by design: Performance and transformation at Infosys Technologies. *Organization Science, 17*(2), 277–286.

Gasson, S. (2004). Rigor in grounded theory research: An interpretive perspective on generating theory from qualitative field studies. In M. Whitman & A. Woszczynski (Eds.), *The Handbook of Information Systems Research* (pp. 79–102). Hershey, PA: Idea Group.

Gefen, D., & Carmel, E. (2008). Is the world really flat? A look at offshoring at an online programming marketplace. *MIS Quarterly, 32*(2), 367–384.

Gelfand, M. J., Nishii, L. H., & Raver, J. L. (2006). On the nature and importance of cultural tightness-looseness. *Journal of Applied Psychology, 91*(6), 1225–1244.

Gilbert, D. T. (1991). How mental systems believe. *American Psychologist, 46*(2), 107–119.

Giddens, A. (1984). *The constitution of society: Outline of the theory of structuration*. Berkeley, CA: University of California Press.

Goffman, E. (1974). *Frame analysis: An essay on the organization of experience*. Cambridge, MA: Harvard University Press.

Goodall, K. (2002). Managing to learn: From cross-cultural theory to management education practice. In M. Warner & P. Joynt (Eds.), *Managing across cultures: Issues and perspectives* (2nd ed., pp. 256–268). London: Thomson Learning.

Gould, S. J., & Grein, A. F. (2009). Think glocally, act glocally: A culture-centric comment on Leung, Bhagat, Buchan, Erez and Gibson (2005). *Journal of International Business Studies, 40,* 237–254.

Guthrie, D. J. (1998). The declining significance of Guanxi in China's economic transition. *The China Quarterly, 154,* 254–282.

Guthrie, D. (2005). Organizational learning and productivity: State structure and foreign investment in the rise of the Chinese corporation. *Management and Organization Review, 1*(2), 165–195.

Hahn, E. D., Doh, J. P., & Bunyaratavej, K. (2009). The evolution of risk in information systems offshoring: The impact of home country risk, firm learning, and competitive dynamics. *MIS Quarterly, 33*(3), 597–616.

Harris, S. G. (1994). Organizational culture and individual sensemaking: A schema-based perspective. *Organization Science, 5*(3), 309–321.

Heeks, R., Krishna, S., Nicholson, B., & Sahay, S. (2001). Synching or sinking: Global software outsourcing relationships. *IEEE Software, 18*(2), 54–60.

Herskovitz, M. J. (1955). *Cultural anthropology*. New York, NY: Knopf.

Hinds, P., Liu, L., & Lyon, J. (2011). Putting the global in global work: An intercultural lens on the practice of cross-national collaboration. *The Academy of Management Annals, 5*(1), 135–188.

Hofstede, G. (1980). *Culture's consequences*. Beverly Hills, CA: Sage.

Hofstede, G. (1991). *Cultures and organizations*. London, UK: McGraw-Hill.

Hong, H. (2010). Bicultural competence and its impact on team effectiveness. *International Journal of Cross Cultural Management, 10*(1), 93–120.

Hong, Y. Y., & Chiu, C. Y. (2001). Toward a paradigm shift: From cross-cultural differences in social cognition to social-cognitive mediation of cultural differences. *Social Cognition*, *19*(3), 181–196.

Hong, Y. Y., Morris, M. W., Chiu, C. Y., & Benet-Martinez, V. (2000). Multicultural minds: A dynamic constructivist approach to culture and cognition. *American Psychologist, 55,* 709–720.

Hong, Y. Y., Ip, G., Chiu, C. Y., Morris, M., & Menon, T. (2001). Cultural identity and dynamic construction of the self: Collective duties and individual rights in CHINESE AND American cultures. *Social Cognition, 19,* 251–268.

House, R. J., Hanges, P. J., Javidan, M., Dorfman, P. W., & Gupta, V. (Eds.). (2004). *Culture, leadership, and organizations*. Thousand Oaks, CA: Sage.

Howard-Grenville, J., Golden-Biddle, K., Irwin, J., & Mao, J. (2011). Liminality as cultural process for cultural change. *Organization Science, 22*(2), 522–539.

Hunt, L. (1984). *Politics, culture, and class in the French revolution*. Berkeley, CA: University of California Press.

Hymer, S. H. (1976). *The international operations of national firms: A study of direct foreign investment*. Cambridge, MA: MIT Press.
IAOP (International Association of Outsourcing Professionals). (2008). The 2008 global outsourcing 100. http://www.iaop.org/content/23/152/1619/.
IAOP (International Association of Outsourcing Professionals). (2009). The 2009 global outsourcing 100. http://www.iaop.org/content/23/152/1858/.
IAOP (International Association of Outsourcing Professionals). (2010). The 2010 global outsourcing 100. http://www.iaop.org/Content/19/165/3179.
IAOP (International Association of Outsourcing Professionals). (2011). The 2011 global outsourcing 100. http://www.iaop.org/content/23/152/2040.
IAOP (International Association of Outsourcing Professionals). (2012). The 2012 global outsourcing 100. http://www.iaop.org/content/19/165/3437.
Inkpen, A. C., & Beamish, P. W. (1997). Knowledge, bargaining power, and the instability of international joint ventures. *Academy of Management Review, 22*(1), 177–202.
Jarvenpaa, S. L., & Leidner, D. E. (1999). Communication and trust in global virtual teams. *Organization Science, 10*(6), 791–815.
Jarvenpaa, S. L., & Mao, J. Y. (2008). Operational capabilities development in mediated offshore software services models. *Journal of Information Technology, 23*(1), 3–17.
Johanson, J., & Wiedersheim-Paul, F. (1975). The internationalization of the firm: Four Swedish cases. *Journal of Management Studies, 12*(3), 305–322.
Johanson, J., & Vahlne, J. E. (1977). The internationalization process of the firm: A model of knowledge development and increasing foreign market commitment. *Journal of International Business Studies, 8*(1), 23–32.
Johanson, J., & Vahlne, J. E. (2009). The uppsala internationalization process model revisited: From liability of foreignness to liability of outsidership. *Journal of International Business Studies, 40*(9), 1411–1431.
Kaiser, K., & Hawk, S. (2004). Evolution of offshore software development: From outsourcing to cosourcing. *MIS Quarterly Executive, 3*(2), 69–81.
Kale, P., & Singh, H. (2007). Building firm capabilities through learning: The role of the alliance learning process in alliance capability and firm-level alliances success. *Strategic Management Journal, 28*(10), 981–1000.
Kaplan, S. (2008). Framing contests: Strategy making under uncertainty. *Organization Science, 19*(5), 729–752.
Kappos, A., & Rivard, S. (2008). A three-perspective model of culture, information systems, and their development and use. *MIS Quarterly, 32*(3), 601–634.
Katsikeas, C. S., Skarmeas, D., & Bello, D. C. (2009). Developing successful trust-based international exchange relationships. *Journal of International Business Studies, 40,* 132–155.
King, W. R., & Torkzadeh, G. (2008). Information systems offshoring: Research status and issues. *MIS Quarterly, 32*(2), 205–225.
Klein, H. K., & Myers, M. D. (1999). A set of principles for conducting and evaluating interpretive field studies in information systems. *MIS Quarterly, 23*(1), 67–94.
Knight, G. (1999). International services marketing: Review of research, 1980–1998. *Journal of Services Marketing, 13*(4/5), 347–360.
Koh, C., Ang, S., & Straub, D. W. (2004). IT outsourcing success: A psychological contract perspective. *Information System Research, 15*(4), 356–373.
KPMG. (2009). A new dawn—China's emerging role in global outsourcing. http://www.kpmg.com/CN/en/IssuesAndInsights/ArticlesPublications/Documents/china-global-outsourcing-O-0904.pdf.
Krishna, S., Sahay, S., & Walsham, G. (2004). Managing cross-cultural issues in global software outsourcing. *Communications of the ACM, 47*(4), 62–66.
Lacity, M. C., Khan, S., Yan, A., & Willcocks, L. P. (2010). A review of the IT outsourcing empirical literature and future research directions. *Journal of Information Technology, 25,* 395–433.

LaFromboise, T., Coleman, H., & Gerton, J. (1993). Psychological impact of biculturalism: Evidence and theory. *Psychological Bulletin, 114,* 395–412.
Lane, P. J., Salk, J. E., & Lyles, M. A. (2001). Absorptive capacity, learning, and performance in international joint ventures. *Strategic Management Journal, 22*(12), 1139–1161.
Lanzara, G. F., & Patriotta, G. (2007). The institutionalization of knowledge in an automotive factory: Templates, inscriptions, and the problem of durability. *Organization Studies, 28*(5), 635–660.
Lau, I. Y., Chiu, C., & Lee, S. (2001). Communication and shared reality: Implications for the psychological foundations of culture. *Social Cognition, 19,* 350–371.
Lavie, D., & Miller, S. R. (2008). Alliance portfolio internationalization and firm performance. *Organization Science, 19*(4), 623–646.
Leidner, D. E., & Kayworth, T. (2006). A review of culture in information systems research: Toward a theory of information technology culture conflict. *MIS Quarterly, 30*(2), 357–399.
Leonardi, P. M., & Bailey, D. (2008). Transformational technologies and the creation of new work practices: Making implicit knowledge explicit in task-based offshoring. *MIS Quarterly, 32*(2), 159–176.
Leung, K., Bhagat, R. S., Buchan, N. R., Erez, M., & Gibson, C. B. (2005). Culture and international business: recent advances and their implications for future research. *Journal of International Business Studies, 36,* 357–378.
Leung, K., Bhagat, R., Buchan, N. R., Erez, M., & Gibson, C. B. (2011). Beyond national culture and culture-centricism: A reply to Gould and Grein (2009). *Journal of International Business Studies, 42,* 177–181.
Levina, N., & Ross, J. (2003). From the vendor's perspective: Exploring the value proposition in IT outsourcing. *MIS Quarterly, 27*(3), 331–364.
Levina, N., & Vaast, E. (2005). The emergence of boundary spanning competence in practice: Implications for implementation and use of information systems. *MIS Quarterly, 29*(2), 335–363.
Levina, N., & Su, N. (2008). Global multisourcing strategy: The emergence of a supplier portfolio in services offshoring. *Decision Sciences, 39*(3), 541–570.
Levina, N., & Vaast, E. (2008). Innovating or doing as told? Status differences and overlapping boundaries in offshore collaboration. *MIS Quarterly, 32*(2), 307–332.
Levina, N., Gallivan, M., Su, N., Pan, S. L., Hsu, J., & Mao, J. Y. (2011). Studying China's IT services industry: Generalizations, particularities, and the competitive outlook. In *Proceedings of the 32nd International Conference on Information Systems (ICIS)*, Paper 6.
Lounsbury, M., & Glynn, M. A. (2001). Cultural entrepreneurship: Stories, legitimacy, and the acquisition of resources. *Strategic Management Journal, 22*(6/7), 545–564.
Luo, Y., & Shenkar, O. (2011). Toward a perspective of cultural friction in international business. *Journal of International Management, 17,* 1–14.
MacDuffie, J. P. (2011). Inter-organizational trust and the dynamics of distrust. *Journal of International Business Studies, 42,* 35–47.
Madon, S. (1992). Computer-based information systems for development planning: The significance of cultural factors. *Journal of Strategic Information Systems, 1*(5), 250–257.
Malone, T. W., & Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Computing Surveys, 26*(1), 87–119.
Martinsons, M. G., & Hempel, P. S. (1998). Chinese business process re-engineering. *International Journal of Information Management, 18*(6), 393–407.
McEvily, B., & Marcus, A. (2005). Embedded ties and the acquisition of competitive capabilities. *Strategic Management Journal, 26*(11), 1033–1055.
McSweeney, B. (2009). Dynamic diversity: Variety and variation within countries. *Organization Studies, 30*(9), 933–957.
Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis*. Thousand Oaks, CA: Sage.
Morck, R., Yeung, B., & Zhao, M. (2008). Perspectives on China's outward foreign direct investment. *Journal of International Business Studies, 39,* 337–350.

Morosini, P., Shane, P., & Singh, H. (1998). National cultural distance and cross-border acquisition performance. *Journal of International Business Studies, 29*(1), 137–158.

Morris, M. W., & Fu, H. (2001). How does culture influence conflict resolution: A dynamic constructivist analysis. *Social Cognition, 19*(3), 324–349.

Myers, M. D. (2004). Hermeneutics in information systems research. In J. Mingers & L. P. Willcocks (Eds.), *Social Theory and Philosophy for Information Systems* (pp. 103–128) Chichester, NY: Wiley.

Myers, M. D., & Tan, F. B. (2002). Beyond models of national culture in information systems research. *Journal of Global Information Management, 10*(1), 24–32.

Naughton, B. (1996). *Growing out of the plan: Chinese economic reform, 1978–1993*. Cambridge, UK: Cambridge University Press.

Newsweek. (2008). Rise of the sea turtles. *Newsweek*. http://www.newsweek.com/id/151730/page/1.

Nicholson, B., & Sahay, S. (2001). Some political and cultural implications of the globalisation of software development: Case experience from UK and India. *Information and Organisation, 11*(1), 25–43.

Olsson, H. H., Conchúir, E. Ó., Ågerfalk, P. J., & Fitzgerald, B. (2008). Two-stage offshoring: An investigation of the Irish bridge. *MIS Quarterly, 32*(2), 257–279.

Oshri, I., Kotlarsky, J., & Willcocks, L. (2007). Managing dispersed expertise in IT offshore outsourcing: Lessons from Tata Consulting Services. *MISQ Executive, 6*(2), 53–65.

Osland, J. S., & Bird, A. (2000). Beyond sophisticated stereotyping: Cross-cultural sensemaking in context. *Academy of Management Executive, 14*(1), 1–12.

Osland, J. S., Bird, A., & Gundersen, A. (2007). *Trigger events in intercultural sensemaking*. Philadelphia, PA: Academy of Management Meeting.

Oyersman, D., Coon, H. M., & Kemmelmeier, M. (2002). Rethinking individualism and collectivism: Evaluation of theoretical assumptions and meta-analyses. *Psychological Bulletin, 128,* 3–72.

Pattillo-McCoy, M. (1998). Church culture as a strategy of action in the black community. *American Sociological Review, 63*(6), 767–784.

Patton, M. Q. (2002). *Qualitative research and evaluation methods*. Thousand Oaks, CA: Sage Publications.

Peng, M. W. (2003). Institutional transitions and strategic choices. *Academy of Management Review, 28*(2), 275–296.

Perretti, F., Negro, G., & Lomi, A. (2008). E Pluribus Unum: Framing, Matching, and Organizational Form Emergence in U.S. Television Broadcasting, 1940-1960. *Organization Science, 19*(4), 533–547.

Rai, A., Maruping, L. M., & Venkatesh, V. (2009). Offshore information systems project success: The role of social embeddedness and cultural characteristics. *MIS Quarterly, 33*(3), 617–641.

Rao, H. (1994). The social construction of reputation: Certification contests, legitimation, and the survival of organizations in the American automobile industry, 1895–1912. *Strategic Management Journal*, *15*(Winter Special Issue), 29–44.

Rao, H. (1998). Caveat emptor: The construction of nonprofit consumer watchdog organizations. *American Journal of Sociology, 103*(4), 912–961.

Rao, H., Monin, P., & Durand, R. (2003). Institutional change in Toque Ville: Nouvelle Cuisine as an identity movement in French gastronomy. *American Journal of Sociology, 108*(4), 795–843.

Redding, S. G. (1994). Comparative management theory: Jungle, zoo or fossil bed? *Organization Studies, 15*(3), 323–359.

Rindova, V., Dalpiaz, E., & Ravasi, D. (2011). A cultural quest: A study of organizational use of new cultural resources in strategy formation. *Organization Science, 22*(2), 413–431.

Ring, P. S., & van de Ven, A. H. (1994). Developmental processes of cooperative interorganizational relationships. *Academy of Management Review, 19*(1), 90–118.

Rottman, J., & Hao, L. (2008). Can China compete with India in the global ITO/BPO market? In M. Lacity & J. Rottman (Eds.), *Offshore outsourcing of IT work* (pp. 180–208). London, U.K.: Palgrave.

Sahay, S., Nicholson, B., & Krishna, S. (2003). *Global IT outsourcing: Software development across borders*. Cambridge, UK: Cambridge University Press.

Salk, J. E., & Brannen, M. Y. (2000). National culture, networks, and individual influence in a multinational management team. *Academy of Management Journal, 43*(2), 191–202.

Sarker, S., & Sarker, S. (2009). Exploring agility in distributed information systems development teams: An interpretive study in an offshoring context. *Information Systems Research, 20*(3), 440–461.

Schoorman, F. D., Mayer, R. C., & Davis, J. H. (2007). An integrative model of organizational trust: Past, present, and future. *Academy of Management Review, 32*(2), 344–354.

Schreiner, M., Kale, P., & Corsten, D. (2009). What really is alliance management capability and how does it impact alliance outcomes and success? *Strategic Management Journal, 30*(13), 1395–1419.

Sia, C. L., Lim, K. H., Leung, K., Lee, M. K. O., Huang, W. W., & Benbasat, I. (2009). Web strategies to promote internet shopping: Is cultural-customization needed? *MIS Quarterly, 33*(3), 491–512.

Simonin, B. L. (1999). Ambiguity and the process of knowledge transfer in strategic alliances. *Strategic Management Journal, 20*(7), 595–623.

Snow, D. A., Rochford, E. B., Worden, S. K., Jr., & Benford, R. D. (1986). Frame alignment processes, micromobilization, and movement participation. *American Sociological Review, 51*(4), 464–481.

Soh, C., Kien, S. S., & Tay-Yap, J. (2000). Enterprise resource planning: Cultural fits and misfits: is ERP a universal solution? *Communications of the ACM, 43*(4), 47–51.

Srite, M., & Karahanna, E. (2006). The influence of national culture on the acceptance of information technologies: An empirical study. *MIS Quarterly, 30*(3), 679–704.

Starbuck, W. H., & Milliken, F. J. (1988). Executives' personal filters: What they notice and how they make sense. In D. C. Hambrick (Ed.), *The executive effect: Concepts and methods for studying top managers* (p. 51). Greenwich, CT: JAI Press.

Steensma, H. K., Tihanyi, L., Lyles, M. A., & Dhanaraj, C. (2005). The evolving value of foreign partnerships in transitioning economies. *Academy of Management Journal, 48*(2), 213–235.

Straub, D., Loch, K., Evaristo, R., Karahanna, E., & Srite, M. (2002). Toward a theory-based measurement of culture. *Journal of Global Information Management, 10*(1), 13–23.

Strauss, A. L., & Corbin, J. M. (1997). *Grounded theory in practice*. Thousand Oaks, CA: Sage Publications.

Stolte, J. F., & Fender, S. (2007). Framing social values: An experimental study of culture and cognition. *Social Psychology Quarterly, 70*(1), 59–69.

Strang, D., & Meyer, J. W. (1993). Institutional conditions for diffusion. *Theory and Society, 22,* 487–511.

Swidler, A. (1986). Culture in action: Symbols and strategies. *American Sociological Review, 51*(2), 273–286.

Taras, V., Kirkman, B. L., & Steel, P. (2010). Examining the impact of culture's consequences: A three-decade, multilevel, meta-analytic review of Hofstede's cultural value dimensions. *Journal of Applied Psychology, 95*(3), 405–439.

Tiwana, A., & Bush, A. A. (2007). A comparison of transaction cost, agency, and knowledge-based predictors of IT outsourcing decisions: A U.S.-Japan cross-cultural field study. *Journal of Management Information Systems, 24*(1), 259–300.

Tiwana, A., Bush, A. A., Tsuji, H., Yoshida, K., & Sakurai, A. (2008). Myths and paradoxes in Japanese IT offshoring. *Communications of the ACM, 51*(10), 141–145.

Triandis, H. C. (1994). *Culture and social behavior*. New York, NY: McGraw-Hills.

Triandis, H. C., Vassiliou, V., Vassiliou, G., Tanaka, Y., & Shanmugam, A. V. (1972). *The analysis of subjective culture*. New York, NY: Wiley.

Tsoukas, H., & Vladimirou, E. (2001). What is organizational knowledge? *Journal of Management Studies, 38*(6), 973–993.

Tung, R. L. (2008). The cross-cultural research imperative: The need to balance cross-national and intra-national diversity. *Journal of International Business Studies, 39,* 41–46.

Tung, R. L., & Verbeke, A. (2010). Beyond Hofstede and GLOBE: Improving the quality of cross-cultural research. *Journal of International Business Studies, 41,* 1259–1274.

Vaast, E., & Levina, N. (2006). Multiple faces of codification: Organizational redesign in an IT organization. *Organization Science, 17*(2), 190–201.

Vlaar, P. W. L., Van Fenema, P. C., & Tiwari, V. (2008). Cocreating understanding and value in distributed work: How members of onsite and offshore vendor teams give, make, demand, and break sense. *MIS Quarterly, 32*(2), 227–255.

Walsham, G. (2002). Cross-cultural software production and use: A structurational analysis. *MIS Quarterly, 26*(4), 359–380.

Wan, C., Chiu, C. Y., Tam, K. P., Lee, S. L., Lau, I. Y.-M., & Peng, S. Q. (2007). Perceived cultural importance and actual self-importance of values in cultural identification. *Journal of Personality and Social Psychology, 92,* 337–354.

Wedeen, L. (2002). Conceptualizing culture: Possibilities for political science. *The American Political Science Review, 96*(4), 713–728.

Weick, K. E. (1999). Sensemaking as an organizational dimension of global change. In D. L. Cooperrider & J. E. Dutton (Eds.), *Organizational dimensions of global change* (pp. 39–56). Thousand Oaks, CA: Sage.

Weick, K. E., Sutcliffe, K. M., & Obstfeld, D. (2005). Organizing and the process of sensemaking. *Organization Science, 16*(4), 409–421.

Wiener, M., Vogel, B., & Amberg, M. (2010). Information systems offshoring—A literature review and analysis. *Communications of the Association for Information Systems*, *27*(1), 455–492.Article 25.

Yang, M. M. (1994). *Gifts, favors and banquets: The art of social relationships in China*. Ithaca, NY: Cornell University Press.

Yin, R. K. (2003). *Case study research: Design and methods* (3rd ed.). Beverly Hills, CA: Sage Publications.

Zaheer, S. (1995). Overcoming the liability of foreignness. *Academy of Management Journal, 38*(2), 341–363.

Zaheer, A., & Kamal, D. F. (2011). Creating trust in piranha-infested waters: The confluence of buyer, supplier and host country contexts. *Journal of International Business Studies, 42,* 48–55.

Zaheer, A., McEvily, B., & Perrone, V. (1998). Does trust matter? Exploring the effects of interorganizational and interpersonal trust on performance. *Organization Science, 9*(2), 141–159.

Zhang, M., Sarker, S., & Sarker, S. (2008). Unpacking the effect of IT capability on the performance of export-focused SMEs: A report from China. *Information Systems Journal, 18*(4), 357–380.

Zou, X., Tam, K., Morris, M. W., Lee, S., Lau, I. Y., & Chiu, C. (2009). Culture as common sense: Perceived consensus versus personal beliefs as mechanisms of cultural influence. *Journal of Personality and Social Psychology, 97*(4), 579–597.

Competing Institutional Logics in Impact Sourcing

Fareesa Malik and Brian Nicholson

Abstract This chapter examines competing welfare and market logics in impact sourcing. Impact sourcing is an emerging trend in the global outsourcing industry that aims to contribute to the welfare of marginalised people by providing employment opportunities in outsourcing centres. Drawing on the concepts of institutional logics this paper presents a case study of a USA based IT outsourcing vendor "Alpha-Corp" practising impact sourcing in a Pakistan subsidiary. The findings show that in cases where actors are located in diverse institutional contexts, competing interests determine the respective priority given to the welfare and market logics. Multiple responses to the competing logics are identified and we offer a conceptualisation of "enclaves" of competing institutional logics in impact sourcing.

1 Introduction

This chapter explores competing institutional logics of impact sourcing, those of welfare and the market drawing on the case of a commercial for profit IT outsourcing organisation. The corporate social responsibility (CSR) discourse of creating social and economic value (Emerson 1999; Porter and Mark Kramer 2011) has gained wide acceptance—both in academic research and in the practice of global IT outsourcing (Babin 2011; Hefley and Babin 2013; Madon and Sharanappa 2013). Impact sourcing is an emerging phenomenon in the global IT outsourcing industry aligned with this CSR discourse. It focusses on providing work in outsourcing centres for marginalised people living in regions of lowest employment opportunity in order to contribute towards improving their livelihoods. Examples include centres aimed at women in rural India (Heeks and Arun 2010; Lacity et al. 2011) but also prisoners (Lacity et al. 2015), military veterans and aboriginals (Babin et al. 2016; Khan et al. 2018). The

F. Malik (✉)
NUST Business School, National University of Sciences and Technology, Islamabad, Pakistan
e-mail: fareesa.malik@nbs.nust.edu.pk

B. Nicholson
Alliance Manchester Business School, University of Manchester, Manchester, UK
e-mail: brian.nicholson@manchester.ac.uk

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_9

phenomenon of impact sourcing is still relatively nascent; prior studies have focused on highlighting the positive impacts on marginalised outsourcing employees (Lacity et al. 2014; Madon and Sharanappa 2013; Malik et al. 2016). Few studies have attempted to examine how IT-enabled outsourcing organisations and clients manage the market orientation of impact sourcing (Lacity et al. 2012). Other IS scholars have identified the paradoxical social and market orientations of impact sourcing (Sandeep and Ravishankar 2015a). This prior research is limited to case studies of a relatively small group of specialist impact sourcing vendors (e.g. CloudFactory, DDD) that are exclusively social enterprises or public sector governed schemes. There is a paucity of research into the competing welfare and market logics of impact sourcing in the much larger market driven commercial IT outsourcing sector. In response to this gap in the literature, our research question is as follows:

> RQ: How does a commercial IT outsourcing organisation adapt to competing welfare and market logics when engaging in impact sourcing?

We draw on prior literature of institutional logics and hybrid organisation (Battilana and Dorado 2010; Pache and Santos 2013; Smets et al. 2015) to explore the implications of competing welfare and market logics of impact sourcing in the case of a NASDAQ listed market driven IT outsourcing organisation. Hybrid organisations 'incorporate elements from different institutional logics' (Pache and Santos 2013, p. 972) and this theoretical lens allows us to make sense of the case of an IT-outsourcing organisation, "AlphaCorp", located in the USA and Pakistan. Impact sourcing was practiced in the Pakistan based subsidiary located in the small, earthquake affected district of Bagh. This centre had a welfare mission of providing employment opportunities to the marginalised educated youth of that region particularly women.

2 Literature Review and Conceptual Frame

2.1 Institutional Logics and "Hybrids"

The institutional logics (IL) perspective emerged as a development of neo-institutional theory (DiMaggio et al. 2000; Meyer and Rowan 1977). Rooted in the seminal work of Friedland and Alford (1991), institutional logics recognises the arguments of neo-institutional theory that institutions and macro structures contribute to shaping organisational and individual behaviours. Institutional logics are the organising principles that define the content and meaning of institutions, the socially constructed sets of materials, practices, assumptions, values, and beliefs that shape cognition and behaviour (Thornton et al. 2012). The significant theoretical advancement is in the emphasis on multiple contradictory and interdependent institutional logics in the larger context and their effects on individuals and organisations (Thornton and Ocasio 2008). Empirical studies of multiple institutional logics have been

Table 1 Organisational response to competing logics

Response strategy	Explanation
Decoupling	Symbolically endorse one logic and practically implement another competing logic (Bromley and Powell 2012; Crilly et al. 2012; Boxenbaum and Jonsson 2008)
Combining	Combine the demands of all competing logics (Lounsbury 2007; Battilana and Dorado 2010)
Compromise	Maintain a balance to meet the minimal demand of all competing logics (Kraatz and Block 2008)
Selective Coupling	Couple different organisational elements responding to each competing logic individually (Pache and Santos 2013)

undertaken widely in different fields—for example, social enterprise (Pache and Santos 2013; Seelos et al. 2011); public-private partnerships (Jay 2013); healthcare organisations (Dunn and Jones 2010; Reay et al. 2009); microfinance and banking (Lounsbury 2007); biotechnology (Powell and Sandholtz 2012); and professional services (Smets et al. 2012). Research into hybrid organisations has focussed on making sense of the complex pluralistic institutional environment due to exposure to multiple competing institutional logics (Greenwood et al. 2011). Furthermore, several scholars have conceptualised the responses to multiple institutional logics; the four main types are summarised in Table 1.

"Decoupling" involves separation of normative structures from their operational processes by symbolically endorsing one logic while practically implementing another competing logic (Boxenbaum and Jonsson 2008; Bromley and Powell 2012). In practice, this is where an organisation decouples the symbolic endorsement of one logic from the operational practices of the other logic. An example is shown in the practice of "greenwashing" that occurs when an organisation presents an image of environmentally responsibility to external stakeholders while not actually implementing appropriate practices.

"Combining" involves the amalgamation of elements from two or more competing logics illustrated in Battilana and Dorado (2010) analysis of the combination of banking and social welfare logics adopted by microfinance organisations. Combination involves balancing social welfare and banking logics and remain sustainable. Pache and Santos (2013) describe a variant form of combination as "selective coupling" which can be understood as selecting different organisational elements responding to different competing logics and couple them together.

"Compromising" involves the modification of prescriptions of the competing logics into an intermediate position, for example a micro-finance organisation may compromise in setting a lending interest rate lower than the market rate but higher than the very poorest may afford. Thus, compromise enables organisations to maintain a balance between conflicting demands exerted by competing institutional logics to meet the minimal requirements of internal and external actors (Kraatz and Block 2008).

2.2 Institutional Logics in Impact Sourcing

Cost reduction, access to skills, focus on core capabilities and improved business processes are widely recognised as reasons for outsourcing IT-enabled business process functions (Lacity et al. 2010, 2011). However, the provision of low cost and high quality is challenged by rising operational cost in metropolitan cities (e.g. Bangalore in India) caused primarily by wage inflation (Lacity et al. 2010; Madon and Sharanappa 2013). To overcome this problem there has been a trend towards opening centres and shifting work into locations outside the metropolitan cities (Lacity et al. 2010, 2011). Alongside this market rationale for relocation, another focus of this recent trend has been to create social and economic value by providing work opportunities to the marginalised unemployed population (Heeks and Arun 2010; Lacity et al. 2014). Marginalisation is not only restricted to poverty, it also includes marginalisation due to location, gender, religion, disability, ethnicity and education (Carmel 2014).

The practitioner literature (e.g. Rockefeller Foundation) posits that impact sourcing can contribute to the social welfare of marginalised outsourcing employees as well as maximising outsourcing organisation's business profitability. This "win–win" outcome is attributed to reduced operational costs and lower employee turnover rate in the marginalised communities (Accenture 2012; Monitor 2011). These reports, supporting the discourse of dual value creation of impact sourcing, provide measures of social impact assessments and report the positive social outcomes with limited empirical evidence (Accenture 2013; William Davidson Institute 2013). The early academic studies have also focussed on the individual level of analysis to assess the potential for creation of social value through impact sourcing. For example, Heeks and Arun (2010), Sandeep and Ravishankar (2015b) and Madon and Sharanappa (2013) all provide evidence of social development in poor rural communities in India associated with impact sourcing. However, these studies are limited by their basis on cases of social enterprise, development or public sector supported initiatives where welfare logics are typically dominant over market logics (Ismail et al. 2018; Heeks and Arun 2010; Malik et al. 2017).

Khan et al. (2018) present a conceptual framework to demonstrate dual orientation of impact sourcing in terms of social and commercial logics. The conceptual framework identifies four dimensions of impact sourcing: primary mission, success criteria, resource mobilization, and innovation. Two of these dimensions, primary mission and resource mobilizations, show social logics; success criteria anchor around commercial logics; and innovation incorporates both social and commercial logics. Few recent research studies, however, identify the possible tensions arise from various stakeholders (particularly local communities) while managing both welfare and market logics of impact sourcing (Sandeep and Ravishankar 2015a, Malik and Nicholson 2019). The literature also shows evidence of conflicts where commercial interests may suppress the social logics in a public-private partnership initiative of impact sourcing (Ismail et al. 2018). These studies significantly improve our understanding of competing nature of market and welfare logics of impact sourcing.

However, more empirical research is required to examine how competing welfare and market logics of impact sourcing can be managed in commercial IT outsourcing arrangements.

3 Research Methods

The research design is centred on an interpretive qualitative case study (Eisenhardt and Graebner 2007; Walsham 1995, 2006) of "AlphaCorp" (a pseudonym). AlphaCorp was chosen because it is a public listed IT outsourcing organisation and thus based on a commercial, for-profit business model. Crucially, AlphaCorp were practicing impact sourcing in one of their Pakistan subsidiaries and thus both welfare and market logics were clearly in evidence. Also, the selected case had operations at multiple locations (USA, Islamabad and Bagh, Pakistan), allowing us to study the effects of diverse operations and contexts on competing institutional logics.

We collected rich qualitative data based on a variety of data sources (Garcia et al. 1997; Stake 1995). One of the authors engaged in fieldwork in Pakistan from September 2013 to March 2014 and also in the USA head office during August 2014. Primary data were collected from multiple data sources; semi-structured open ended interviews as well as field notes based on observations and informal communication with employees over breaks and mealtimes during the 7 months of fieldwork at AlphaCorp's Islamabad and Bagh centres. We conducted 78 interviews with a range of employees from all hierarchal levels (billing executives, middle management, higher executives) in the three centres (Islamabad, Bagh, and USA) of AlphaCorp. The semi-structured interviews with middle and higher level management were first related to historical reconstruction of AlphaCorp's inception in Islamabad and Bagh and then the management challenges that were encountered over time. We started initial interviews with some pre-prepared questions using the institutional logics as a sensitising framework. This interviewing process gradually led to spontaneous questions as informants guided our inquiry to explore more about the case study (Charmaz 2014). To confirm the existence of the welfare logic, the outsourcing employees working in the Bagh centre were asked questions that informed us of the institutional context of Bagh and if/how working in an offshore-outsourcing centre had social welfare impacts as well as barriers to those impacts. Interviewees were asked questions about the establishment of the centres, motivation behind establishing the Bagh centre, operational challenges and business advantages, how challenges are overcome (if at all); relevant formal or informal rules, processes and governance structures. With the exception of two interviews, 76 interviews were audio recorded with the permission of the interviewees and later transcribed verbatim by the authors. Handwritten notes were taken in the remaining interviews which were typed up as soon as possible afterwards. We also collected secondary data related to the case study that was openly accessible on internet: websites, press releases, and AlphaCorp's registration form downloaded from the NASDAQ stock exchange website.

4 Data Analysis

We used template analysis to thematically analyse the qualitative data (King 2004). An initial template was built based on the institutional logics—responses literature review, researcher experience and analysis of a number of interviews (ibid.). We analysed the case study by going back and forth in the empirical data, mapping it with the literature (Miles et al. 1994), and revising the initial template until all the data had been coded. NVivo 10 software was used to assist data analysis as computer aided software supports the organisation and management of large amounts of qualitative data (Bazeley and Jackson 2013). The data analysis was completed in three steps.

First step—create initial template: To create the initial template, we adopted top-down coding where the initial coding scheme was generated from the literature on institutional logics (Urquhart 2013). Along with the literature review, personal field research experience and a sub-set of interview transcripts were used to generate primary codes for the initial template (King 2004).

Second step—coding the qualitative data: In the second step, we imported all transcripts in NVivo 10, and went through the textual data line by line to organise the relevant data according to different codes. The initial template was modified, as new codes were added, previous codes revised and a few deleted. We updated the coding process to ensure that all relevant data had been sorted to respective codes and the final coding template had been generated.

Third step—identifying empirical themes and linking them to the literature: In the final step, we went through the filtered data organised against all codes and added our own interpretation and reflection in memos that we created for each code in NVivo 10. We merged related codes to drive the empirical themes of this case study; for example, two codes, explicit business objectives and implicit corporate social objectives, were merged to become a single empirical theme: 'explicit commercial and implicit social responsibility goals'. To address the research question, we reread the sorted data in the final empirical themes and all associated memos containing data interpretation and reflection. Referring to the institutional logic literature, we tried to make sense of the empirical findings by mapping them with the theoretical categories (Eisenhardt 1989) of decoupling, combining, compromising, and selective coupling.

5 The Case Organisation

This case study is based on a medical billing process outsourcing organisation, with USA headquarter and listed on the NASDAQ stock exchange in 2014. AlphaCorp has two offshore processing centres in Pakistan: one in Islamabad and other in the small district of Bagh employing 200 staff. The company provides IT and business process services to more than 900 medical practitioner clients based across the USA. These services include medical practice management, centralised management information systems, mobile healthcare solutions, medical billing services and electronic healthcare record management services. All of these IT and business process services are provided from AlphaCorp's two offshore subsidiaries in Pakistan. In 2009, AlphaCorp established a centre in Bagh to provide a backup office. The establishment of the centre was also motivated by a welfare related objective of socio-economic development of the region by creating employment opportunities for the educated unemployed youth of the area, especially women. In addition, the residents of Bagh were affected by a deadly earthquake in October 2005, which destroyed homes and infrastructure and led to 10,000 causalities. AlphaCorp is the only private sector IT and business process outsourcing organisation in this region and previously, the majority of educated people raised in Bagh joined the public sector or moved to other cities of Pakistan to find white collar jobs. However, due to the conservative social norms of that area, most parents did not allow their daughters to move to other cities (Ferdoos 2005). The physical context is of relevance as Bagh is located in the Pakistan Himalaya region. In winter the local weather conditions become extreme as landslides are common in the mountainous terrain, and this influences daily routines.

6 Empirical Findings

This section presents the findings related to how AlphaCorp adapted to competing welfare and market logics in relation to the organizational responses to competing logics shown in Table 1.

6.1 Decoupling: Explicit Market and Welfare Logics

The market logic was strongly reflected in the organisational goals of AlphaCorp expressed as the intention of becoming a leading provider of IT solutions and business process services to the US healthcare industry. This was explicitly stated in all publicly available secondary data, such as the company website and company reports available on the NASDAQ website:

> Our objective is to become the leading provider of end-to-end software and business service solutions to healthcare providers practicing in an ambulatory setting. (NASDAQ Doc. p. 5)

In parallel, we found evidence of competing logics in all three centres of AlphaCorp. Specifically we identified an implicit welfare objective that was enacted in the organisational practices of the Bagh centre but was not formally documented. During interviews, middle and higher level management in the Islamabad and Bagh centres repeatedly talked about various social initiatives that the company had been practising in particular to support activities to encourage and facilitate female employment. Although these initiatives were not formally recognised in any documents nor were they included as official corporate social responsibility activity they were informally recognised as such by the Pakistan management team. The financial planning manager from Islamabad centre expressed this in terms of profit and social responsibility logics:

> Some times it is not only financial gain you are looking for, instead other things like corporate social responsibilities which you have to fulfil. (FP Manager—Islamabad)

Interviews also pointed to various levels at which organisational actors in AlphaCorp were informed about the implicit social responsibility organisational goal. The employees, and management in the Bagh and Islamabad centres were well aware of the welfare logic in the Bagh centre:

> Obviously this is a business, one would be absolutely wrong to say that we are not making profit… and I think our business is doing really very well…. but what we have invested in Bagh is beyond the consideration of monetary profit. (Company Spokesperson—Islamabad)

However, there was an absence of any written documentation related to social responsibility goals of the Bagh centre nor is it mentioned on the AlphaCorrp or NASDAQ website. A corollary of this is that shareholders and clients in USA were unaware of the welfare logic in Bagh. To probe this chain of evidence, we asked management in Pakistan about the USA clients' reaction to the welfare logic. The pattern of response we got from many interviewees was similar—all stated that clients were only concerned with obtaining quality services at the lowest possible cost.

All this provides evidence of a decoupling response to competing market and welfare logics which is linked to the perception of clients and shareholders lack of interest in welfare related social responsibility activities in the Bagh centre. Interviews with senior management in the USA head office, who were also shareholders and members of the board of directors, and had associated with AlphaCorp for more than ten years, supported our observation of the decoupling of welfare and market logics for organisational actors in the USA. Although a majority of the higher management in the USA mentioned social responsibility concerns in general, they were not able to articulate the organisational practices of AlphaCorp in Pakistan, particularly the welfare related initiative in Bagh.

In summary, the findings show that AlphaCorp adopted a decoupling strategy to keep both market and welfare logics separate in defining the company's goals. The market logic was explicitly expressed in the company's documents, reflecting the commercial orientation of the company. Thus, all actors we met were aware of the market logic of AlphaCorp. The welfare logic, however, was implicitly recognised and expressed in organisational practice in the "enclave" of Pakistan.

6.2 Combining: Location Decision to Achieve Market and Welfare Logics

AlphaCorp started its offshore-outsourcing operations in Pakistan for three reasons: First, the company commenced operations in Islamabad in collaboration with a female only non-profit organisation that offered vocational training and other skills development courses to Pakistani women. In 2002, they had started a small scale project (employing 25–30 females) that enabled them to learn about computers and earn money through providing IT-enabled outsourcing services to the US clients. Second, the founder of the business was a Pakistani and it was more convenient for him to establish a business in Pakistan compared to any neighbouring country in South Asia, which is a region that is considered as an ideal outsourcing location. Third, all required resources, most predominantly ICT infrastructure and a large pool of skilled human resources, were available at relatively low cost. The small project grew tremendously and turned into one of the largest business processing and IT outsourcing organisation in Pakistan.

In 2007, the assassination of the former Prime Minister of Pakistan caused civil unrest. Although AlphaCorp managed to maintain important operations and client services with a limited number of employees who lived near the office premises or stayed in office building accommodation areas, the complex situation caused Pakistan management to consider the need for a backup office in Pakistan but at a remote location. Bagh was the obvious choice as it was already being considered for welfare reasons as previously stated above. The vice president of AlphaCorp explained the market logic behind this location decision as follows:

> This is the city versus the rural model .... So I think from a positioning stand point, investors and also clients feel some comfort that there are two outsourcing centres operational in Pakistan to provide them continuous services. (VP—Islamabad)

Availability of skilled human resources is a major factor to be considered in the location decision of any IT outsourcing organisation (Lacity et al. 2011). Many educational institutions had been established in Bagh as a result of rehabilitation and development work carried by public sector, national, and international non-governmental organisations (NGOs) after 2005's deadly earthquake. These educational institutions had produced a large number of graduates who remained unemployed because of limited employment opportunities available in that area. The presence of a large pool of educated human resources supported the market logic of AlphaCorp's location decision. The Manager of Compliance reported:

> At that time, we realized that we needed a backup office away from centralizing everything in Islamabad .... Every curriculum has computer subjects throughout the country so young graduates of Bagh are also technology aware. Our intention was to create ICT based employment opportunity in this small district, especially for young educated females, the majority of whom won't get permission from their family to move to other cities for work. (Manager Compliance—Islamabad)

Along with the market impetus to select a small town as a backup office for AlphaCorp, there were other significantly strong welfare and identifiable emotional motivations behind this location decision. The earthquake had myriad negative effects on the socio-economic position of the region and its community including homelessness in the relatively short term but also a large group of displaced and orphaned children. The founder of AlphaCorp was born and raised in Bagh and he expressed his desire to 'share his success with people of his home town'. The decision to establish a backup office there was derived by this individual's aim to provide employment opportunities to the marginalised educated population of Bagh and contribute to the socio-economic development of the community.

We thus observe a combination response behind the location decision, supporting both market and welfare logics. Combination is reflected in a quotation from the General Manager on AlphaCorp's location decision:

> First of all, we needed to have a contingency office, just in case any thing happens to the main operational office in Islamabad, there should have been another office where operations could continue without any interruption. The founder of the company belongs to Bagh. He wanted to do something for people of his home town…. Majority of the youth here is educated; they move to cities to complete their education and find the work…. We knew that if we open an office in Bagh we will be able to get the human resource easily. (GM – Islamabad)

This shows that both welfare and market logics were combined for the location decision and were argued to be compatible.

6.3 Compromising: Hiring of Marginalised Employees

The findings above indicate that AlphaCorp's location decision to operate in Bagh was also motivated by a social welfare impulse to provide employment opportunities to the marginalised, earthquake affected local youth, especially females. As discussed previously, educational institutions in Bagh had been producing a large number of young graduates. The respondents told us that most of these institutions, especially higher education institutions, were recently opened and still in the process of raising educational standards as compared to the well-established older educational institutions in capital or other metropolitan areas of Pakistan. Whilst their graduates met the minimum educational requirement standards of AlphaCorp, they were lacking in communication and other interpersonal (soft) skills:

> We don't hire people that are not educated. We hire educated people but the level of competency is different in Bagh and that competency difference is mostly in term of soft skills. (Chief Operating Officer—Islamabad)

The institutional context played a significant role in shaping the capabilities of marginalised outsourcing employees. According to the HR manager, who was an educator for more than ten years before joining AlphaCorp, the primary level educational systems and structures were responsible for the weakness in communication and other soft skills. Interpretation of data gathered from middle level management in the Islamabad centre indicated another factor—lack of market exposure—as in part responsible for the less well developed local graduates. The marketing manager in Islamabad pointed out that the many work opportunities available in large metropolitan cities meant that graduates could obtain market and professional experience before formally starting their professional careers via internships, practical projects, etc. Such exposure is not available in the small town environment of Bagh.

Outsourcing service providers market themselves as having highly compatible human resources that can provide high quality services to the outsourcing clients (Lacity et al. 2011). However, the market logic of the global IT outsourcing industry is in competition with welfare logics of hiring the marginalised outsourcing employees. The Bagh centre HR manager commented on the "limited professional skills" of the available human resource and the company's efforts to "polish" them according to international standards:

> We don't get human resource according to the company standards in Bagh. But we hire the potential candidates and put extra effort to bring them to our company standards. (Manager HR—Bagh)

AlphaCorp has achieved a series of international certifications ratifying its process and service quality standards in the global IT outsourcing industry, for example, ISO 27001 (management of information security certification) and ISO 9001:2000 (process and quality management certifications) from the International Organization for Standardization (ISO). To achieve successful quality results and meet the certified quality standards, the compromise response to the market logic is manifested in the cost of additional training and monitoring to maintain the quality standards exerted by the more macro level market logic of the global IT outsourcing industry. The senior manager operations (SMO) mentioned the extra training need of the Bagh employees to reach the high quality standards and need for extra monitoring of the outsourcing service delivered to the clients:

> We have to be more vigilant there. Our primary focus is to train them and bring them to the same quality level. To maintain the service quality, in the beginning for a month or two months, we put in more checks on the work produced by newly hired employees in Bagh. But after a little they achieve the certain quality level. (Senior Manager Operations—Islamabad)

6.4 Selective Coupling: Practices to Support Market as Well as Welfare

All IT outsourcing organisations need to provide uninterrupted quality services to clients in support of contracts in a highly competitive global IT outsourcing industry (Lacity et al. 2011). Due to time zone difference between the locations where outsourcing services were provided and received, AlphaCorp operates three consecutive shifts of eight hours. The morning shift is reserved for female employees to facilitate and encourage female employment:

> We run three operation shifts round the clock and our morning shift is reserved for female employees so that they can independently and comfortably work. (Manager HR—Islamabad)

The morning shift is an example of selective coupling response as will be explained in sections to follow. Historically, few women work in the private sector in Bagh. The social and cultural norms of conventional society (Ferdoos 2005; Mughal and

Zeb. 2014) had institutionalised job and role segregation for men and women. The majority of educated women enter teaching or the public sector departments that are considered as female friendly professions (Ferdoos 2005; Rehman and Roomi 2012). These local norms encountered competition after the 2005 earthquake, when national and international NGOs arrived for reconstruction and many educated local women joined them. Some NGOs did not consider the local cultural and normative values and continued their western management strategies in the region. However the local community had negative perceptions about these NGOs because of their work practices, which did not align well with their norms and values. For example, the local community found it disrespectful for their daughters to work late in the evening alongside men.

Initially, the local community considered AlphaCorp as an international NGO because of its origin in the USA and their ignorance about the healthcare IT and business process outsourcing industry. Because of the negative perception about international organisations in the community, local people became highly sceptical about AlphaCorp. They could not make sense of the rationale for the establishment of the centre and started sharing conspiracy theories. As a result of the many concerns, they did not allow their children to work in AlphaCorp. One female billing executive described the situation in this way:

> After the earthquake, many NGOs were operational in our town. They had hired female staff. Afterward the cultural issues had started coming up. It was not acceptable in our culture for girls to be with boys till late in the evening. When I intended to work in AlphaCorp my father said, 'I would not allow you to go door to door in villages all the time; it is not respectable for girls.' (Billing Executive 26—Bagh)

To overcome these challenges, AlphaCorp adopted selective coupling responses including the dedicated morning shift for female outsourcing employees. This helped to build a relationship of trust between the local community and AlphaCorp. They started accepting AlphaCorp in the community and acknowledged its contribution to local employment generation. The dedicated morning shift to encourage women's participation in economic activities for their empowerment directly reflects a selective coupling to the welfare logic, more specifically the logic of gender norms in Bagh.

A female billing executive endorsed it thus:

> The most important thing for us is the secure and relaxed environment. We are all girls working here. My family is satisfied that they have treated us protectively; they send us here by trusting company management." (Billing Executive 12—Bagh)

However, while this strategy satisfied the logics of gender in one respect, we observed negative consequences for the market logic. After the morning shift was institutionalised, male outsourcing employees' turnover rate in the Bagh centre dramatically increased relative to female employees. Along with some obvious reasons for male employees' turnover, such as moving abroad to earn more money, continuing higher education, or moving to cities for professional growth, male interviewees expressed dissatisfaction with the lack of opportunity to work on the morning shift. Many male outsourcing employees of AlphaCorp were not locally resident; they had to commute from nearby villages in the surrounding Himalayan hills. The Operations

Manager in Bagh told us that outsourcing employees who travelled from adjacent mountain villages could not continue their jobs on evening and night shifts because of the severe weather in winter. Families of many male employees did not accept their work pattern of nights or evening shifts and pressured them to resign:

> People in Bagh don't like to work in night shifts particularly. They don't feel secure; their parents are very anxious because it is not the trend there to work in nights. It is not a problem in Islamabad. Here boys can easily work in the evening and night shifts. (Manager Administration—Islamabad)

These findings show the selective coupling response satisfied the nascent welfare logic and female gender logics. However, the selective coupling was not altogether compatible with the dominant market logic.

7 Discussion and Conclusion

In this paper, we focus on the research question of how does a commercial IT outsourcing organisation adapt to competing welfare and market logics when engaging in impact sourcing. In a detailed case study of a commercial IT outsourcing organisation adapting to competing logics we find multiple responses in action including decoupling, combination, compromise, and selective coupling.

7.1 Contributions to the Study of Impact Sourcing and Institutional Logics

Our findings show the complexity of managing impact sourcing for commercial IT outsourcing organisations. This is shown to be related to multiple organisational actors in different institutional contexts that may have different interests, which ultimately influence the competing institutional logics. Considering first the compromise response, from the client perspective, the IT outsourcing literature has highlighted that clients may have concerns about service quality, since a highly skilled work force is the main selling point of the IT-enabled outsourcing organisations (Lacity et al. 2010b, 2011b). Thus, simply supplanting marginalised outsourcing employees does not present an easy solution to an impact sourcing arrangement as compromise is required to satisfy the competing welfare and market institutional logics. Compromise may be an expensive response because of the additional cost of maintaining balance between competing institutional logics. Our results concur with the existing studies that suggest that if minimum standards for compromise are not set and managed properly, a compromise strategy may put organisational actors' endorsement at risk and raise legitimacy concerns (Carrick-Cagna and Santos 2009). However, the case analysis also determines that compromise is, in some cases, unavoidable for the survival of the nascent welfare logic. For example, compromise was the only option when hiring the marginalised outsourcing employees due to the investment needed in extra training.

The combination response is where organisational mechanisms are influenced by compatible demands of competing logics (Battilana and Dorado 2010; Lounsbury 2007; Reay et al. 2009). In this case, we can see the compatibility of two competing logics in the location decision. Although customised operating procedures were implemented in Bagh along with standard operating procedures to achieve the demand of welfare logic, we did not find any evidence of their formal incorporation as part of documented organisational policies and procedures. This case builds on prior research to present an example of responses to logics that are implicitly held. If in the longer run challenges arise that affect the compatible components of the competing logics; for example, if the cost of operating in rural or remote areas increases, it might exert pressure to amend such ad hoc practices, for example by considering moving the outsourcing centre to another region or country.

Pache and Santos (2013) describe selective coupling as the most viable strategy to respond to competing institutional logics where different elements of the organisation, each responding to just one of the competing institutional logics, couple together to satisfy the collective demands of the competing institutions. The case builds on prior research by showing the dynamics and interplay between logics and unexpected consequences that may emerge. Responding to welfare logic by allocating a separate operational shift for female employees had negative consequences on the market logic. If the option of the morning shift was available to those male employees who needed to commute from a distance, or those whose families had some serious concerns, the male outsourcing employees' turnover in Bagh may have been reduced. However, selective coupling in this case was not be completely viable, contrary to the findings of Pache and Santos (2013). Our findings concur with Besharov and Smith (2014) that competing logics may not be uniform throughout an organisation. Building on this insight, we found competing logics in enclaves of all three centres of the case study organisation. For example, the market logic was very strong in the USA head office mainly due to the closeness to the clients, with only traces of mainly 'generic' welfare logic wrapped up in the social responsibility statements. We found a more balanced influence of both competing logics in the Islamabad enclave; whereas welfare logic was more prominent in the Bagh centre, as compared to market logic. The welfare enclave in Bagh was sustained by the local management and remained detached from any formal inscriptions on websites or reports reported to the USA and clients were unaware of any welfare logic in Pakistan. Taken along with the associated responses, these findings concur with Besharov and Smith (2014) that competing institutional logics may vary across different sub-units of the organisation (Besharov and Smith 2014). We add a more nuanced perspective on how these logics may be created and sustained to retain the legitimacy of other enclaves.

References

Accenture (2012). *Exploring the Value Proposition for Impact Sourcing - The Buyer's Perspective*. https://www.accenture.com/us-en/insight-exploring-value-proposition-impact-sourcing.aspx. Retrieved 14 November 2012.

Accenture (2013). *Recruitment, training, and impact measurement; a recommended approach for impact sourcing [Online]*. Rockefeller Foundation. Available: https://www.rockefellerfoundation.org/report/recruitment-training-and-impact-measurement-a-recommended-approach-for-impact-sourcing

Babin, R. (2011). *Corporate social and environmental responsibility in global IT outsourcing*. Manchester, UK: University of Manchester.

Babin, R., Brian, N., & Megan, Y. (2016). Impact sourcing: aboriginal case study In *Socially Responsible Outsourcing: Global Sourcing with Social Impact. (Technology, Work and Globalization)*, eds. Brian Nicholson, and Ron Babin, and Mary Lacity. Basingstoke, Hampshire, UK: Palgrave Macmillan

Battilana, J., & Dorado, Silvia. (2010). Building sustainable hybrid organizations: The case of commercial microfinance organizations. *Academy of Management Journal, 53*(6), 1419–1440.

Bazeley, P., & Jackson, K. (2013). *Qualitative data analysis with NVivo*. UK: Sage Publications Ltd.

Besharov, M. L. & Smith, W. K. (2014). Multiple institutional logics in organizations: Explaining their varied nature and implications. *Academy of Management Review*, *39*, 364–381.

Boxenbaum, E., & Jonsson, S. (2008). Isomorphism, Diffusion and Decoupling. In *The Sage handbook of organizational institutionalism.*, (Eds.) R. Greenwood, C. Oliver, & R. Suddaby, K. Sahlin-Andersson, (pp. 78–98). UK: Sage Publications Ltd.

Bromley, P., & Powell, W. W. (2012). From smoke and mirrors to walking the talk: Decoupling in the contemporary world. *The Academy of Management Annals, 6*(1), 483–530.

Carmel, E., Mary C. L., & Andrew Doty. 2014. The Impact of Impact Sourcing: Framing a Research Agenda. In *Information Systems Outsourcing. Towards Sustainable Business Value*, eds. Rudy Hirschheim, and Armin Heinzl, and Jens Dibbern, 397–429. vol. The Impact of Impact Sourcing: Framing a Research Agenda. Heidelberg, Berlin Springer

Carrick-Cagna, A.-M., & Santos, F. (2009). *Social vs*. Commercial Enterprise: The Compartamos Debate and the Battle for the Soul of Microfinance.

Charmaz, K. (2014). *Constructing grounded theory* (2nd ed.). UK: Sage Publications Ltd.

Crilly, D., Zollo, M., & Hansen, M. (2012). Faking it or muddling through? Understanding decoupling in response to stakeholder pressures. *Academy of Management Journal, 55*(6), 1429–1448.

DiMaggio, P. J., & Walter, W. P. (2000). The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields. In *Advances in Strategic Managemen*, pp. 143–166. Emerald Group Publishing Limited.

Dunn, M. B., & Jones, C. (2010). Institutional logics and institutional pluralism: The contestation of care and science logics in medical education, 1967–2005. *Administrative Science Quarterly, 55*(1), 114–149.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review, 14*(4), 532–550.

Eisenhardt, K. M., & Graebner, M. E. (2007). Theory building from cases: opportunities and challenges. *Academy of Management Journal, 50*(1), 25–32.

Emerson, J., Jay Wachowicz, & Suzi Chun. 1999. Social return on investment: Exploring aspects of value creation. San Francisco, USA: The Roberts Foundation. (Orig. pub.

Ferdoos, A. (2005). *Social status of rural and urban working women in Pakistan—A comparative study*. Ph.D.: University of Osnabrück, Germany, Mülheim, Germany.

Friedland, Roger, and Robert R Alford. 1991. Bringing society back in: Symbols, practices and institutional contradictions. In *The new institutionalism in organizational analysis*, 232–263. USA: The University of Chicago Press.

Garcia, Lucia, and Francis Quek. 1997. Qualitative research in information systems: time to be subjective? In *Information systems and qualitative research*, 444–465. Springer.

Greenwood, Royston, Raynard, Mia, Kodeih, Farah, Micelotta, Evelyn R., & Lounsbury, Michael. (2011). Institutional complexity and organizational responses. *The Academy of Management Annals, 5*(1), 317–371.

Heeks, Richard, & Arun, Shoba. (2010). Social Outsourcing as a Development Tool: The Impact of Outsourcing IT Services to Women's Social Enterprises in Kerala. *Journal of International Development, 22*(4), 441–454.

Hefley, Bill, & Babin, Ron. (2013). *Outsourcing Professionals' Guide to Coporate Responsibility*. Netherlands: Van Haren Publishing.

Ismail, S. A., Heeks, R., Nicholson, B., & Aman, A. (2018). Analyzing conflict and its management within ICT4D partnerships: an institutional logics perspective, *Information Technology for Development*, *24*(1), 165–187. https://doi.org/10.1080/02681102.2017.1320962.

Jay, Jason. (2013). Navigating paradox as a mechanism of change and innovation in hybrid organizations. *Academy of Management Journal, 56*(1), 137–159.

Khan, Shaji, Lacity, Mary, & Carmel, Erran. (2018). Entrepreneurial impact sourcing: a conceptual framework of social and commercial institutional logics. *Information Systems Journal, 28*(3), 538–562. https://doi.org/10.1111/isj.12134.

King, Nigel. (2004). Using templates in the thematic analysis of texts. In Catherine Cassell & Gillian Symon (Eds.), *Essential guide to qualitative methods in organizational research* (pp. 256–270). London, UK: Sage Publications Ltd.

Kraatz, Matthew S., & Block, Emily S. (2008). Organizational implications of institutional pluralism. In R. Greenwood, C. Oliver, R. Suddaby, & K. Sahlin-Andersson (Eds.), *The Sage Handbook of Organizational Institutionalism*. UK: Sage Publications Ltd.

Lacity, Mary C., Khan, Shaji, Yan, Aihua, & Willcocks, Leslie P. (2010a). A review of the IT outsourcing empirical literature and future research directions. *Journal of Information Technology, 25*(4), 395–433.

Lacity, Mary, Rottman, Joseph, & Khan, Shaji. (2010b). Field of dreams: Building IT capabilities in rural America. *Strategic Outsourcing: An International Journal, 3*(3), 169–191.

Lacity, Mary C., Solomon, Stan, Yan, Aihua, & Willcocks, Leslie P. (2011a). Business process outsourcing studies: a critical review and research directions. *Journal of Information Technology, 26*(4), 221–258.

Lacity, Mary, Carmel, Erran, & Rottman, Joseph. (2011b). Rural outsourcing: Delivering ITO and BPO services from remote domestic locations. *IEEE Computers, 44*(12), 55–62.

Lacity, Mary, Rottman, Joseph, & Carmel, Erran. (2012). *Emerging ITO and BPO Markets: Rural Outsourcing and Impact Sourcing*. USA: IEEE Ready Notes Series, IEEE Computer Society.

Lacity, Mary, Rottman, Joseph, & Carmel, Erran. (2015). Prison sourcing:'doing good'or 'good for business' & quest. *Journal of Information Technology Teaching Cases, 4*(2), 99–106.

Lacity, Mary, Rottman, Joseph W., & Carmel, Erran. (2014). Impact sourcing: Employing prison inmates to perform digitally-enabled business services. *Communications of the Association for Information Systems, 34*(1), 914–932.

Lounsbury, Michael. (2007). A tale of two cities: Competing logics and practice variation in the professionalizing of mutual funds. *Academy of Management Journal, 50*(2), 289–307.

Madon, Shirin, & Sharanappa, Sandesh. (2013). Social IT outsourcing and development: Theorising the linkage. *Information Systems Journal, 23*(5), 381–399.

Malik, Fareesa, Brian Nicholson, and Sharon Morgan. 2016. Assessing the Social Development Potential of Impact Sourcing. In *Socially Responsible Outsourcing: Global Sourcing with Global Impact*, eds. Mary C. BrLacity, and Brian Nicholson, and Ron Babin. United Kingdom: Palgrave Macmillan UK.

Malik, F., & Nicholson, B. (2019). Understanding the interplay of institutional logics and management practices in impact sourcing. *Information Systems Journal*, *30*(1), 125–149.

Malik, F., Nicholson, B., & Heeks, R. (2017). *Understanding the development implications of online outsourcing*. The 14th International Conference on Social Implications of Computers

in Developing Countries (IFIP WG 9.4) IFIP Advances in Information and Communication Technology Series (Vol. 504). Yogyakarta, Indonesia: Springer Link.
Meyer, John W., & Rowan, Brian. (1977). Institutionalized organizations: Formal structure as myth and ceremony. *American Journal of Sociology, 83*(2), 340–363.
Miles, Matthew B., Michael, A., & Huberman, (1994). *Qualitative data analysis: An expanded sourcebook*. Thousand Oaks, USA: Sage Publications Ltd.
Monitor. 2011. Job creation through building the field of impact sourcing. http://www.rockefellerfoundation.org/media/download/c2cbb5d3–500c-4ced-8387-a7469b567b98. Accessed 20 November 2012.
Mughal, M. A., & Zeb., (2014). *Time, space and social change in rural Pakistan: An ethnographic study of Jhokwala village*. Lodhran district: Durham University.
Pache, Anne-Claire, & Santos, Filipe. (2013). Inside the hybrid organization: Selective coupling as a response to conflicting institutional logics. *Academy of Management Journal, 56*(4), 972–1001.
Porter, Michael, E., and Mark Kramer, R. 2011. Creating shared value. *Harvard Business Review* 89 (12):62–77.
Powell, Walter W., & Sandholtz, Kurt W. (2012). Amphibious entrepreneurs and the emergence of organizational forms. *Strategic Entrepreneurship Journal, 6*(2), 94–115.
Reay, Trish, Robert, C., & Hinings., (2009). Managing the rivalry of competing institutional logics. *Organization Studies, 30*(6), 629–652.
Rehman, Sumaira, & Roomi, Muhammad Azam. (2012). Gender and work-life balance: a phenomenological study of women entrepreneurs in Pakistan. *Journal of Small Business and Enterprise Development, 19*(2), 209–228.
Sandeep, M., & Ravishankar, M. (2015a). Impact sourcing ventures and local communities: a frame alignment perspective. *Information Systems Journal*, *26*(2), 127–155.
Sandeep, M., & Ravishankar, M. (2015b). Social innovations in outsourcing: An empirical investigation of impact sourcing companies in India. *The Journal of Strategic Information Systems*, *24*(4), 270–288.
Seelos, Christian, Mair, Johanna, Battilana, Julie, Tina, M., & Dacin., (2011). The embeddedness of social entrepreneurship: Understanding variation across local communities. In Christopher Marquis, Michael Lounsbury, & Royston Greenwood (Eds.), *Research in the Sociology of Organizations* (pp. 333–363). USA: Emerald Group Publishing Limited.
Smets, Michael, Jarzabkowski, Paula, Burke, Gary T., & Spee, Paul. (2015). Reinsurance trading in Lloyd's of London: Balancing conflicting-yet-complementary logics in practice. *Academy of Management Journal, 58*(3), 932–970.
Smets, Michael, Morris, T. I. M., & Greenwood, Royston. (2012). From practice to field: A multi-level model of practice-driven institutional change. *Academy of Management Journal, 55*(4), 877–904.
Stake, Robert E. (1995). *The art of case study research*. Thousand Oaks, USA: Sage Publications Ltd.
Thornton, Patricia H., & Ocasio, William. (2008). Institutional Logics. In Royston Greenwood, Christine Oliver, Kerstin Sahlin, & Roy Suddaby (Eds.), *Organizational Institutionalism*. London, UK: Sage Publications Ltd.
Thornton, Patricia H., Ocasio, William, & Lounsbury, Michael. (2012). *The institutional logics perspective: A new approach to culture, structure, and process*. NY, US: Oxford University Press.
Urquhart, Cathy. (2013). *Grounded theory for qualitative research: A practical guide*. London, UK: Sage Publications Ltd.
Walsham, G. (1995). Interpretive case studies in IS research: nature and method. *European Journal of Information Systems, 4*(2), 74–81.
Walsham, G. (2006). Doing interpretive research. *European Journal of Information Systems, 15*(3), 320–330.

Adopting and Innovating Cloud Services

The Differential Benefits of Cloud Computing for Small and Medium Versus Large Firms

Laxmi S. V. Gunupudi and Raijv Kishore

Abstract Cloud computing is a fundamental shift in the way computing power is acquired and consumed. It is a special case of IS outsourcing where some IT-related tasks and resources are outsourced to cloud vendors. Cloud computing confers unique capabilities such as heterogeneity, scalability, accessibility, availability, consumption based pricing, fully managed and standardized services on the adopting organizations. These capabilities in-turn result in several strategic and operational benefits to the adopting organizations. However, these capabilities are not leveraged by all adopting firms uniformly. Firms utilize these capabilities based on their orientation towards exploration and exploitation. While SMEs are more aligned towards exploratory activities, large enterprises are inclined towards exploitation. Therefore, we posit that the SMEs and large enterprises accrue differential benefits from cloud adoption such that SMEs leverage cloud to attain strategic benefits and large enterprises seek operational benefits from cloud adoption. Through a survey based exploratory study we found that there is a systematic difference between SMEs and large enterprises in the way they leverage cloud capabilities. Based on the results, we deduce that managers of SMEs believe that they can use cloud computing for exploration and on the contrary, managers of large enterprises believe that they can use cloud computing for exploitation.

1 Introduction

Global competition is driving organizations to reduce costs, increase profitability and enhance productivity (Misra and Mondal 2011). This competition imposes enormous pressure on organizations to adopt latest innovations that enable them to reduce

L. S. V. Gunupudi
Indian Institute of Management Bangalore, Bangalore, India
e-mail: laxmi.gunupudi@gmail.com

R. Kishore (✉)
Management, Entrepreneurship & Technology Department, University of Nevada, Las Vegas, USA
e-mail: rajiv.kishore@unlv.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_10

costs, sustain competitive advantage and improve the bottom line (Demirkan et al. 2009). Digital transformation has become imperative for strategic innovation activities of firms. Many firms are recognizing the power of digital technologies and are crafting strategies around new products and services (Rai et al. 2012). Cloud computing is enabling and supporting digital transformation. Increasingly, it serves to support supply chains, marketing and service operations and other functional areas as well (Bharadwaj et al. 2013). Adoption of cloud based services can help organizations become more agile (Lacity and Willcocks 2013, 2016; Sambamurthy et al. 2003; Steininger 2019). Prior research has identified several characteristics of cloud services that promote cloud adoption by organizations (Low et al. 2011; Oliveira et al. 2014). Cloud adoption is in many ways similar to the tradition IT outsourcing model based on multi-tenant architecture. Vendors provide a variety of IT resources which are pooled across various customers. Hence, they are more efficient in handling demand and technological uncertainties and provide more flexibility for the firm.

Cloud computing has fundamentally shifted the way computing power is acquired and consumed. It is posited as a fundamental shift in the way computing power is acquired and consumed. It is claimed to offer numerous benefits to the adopting organizations and to have the potential to become the next major driver of business innovation. It is a sharp departure from the scenario where companies own their software and hardware and keep them on-premises in data centers and other specialized facilities.

Cloud computing promises numerous benefits to the adopting organizations. It provides the consumers with a capability to use applications and infrastructure services provided by the cloud computing vendors without having to manage or control the infrastructure or the applications. To the vendor organizations, it promises to enable new business models and services across many industries and provide access to new market segments and new revenue streams. It is a fundamental shift in the way supplying organizations deliver computing services and the way consuming organizations perceive and use those services. Cloud computing is regarded as a favorable solution to enhance a modern organization's IT performance and competitiveness.

Organizations can outsource their IT related operational activities to cloud providers and focus on their core competencies (Glaser 2011). Leveraging the capabilities offered by cloud services helps organizations respond to changing business needs very effectively. Since upfront investment is not required, organizations can reduce the risks of experimentation with new products and new business models (Iyer and Henderson 2012). This flexibility in experimentation lowers the barriers to innovation and encourages new players to enter the markets (Marston et al. 2011). Organizations can also expand their global business operations as cloud services can be accessible anywhere and at anytime (Christauskas and Miseviciene 2012). Therefore we can say that cloud computing has the potential to support both exploratory and exploitative activities of firms. However, all the adopting firms may not accrue all these benefits uniformly. They may be contingent upon contextual factors such as firm size. SMEs and large enterprises have different orientation towards exploration and exploitation. While most capabilities provided by cloud are beneficial for all the adopting firms, large enterprises and small and medium enterprises (SMEs) may leverage different capabilities due to their different exploration and exploitation orientations, and may accordingly gain different benefits from cloud computing. Literature suggests that firm size is an important factor that influences the perception of capabilities provided by IT and the benefits that accrue to the organization (Kerin et. al. 1992; Lee and Chen 2009). In this paper we seek to address the following research questions in the intersection of cloud computing and innovation:

> How do SMEs and large enterprises differ in the way they leverage cloud computing capabilities and accrue the benefits of adoption?

In this study, using the lens of exploration and exploitation, we propose that SMEs and large enterprises differ in their orientation towards exploration and exploitation such that they leverage capabilities of cloud differently and derive different benefits from cloud adoption. We posit that, SMEs are more aligned towards exploratory actions and therefore they utilize capabilities of cloud to derive strategic benefits. On the contrary, large enterprises would like to exploit their existing resources to leverage their scale and scope, and therefore focus on those capabilities of cloud which result in operational benefits. In this study, we examine this phenomenon through an exploratory study. We begin with the study of evolution of cloud model and examine the unique characteristics of cloud computing. We follow this up with the identification of capabilities conferred by cloud model for adopting firms. Based on a survey aimed at understanding the perception of IT managers about the capabilities of cloud computing and the benefits that result from adoption, we examine the differences between Small and Medium Enterprises (SMEs) and large enterprises in their beliefs of capabilities of cloud and the nature of benefits that accrue to them upon adoption.

2 Cloud Computing: Evolution and Characteristics

The working definition published by the US National Institute of Standards and Technology (NIST) (Mell and Grance 2011) captures the most commonly agreed aspects of cloud computing:

> a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (for example, networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Cloud computing services are classified along two dimensions: the nature of services being offered and the deployment methodology. Based on the service models, there are three major classes of services being offered by cloud: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) (Armbrust et al. 2010). According to the mode of deployment, there are four classes of cloud models (Mell and Grance 2011): Private cloud, Community Cloud, Public Cloud and Hybrid Cloud. Service models can be deployed on top of any of the deployment models. Figure 1a, b provide an overview of this classification.

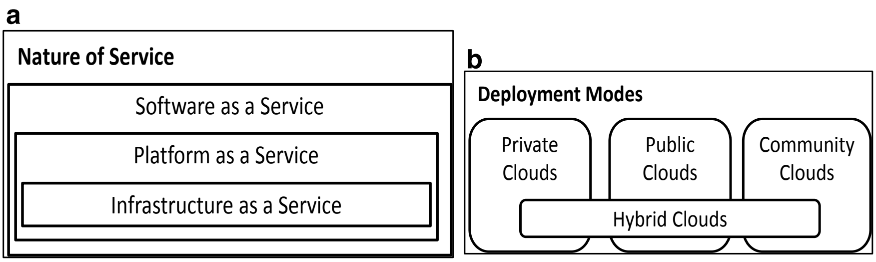

Fig. 1 **a** Nature of service. **b** Deployment modes

The word "Cloud Computing" is not associated with any particular technology, protocol or vendor. It is a model which is both a technological innovation and a business imperative. It is an amalgamation of many years of technological innovations and developments (Goodburn and Hill 2010). Cumulative developments in various fields contributed to the development of cloud computing model. The features of cloud computing, such as hosting of software and hardware in data centres of suppliers, the rent of information technology professionals, are not new (Budrienė and Zalieckaitė 2012). All these have been made use of by organizations for years. However, the cloud computing concept emerged when the technologies supporting it, namely, broadband internet connection and virtual solutions, emerged. Cloud computing is not only a technology offering the services tailored for businesses, but also a new business model (Budrienė and Zalieckaitė 2012). Developments in the fields of computer science, telecommunications, design paradigms and business models contributed to the cloud model. These developments can be summarized as follows.

2.1 Developments in the Field of Computer Science

Virtualization, time sharing, grid computing, Web 2.0 and ubiquitous computing are some of the technologies in the field of computer science that form the building blocks of cloud computing. Virtualization allows a single physical machine to be divided into multiple virtual instances that ensure better utilization of resources also has ability to shift virtual instances from one machine to another. Virtualization which is achieved by hypervisors gives the power of seamless scalability.[1] Virtualization also enabled device independence aspect of ubiquitous computing for cloud computing. This enables information and tasks to be made available everywhere and to support intuitive human usage (Poslad 2011). Virtualization also enables the abstraction of IT resources and access to these resources from across heterogeneous devices. This made cloud services more dynamic such that they can be provisioned on-demand based on self-service.

[1]Scalability—Scalability is the ability of a system, network, or process to handle a growing amount of work in a capable manner or its ability to be enlarged to accommodate that growth.

Time sharing and grid computing are fundamentally based on the concepts of resource sharing and resource pooling. Enabling multiple end users to share equipment allows higher utilization of the underlying hardware. The reductions in the variability resulting from aggregating demand help cloud computing firms achieve higher utilization rates than individual companies can achieve on their own.

2.2 Developments in the Area of Telecommunications

Developments in the field of telecommunications, like high-speed internet connections, mobile telephony, and mobile broadband, support cloud computing with connectivity and access. These developments also contribute to location independence aspect of ubiquity for cloud computing. Mobile broadband coupled with the strength of virtualization also provided device independence.

2.3 Developments in the Area of Design Paradigms

Service orientation, which is a design paradigm, and SOA architecture, which is governed by the principles of service orientation, contributed significantly to the cloud model. Service orientation defines a set of design principles to help build computer software in the form of services SOA allows smooth cooperation of a large number of computers that are connected over a network. These principles make it easier to configure cloud-based IT resources.

2.4 Developments in the Field of Business Models

Business models are essential for extracting commercial value from new innovations/technologies (Sako 2012). Thus, the business model can be a subject of innovation in itself, pursuing novel forms of value creation and capturing mechanisms. Business models based on consumption based pricing are a significant contribution towards cloud computing. This new business model has transformed the provision and consumption models of IT resources.

Table 1 lists these concepts/technologies/innovations that form the building blocks of cloud computing in a chronological order and their respective contribution towards cloud computing. While cloud computing draws many characteristics from its underlying technologies and innovations, it has its own unique characteristics that can be referred to as emergent characteristics. These emergent features are scalability/Elasticity and availability. Elasticity refers to resource allocation where the resources are provided in the amount required and disposed of when no longer needed (Durkee 2010). Elasticity enables scalability where the resources can be scaled up

Table 1 Developments leading to cloud computing

Underlying concept	Cloud computing characteristics
Time Sharing-1955	Sharing of resources across multiple customers → Resource pooling/aggregation of demand → leads to multi-tenancy (Campbell-Kelly 2009)
Virtualization-1960	Abstracted infrastructure and virtualized physical resources → leads to multi-tenancy (Owens 2010)
Internet-1985	Connectivity → leads to location ubiquity (Marston et al. 2011)
Mobile broadband-1991	Mobile technology innovations → leads to device independence form of ubiquity (Marston et al. 2011)
Grid Computing-1990	Variety of IT resources, optimum utilization—efficiency → leads to economies of scale (Foster et al. 2008)
Utility Computing-1995	Service offering of IT resources, Measured service; → leads to pay-per-service business model (Yoo 2011)
SOA-1996	Based on prior design paradigms aimed at modularity and loose coupling → leads to agility and configurability (Creeger 2009)
Web 2.0-1999	Based on interactive and dynamic browser experience → leads to rapid, dynamic, on-demand access based on self-service (Jamil 2009)

and down rapidly. Availability is the proportion of time the service is in a functioning condition.

3 Cloud Computing: Unique Characteristics and Capabilities

From the previous section, characteristics of cloud computing model can be summarized as follows. Cloud computing provides access to abstracted or virtual IT resources. It provides the necessary abstraction such that the underlying fabric can be unified as a pool of resources. This abstraction of end-user applications from the underlying hardware is critical to cloud model. It means that application software is not tied to any particular server or physical hardware. Instead, it can utilize the massive scalability and resiliency of the underlying global scale datacenters to deliver the same services to one user or several million users (Kushida et al. 2011). It enables each application to be encapsulated such that they can be configured, deployed, started, migrated, suspended, stopped, etc. and these provide better security, manageability, and isolation. Cloud computing promises a very high availability of IT resources. These resources are pooled and shared across multiple customers. Cloud providers thus benefit from economies of scale and efficient utilization of resources. Virtualization together with resource pooling enables multitenant architecture. Multi-tenancy refers to the concept where a single instance of the application runs on a server, serving multiple client organizations (tenants). This allows better utilization of a system's resources (in terms of memory and processing overhead), the requirements of which could otherwise be considerable if the software instance had to be duplicated for each client (Marston et al. 2011). These resources are provided as services that are charged based on the consumption of resources. The above mentioned, essential characteristics of cloud enable some IT capabilities to the adopting organizations. Following are the capabilities enabled by these unique characteristics of the cloud for the adopting firms.

3.1 Heterogeneity

Cloud model provides access to heterogeneous IT resources such as hardware, systems software, and services (Vaquero et al. 2008). Firms can pick and choose the IT resources they require from cloud market.

3.2 Scalability

Cloud resources are extremely scalable and can be provisioned rapidly based on need/demand. Elasticity refers to resource allocation where the resources are provided in the amount required and disposed of when no longer needed (Durkee 2010). Elasticity enables scalability where the resources can be scaled up and down rapidly.

3.3 Consumption Based Pricing

Cloud computing is based primarily on the business model of 'pay-per-use'. Resources are offered as services that are charged based on usage (Armbrust et al. 2009). Measured services provided by cloud markets imply metering of the services in order to monitor, control and report usage. Thus usage is measured, and consumers are charged based on this measure (Brynjolfsson et al. 2010).

3.4 Fully Managed

Cloud services are entirely managed by cloud vendors (Anthes 2010; Mircea et al. 2011; Vaquero et al. 2008). It is similar to outsourcing of services from third-party offerings. Cloud services are provisioned as services that are totally managed by the service providers (Bardhan et al. 2010). Developments such as web 2.0 coupled with virtualization provide on-demand, dynamic and automatic access based on self-service. Services are also based on SOA, with makes it very modular and configurable

(Yoo 2011). It is very convenient for the managers to provision IT resources with minimal management effort (Vaquero et al. 2008).

3.5 Standardized Services

Since cloud services are offered to several customers, services provided by cloud vendors are more standardized (Aljabre 2012). These services have generic functionalities (Géczy et al. 2012) and standardized technical specifications and interfaces.

3.6 Availability

Resource pooling and redundant equipment invested by cloud vendors ensures the provision of services with minimal downtimes and system failures. Thus, cloud vendors promise a very high availability of these resources (Leavitt 2009). Availability is one of the fundamental design features of cloud offering (Youseff et al. 2008).

3.7 Accessibility

These services are provided over internet making them accessible from anywhere (Farah 2010). This accessibility lends location independence to these services. With advances in virtualization, these services are provided across a variety of devices also making them device independent. Device location and time independence together make cloud computing ubiquitous (Marston et al. 2011).

These afore mentioned capabilities provided by cloud computing can be utilized by adopting organizations to result in several benefits. However, these capabilities may not be leveraged uniformly by all adopting organizations. They can be differently leveraged across any of the thee service models or deployment modes (depicted in Fig. 1a, b). Firms leverage these capabilities based on their orientation towards exploration and exploitation. The following section elaborates on the exploration and exploitation related actions of a firm.

4 Exploration and Exploitation

Exploration and exploitation emerged as a strong underlying theme researched in various streams including organizational learning and strategy (Levinthal and March 1993), innovation (Danneels 2002) and entrepreneurship (Shane and Venkataraman

2000). Theory of learning and actions suggests that organizational actions can be classified as exploration and exploitation (Benner and Tushman 2003; March 1991).

Exploitative actions are aimed towards improving, applying, and incrementally refining firm capabilities. The goal of these actions is to improve operational efficiencies. The outcomes for these innovations are clearly defined in terms of cost reductions, process consistency, process efficiency etc. These actions therefore lead to operational benefits in the form of process improvements and cost efficiencies (Subramani 2004).

In contrast, the exploratory innovations aim towards creating new capabilities and devising solutions to current problems. Their goal is to learn about the environment and discover novel ways of creating value or solving old problems. Exploration reflects innovation and related risk-taking and experimentation. Therefore, the outcomes of such activities result in strategic benefits for the organization (Subramani 2004).

These two types of activities also differ in their task orientations. While exploitation involves strategies that reduce variance so as to streamline activities and improve efficiencies, exploration involves strategies that seek variation in order to develop novel solutions. Certain firm level characteristics have an impact on firm's orientation towards either exploration or exploitation. In the sections that follow, we examine the role of firm size as a contingent factor which influences a firm's orientation towards exploration or exploitation.

5 Firm Size: SMEs Versus Large Enterprises

Firm size is an indicator of the ability to provide resources to manage uncertainties and to support new product development. Extant research suggests that large firms accumulate substantial resources and evolve such that they have the ability to transform new products into firm value. For example, large firms have greater knowledge of the market and customer needs and they can easily convert their new product development programs to firm value. We can therefore say that, larger firms have economies of scope and scale and other marketing capabilities that aid in developing and launching their new products faster and perhaps even better than smaller firms to enjoy the first mover advantage (Kerin et. al. 1992; Lee and Chen 2009). However, scholars have also argued that small firms are likely the major sources of innovation (e.g., Schumpeter 1934). Market expects that small firms are more inclined to revamp current competence to become more innovative than larger firms (Lee and Chen 2009). We, therefore, posit that the perception of capabilities of cloud computing and the resulting benefits accrued by SMEs and large enterprises could be different. We use the theoretical lens of exploration and exploitation to understand these differences.

Taking this perspective of firm actions, we posit that larger firms which are heavily invested in resources, focus on exploitative actions to leverage their scale and scope, whereas SMEs which are very agile, focus on innovation and exploratory activities.

In the following section, we discuss the operational and strategic benefits that result from exploratory and exploitative actions leveraging cloud capabilities in more detail.

6 Benefits of Cloud Adoption

Capabilities provided by cloud computing result in several benefits for the adopting organizations. Subramani (2004) identified and classified first order benefits of such IT use as operational and strategic benefits. Operational benefits arise from reduced costs and improved processes as a result of IT use. In contrast, strategic benefits arise through firms positioning themselves to take advantage of new opportunities that arise as a result of IT use. These include the development of new products and services, a richer understanding of the customers and markets etc. In the context of cloud computing these benefits can be analysed as follows.

6.1 Operational Benefits

Operational benefits are defined as the benefits that arise from (a) cost efficiencies from higher sales volumes and (b) improvements to current processes or creation of new processes (Subramani 2004). These refer to improvements made to the internal functioning of the organization apparent in everyday activities.

Organizations can lower their IT related costs by migrating to cloud and accrue financial benefits. They need not invest upfront in IT resources and thereby reduce capital expenditure (Armbrust et al. 2009). Consumption based pricing model of cloud ensures that firms only pay for what they use. These firms incur only operational expenses. This reduces the costs significantly. As cloud services are fully managed, adopting organizations can also reduce their costs related to in-house IT personnel and costs resulting from delays in building and deploying applications (Harry Katzan 2010). Similarly, cloud based services provide generic functionalities with standard specifications and technical interfaces (Aljabre 2012; Géczy et al. 2012). This can therefore help managers streamline the internal operations of the firm and make them more efficient. Therefore adoption of cloud computing can lead to operational benefits.

6.2 Strategic Benefits

Strategic benefits are the benefits that arise from (a) learning about customers and markets for the firm's products; (b) creation of new products, product enhancements; and (c) development of new business opportunities. These relate to the development of corporate strategies through the building of relationship with customers, suppliers and competitors.

Cloud adoption provides several strategic benefits to organizations where they can come up with new lines of businesses and products more economically and rapidly to the market. First, cloud based resources are fully managed by the vendors and adopting organizations can therefore outsource their IT related operational activities and focus on their core competencies (Glaser 2011). Second, access to heterogeneous resources helps employees enhance their skills and provides them with opportunities to learn (Armbrust et al. 2010). In addition, cloud computing provides several new capabilities to the organizations (Bala Iyer and Henderson 2010) such as the scalability which enable firms to match IT resources to the demand requirements. It creates an Illusion of availability of infinite computing resources thereby eliminating the need to plan far ahead for provisioning (Armbrust et al. 2009). Cloud scalability makes management of demand fluctuations easy for managers (Aslam et al. 2010). It becomes easier to manage demand fluctuations rapidly and economically based on the requirements. This results in a lean organization that can respond to environmental changes and changing business needs very effectively.

While most of these capabilities are beneficial for all the adopting firms, large enterprises and SMEs may leverage different capabilities provided by cloud to receive differentiated benefits. Therefore, we propose that: (a) SMEs leverage those capabilities of cloud computing that help them conducting exploration and therefore derive strategic benefits; and (b) Large enterprises leverage those capabilities of cloud computing that support exploitation of existing resources and therefore derive operational benefits. To understand the exact nature of the capabilities leveraged by each of them and the benefits derived, we conducted an exploratory study. In the following section we describe the study in more detail.

7 Methodology

To investigate our research question, we conducted a survey to understand the perception of IT managers about the capabilities of cloud computing model and the benefits that accrue to them. The key executives who participated in the study are responsible for taking IT procurement decisions in organizations. Several of the respondents were in top management positions (typical designations of CEO, CIO, and IT Head etc.) in their organizations and had work experience greater than 15 years in the industry. Hence, these informants are likely to be the most knowledgeable informants to provide relevant data (Bagozzi and Phillips 1982) in the context of cloud adoption. Data was collected from firms across different industry segments (SIC code 73—Business Services n = 13; SIC code 87—Engineering, Accounting, Research, Management, And Related Services n = 11; SIC code 82—Education Services n = 4; SIC code 80—Health Services n = 3; 'other' n = 14). In total, data was collected from managers across 45 firms. 21 of these firms are SMEs (Small and Medium Enterprises with sales revenue less than equal to 10 million USD) and 24 are large enterprises (with sales revenue greater than 10 million USD). Perception about the extent to which cloud provides each of these capabilities and the extent to which the firms derive benefits from these capabilities is measured using a seven point likert scale (1 is none; 4—To some extent; 7—To a great extent). 45 meetings were conducted for gathering organization level capabilities data (all 45 firms) and split-samples of SMEs (21 firms) and large enterprises (24 firms). Table 2 provides the scales used in the survey. Wherever applicable, existing measurement items were used to develop the scales after modifying for the current context of the study. Data was collected through face to face meetings with key informants where the structured survey questionnaire is administered by the researcher.

The items of perceptions of each of the capabilities of cloud computing have been aggregated. This is followed by assessing the correlation between each of these capabilities and the individual items of operational and strategic benefits constructs. Table 3 provides the results of the correlation analysis.

8 Results and Discussion

Table 4 provides the interpretation of results from correlation analysis. These results clearly demonstrate the differences between the perceptions of SMEs and large enterprises about cloud computing capabilities and the resulting benefits. We discuss these differences along each of the capabilities and the resultant benefits for SMEs and large enterprises in this section.

8.1 Heterogeneity

SMEs: The focus of SMEs is frugal innovation. They are generally operate with limited resources. Cloud based services provide access to state-of-the-art heterogeneous resources which is otherwise very difficult for any SMEs with limited resources to procure and manage in-house. These resources include (a) hardware components such as storage, databases, servers, network components; (b) security solutions; (c) development environment; (d) application software; (e) server management skills; and (f) application development skills. Therefore, organizations benefit from access to state-of-the-art technologies and skilled personnel (Garrison et al. 2012). Since upfront investment is not required, organizations can reduce the risks of experimentation with new products and new models (Iyer and Henderson 2012). This flexibility in experimentation lowers the barriers to innovation and encourages new entrants to the markets (Marston et al. 2011). Cloud adoption therefore helps SMEs seek new business opportunities and enter new markets and reach out to new customers.

Table 2 Survey instrument

Variable	Questions	Scale
Heterogeneity	The extent to which cloud market offers the following IT resources and reduces to have the following in-house Hardware components such as storage, databases, servers, network components Security solutions Development Environment Application Software (eg. ERP, CRM etc.) Server management (Load Balancing, Virtualization) skills Application development skills Application deployment skills	1—None , 4—To some extent, 7—To a great extent
Fully managed	The extent to which adopting cloud market reduces the need to manage: database backups disaster recovery upgrade management	1—None , 4—To some extent, 7—To a great extent
Standardized services	The extent to which cloud based IT resources: have generic (industry wide) functionality have standardized technical specifications have standardized interfaces	1—None , 4—To some extent, 7—To a great extent
Scalability	The extent to which cloud based IT resources: Can be economically expanded or contracted for allocation based on requirements can be rapidly expanded or contracted for allocation based on requirements can economically handle inconsistent loads of traffic can rapidly handle inconsistent loads of traffic	1—None , 4—To some extent, 7—To a great extent
Availability	The extent to which cloud market provides IT infrastructure / systems with: minimal downtimes, minimal system failures, very high availability in an economic manner	1—None , 4—To some extent, 7—To a great extent

(continued)

Table 2 (continued)

Variable	Questions	Scale
Consumption based pricing	The extent to which cloud market provides IT resources: on a chargeback or pay-per-use basis, with minimal capital expenditure, where you only pay for actual resource usage	1—None , 4—To some extent, 7—To a great extent
Accessibility	The extent to which cloud market provides access to IT resources independent of: location, device, time	1—None , 4—To some extent, 7—To a great extent
Operational benefits	Please indicate the extent to which you are receiving the following benefits of IT use in your organization. Improved cost efficiencies. Improvements to current processes or creation of new processes	1—None , 4—To some extent, 7—To a great extent
Strategic benefits	Please indicate the extent to which you are receiving the following benefits of IT use in your organization. Development of new business opportunities. Creation of new products or product enhancements. Learning about customers or markets for our products	1—None , 4—To some extent, 7—To a great extent

8.2 Scalability

SMEs: Cloud markets are based on multi-tenant architecture and cater to several customers. Hence, they are more efficient in handling demand uncertainties and provide more flexibility for the firm. Cloud adoption can help SMEs become agile and more responsive to uncertain environments by providing them the capability of rapid scaling. Cloud service's ability to add and remove resources at a fine-grain and with a very small lead time helps SMEs manage their variable workloads more efficiently (Armbrust et al. 2009). It makes it easier for SMEs to seek new business opportunities without having to make huge investments on IT resources.

Large Enterprises: Variations in demand or market collapse may arise due to various reasons such as change in economic conditions, customer interests, or competitor initiatives (Iyer and Henderson 2012). Firms must be very flexible in managing such variations to avoid losing market share to competitors. In order to stay competitive firms have to engage in activities to understand customers better and to introduction

Table 3 Correlation analysis

	Small and medium enterprises			Small and medium enterprises		Large enterprises			Large enterprises	
	Strategic benefits			Operational benefits		Strategic benefits			Operational benefits	
	New business opportunities	New products or product enhancements	Learning about customers and markets	Cost efficiencies	Improved processes	New business opportunities	New products or product enhancements	Learning about customers and markets	Cost efficiencies	Improved processes
Heterogeneity	**0.458 (0.037)**	**0.395 (0.036)**	**0.475 (0.030)**	0.157 (0.497)	0.270 (0.236)	−0.131 (0.540)	0.649 (0.763)	−0.292 (0.165)	0.283 (0.180)	0.018 (0.932)
Scalability	**0.418 (0.049)**	0.298 (0.189)	0.328 (0.147)	0.163 (0.478)	0.057 (0.806)	0.169 (0.427)	0.773 (0.720)	**0.369 (0.048)**	0.250 (0.239)	0.006 (0.976)
Availability	0.291 (0.201)	0.241 (0.292)	**0.391 (0.039)**	0.147 (0.527)	0.167 (0.469)	0.278 (0.792)	0.256 (0.228)	0.257 (0.227)	0.607 (0.117)	0.256 (0.905)
Accessibility	**0.479 (0.029)**	**0.383 (0.047)**	0.350 (0.120)	0.031 (0.893)	−0.027 (0.901)	0.036 (870)	0.000 (1.000)	0.143 (0.505)	0.158 (0.460)	**0.343 (0.005)**
Consumption based pricing	0.342 (0.129)	0.199 (0.385)	0.056 (0.807)	**0.341 (0.043)**	0.064 (0.782)	0.146 (0.495)	0.115 (0.592)	0.005 (0.997)	**0.397 (0.019)**	0.068 (0.752)
Fully managed	**0.460 (0.036)**	0.325 (0.128)	0.392 (0.078)	0.125 (0.588)	0.009 (0.966)	0.272 (0.198)	0.041 (0.848)	0.262 (0.214)	**0.332 (0.024)**	**0.349 (0.040)**
Standardized services	**0.390 (0.030)**	**0.472 (0.031)**	0.163 (0.480)	0.052 (0.082)	0.001 (0.998)	0.208 (0.328)	0.153 (0.474)	0.494 (0.361)	**0.381 (0.021)**	**0.373 (0.033)**

Correlation coefficients and their p-values (in parentheses) are provided. Significant correlations (p-val < 0.05) are highlighted

Table 4 Interpretation of correlation analysis

	Small and medium enterprises			Small and medium		Large enterprises			Large enterprises	
	Strategic benefits			Operational benefits		Strategic benefits			Operational benefits	
	New business opportunities	New products or product enhancements	Learning about customers and markets	Cost efficiencies	Improved processes	New business opportunities	New products or product enhancements	Learning about customers and markets	Cost efficiencies	Improved processes
Heterogeneity	✓	✓	✓							
Scalability	✓							✓		
Availability			✓							
Accessibility	✓	✓								✓
Consumption based pricing				✓					✓	
Fully managed	✓								✓	✓
Standardized services	✓	✓							✓	✓

of new and creative products. The variations in demand are likely to entice firms to release new offers in order to reduce uncertainty and stay competitive. They create opportunities for above-normal return by targeting premium market segments (Levinthal and March 1993; Zahra et al. 1999) and creating new niches (Lumpkin and Dess 2001). It is posited that highly variable and unpredictable workloads are more suitable for cloud-based services (Misra and Mondal 2011). Therefore large enterprises make use to cloud scalability to understand customers and markets better.

8.3 Availability

SMEs: Cloud services can be accessible anywhere and at anytime and also promise high availability with minimum downtime and system failures. As a result, organizations can also expand their global business operations easily (Christauskas and Miseviciene 2012). Therefore, this kind of access to fully managed, state-of-the-art skills and resources which can be deployed easily helps SMEs get to market with new line of products and services quickly and easily.

8.4 Accessibility

SMEs: Cloud based resources provide access to IT resources independent of location, device and time. In the case of SMEs, this enables new product development and deployment across a variety of devices and helps them seek new business opportunities.

Large Enterprises: Through device and location independence, cloud adoption empowers employees of the adopting organization by allowing them to collaborate with one another anywhere/anytime (Goodburn and Hill 2010). This removes any process lags or inefficiencies. For large enterprises, therefore, cloud adoption enables process improvements through empowerment of employees.

8.5 Consumption Based Pricing

SMEs: Consumption based pricing of cloud services ensures that firms only incur expenditure for IT resources that they actually use (Armbrust et al. 2009). For SMEs this feature helps achieve cost efficiencies where firms only incur operational expenses and there is no capital expenditure. This helps SMEs achieve their objective of frugal innovation.

Large Enterprises: At the same time, large enterprises also benefit from this capability provided cloud services. They can leverage their economies of scale and drive higher cost efficiencies by minimizing capital expenditure.

8.6 Fully Managed

SMEs: SMEs can outsource their IT related operational activities to cloud providers and focus on their core competencies (Glaser 2011). Leveraging the capabilities offered by cloud services helps organizations respond to changing business needs very effectively.

Large Enterprises: Large enterprises make huge investments in IT infrastructure, personnel, platforms and software (Armbrust et al. 2010). Upon migration to cloud, IT services are fully managed by the vendors and it eliminates major infrastructure related tasks such as backup, disaster recovery and system management (Christauskas and Miseviciene 2012) and they are also free from the risk of technological obsolescence (Glaser 2011). This improves cost efficiencies for larger firms. Also, it makes it easier to streamline internal IT processes for large firms as the majority of infrastructure related tasks are fully managed by the cloud vendors.

8.7 Standardized Services

SMEs: Services provided by cloud vendors have standardized technical specifications and interfaces (Aljabre 2012; Géczy et al. 2012). This interoperability of services supports innovation and helps SMEs come up with new products and explore new business opportunities without having to invest much time and effort in integration across various systems/ sub-systems of their products.

Large Enterprises: Adoption of cloud based services which provide generic functionalities with standard specifications and technical interfaces (Aljabre 2012; Géczy et al. 2012) help managers of large enterprises streamline their business processes making them more efficient. They do not have to invest additional resources toward building additional interfaces and integration solutions. This helps them gain cost efficiencies and improved business processes.

From the above results we can infer that predominantly SMEs focus on exploration related activities and utilize cloud capabilities to derive strategic benefits. Similarly, large enterprises focus on exploitation related activities and utilize cloud capabilities to derive operational capabilities. However, we observed two discrepancies to our proposition. First, in the context of scalability we observed that large enterprises utilize cloud capability of rapid scaling for understanding customers and markets, which contributes to strategic benefits for the firm. Similarly, we also observe that SMEs make use of consumption based pricing feature of cloud to gain cost efficiencies, which contribute to operational benefits for the firm.

9 Conclusion and Future Research Direction

Cloud computing is special case of IS outsourcing where some IT-related tasks and resources are outsourced to cloud vendors. In this study, we trace the evolution of cloud computing and derive the unique characteristics of cloud model and the capabilities this model confers on the adopting organizations. These capabilities include heterogeneity, scalability, consumption based pricing and accessibility among others. Owing to these capabilities, cloud computing has the potential to support both exploratory and exploitative activities of firms which result in operational and strategic benefits for the adopting firms. However, these benefits are contingent upon several contextual factors such as firm size.

Using the lens of exploration and exploitation (Benner and Tushman 2003; March 1991), we propose that SMEs and large enterprises differ in their orientation towards exploration and exploitation such that they leverage capabilities of cloud differently and derive different benefits from cloud adoption. Prior research suggests that the goal of exploitation is to improve operational efficiencies and therefore it leads to operational benefits in the form of process improvements and cost efficiencies. Similarly, exploratory innovations aim towards creating new capabilities and devising solutions to current problems and therefore leads to strategic benefits. We posit that SMEs utilize those capabilities of cloud computing that help them conduct exploration and therefore derive strategic benefits. On the contrary, large enterprises leverage those capabilities of cloud computing that support exploitation of existing resources and therefore derive operational benefits. To understand the exact nature of the capabilities leveraged by each of them and the benefits derived, we conducted a survey based exploratory study.

Results of the study are in line with our propositions and suggest that large enterprises focus on utilizing cloud capabilities to support their exploitation and SMEs focus on cloud capabilities which help them support their exploration. This is primarily because large enterprises leverage the economies of scale and scope, whereas, SMEs focus on frugal innovation. Therefore SMEs predominantly derive strategic benefits from cloud adoption whereas large enterprises derive operational benefits.

Prior research suggests that benefits from cloud adoption are contingent upon several factors. Through this study, we make one important contribution to the research stream that focuses on benefits of cloud adoption. We highlight the importance of firm size as an important factor that influences the perception of capabilities provided by cloud and the benefits that accrue to the adopting organization. Future research can shed more light on other factors and contingent benefits of cloud adoption. While firm size is an important factor that we examined in this study, there are several vendor side factors such as vendor firm size, vendor capabilities etc. which require consideration in this context.

Cloud computing provides a platform to perform computing on a massive scale. Growth in cloud computing has led to extensive diffusion of technological paradigms such as Big Data, Artificial Intelligence, Machine Learning, Block Chain and Internet of Things (IoT). These developments have played a major role in leading digital transformations in sectors such as automotive, aerospace, retail, electronics, digital money etc. Future research in this area can also explore how this nexus of cloud computing, big data and IoT is transforming business strategies of organizations in various industry segments. Future research can also look at how unique capabilities conferred by cloud computing on the adopting firms which will enable them to take different strategic actions such as increased scale of operations (Chen et al. 2007), improved market diversity (Miller and Chen 1994), reduced price of products (Derfus et al. 2008) and introduction of new and improved products (Debruyne et al. 2002).

References

Aljabre, A. (2012). Cloud computing for increased business value. *International Journal of Business and Social Science, 3*(1), 234–239.

Anthes, G. (2010). Security in the cloud. *Communications of the ACM*, *53*(11).

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., et al. (2010). A view of cloud computing. *Communications of the ACM, 53*(4), 50–58.

Armbrust, M., Griffith, A. F. R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., … Zaharia, M. (2009). Above the clouds: A berkeley view of cloud computing. In *UC Berkeley Reliable Adaptive Distributed Systems Laboratory*.

Aslam, U., Ullah, I., & Ansari, S. (2010). Open source private cloud computing. *Interdisciplinary Journal of Contemporary Research in Business, 2*(7), 399–407.

Bagozzi, R. P., & Phillips, L. W. (1982). Representing and testing organizational theories: A holistic construal. *Administrative Science Quarterly*, 459–489.

Bardhan, I. R., Demirkan, H., Kannan, P., Kauffman, R. J., & Sougstad, R. (2010). An interdisciplinary perspective on IT services management and service science. *Journal of Management Information Systems, 26*(4), 13–64.

Benner, M. J., & Tushman, M. L. (2003). Exploitation, exploration, and process management: The productivity dilemma revisited. *Academy of Management Review, 28*(2), 238–256.

Bharadwaj, A., El Sawy, O. A., Pavlou, P. A., & Venkatraman, N. (2013). Digital business strategy: Toward a next generation of insights. *MIS Quarterly*, 471–482.

Brynjolfsson, E., Hofmann, P., & Jordan, J. (2010). Cloud Computing and Electricity: Beyond the utility model. *Communications of the ACM, 53*(5).

Budrienė, D., & Zalieckaitė, L. (2012). Cloud computing application in small and medium-sized enterprises. *Issues of Business & Law, 4*.

Campbell-Kelly, M. (2009). Historical reflections the rise, fall, and resurrection of software as a service. *Communications of the ACM, 52*(5), 28–30.

Chen, M.-J., Kuo-Hsien, S., & Tsai, W. (2007). Competitive tension: The awareness-motivation-capability perspective. *Academy of Management Journal, 50*(1), 101–118.

Christauskas, C., & Miseviciene, R. (2012). Cloud-computing based accounting for small to medium sized business. *Engineering Economics, 23*(1), 14–21.

Creeger, M. (2009). CTO roundtable: Cloud computing. *Communications of the ACM, 52*(8), 50–56.

Danneels, E. (2002). The dynamics of product innovation and firm competences. *Strategic Management Journal, 23*(12), 1095–1121.

Debruyne, M., Moenaertb, R., Griffinc, A., Hartd, S., Hultinke, E. J., & Robben, H. (2002). The impact of new product launch strategies on competitive reaction in industrial markets. *Journal of Product Innovation Management, 19*(2), 159–170.

Demirkan, H., Kauffman, R. J., Vayghan, J. A., Fill, H.-G., Karagiannis, D., & Maglio, P. P. (2009). Service-oriented technology and management: Perspectives on research and practice for the coming decade. *Electronic Commerce Research and Applications, 7*(4), 356–376.

Derfus, P. J., Maggitti, P. G., Grimm, C. M., & Smith, K. G. (2008). The Red Queen effect: Competitive actions and firm performance. *Academy of Management Journal, 51*(1), 61–80.

Durkee, D. (2010). Why cloud computing will never be free. *Queue, 8*(4), 20.

Farah, S. (2010). Cloud computing or software as a service—which makes the most sense for HR? *Employment Relations Today, 36*(4), 31–37.

Foster, I., Zhao, Y., Raicu, I., & Lu, S. (2008). *Cloud computing and grid computing 360° compared.* In Paper presented at the Grid Computing Environments Workshop, 2008. GCE'08.

Garrison, G., Kim, S., & Wakefield, R. L. (2012). Success factors for deploying cloud computing. *Communications of the ACM, 55*(9), 62–68.

Géczy, P., Izumi, N., & Hasida, K. (2012). Cloudsourcing: Managing cloud adoption. *Global Journal of Business Research (GJBR)*, *6*(2).

Glaser, J. (2011). Cloud computing can simplify HIT infrastructure management. *Healthcare Financial Management: Journal of the Healthcare Financial Management Association, 65*(8), 52.

Goodburn, M. A., & Hill, S. (2010). The Cloud Transforms Business Cloud is not the future—It's now. It's an enabling force for business evolution that goes beyond technology, allowing organizations to more efficiently and effectively reengineer the company's corporate strategy. For transformative power, the new model is blue—As in "sky.". *Financial Executive*, *26*(10), 34.

Harry Katzan, J. (2010). On an ontological view of cloud computing. *Journal of Service Science, 3*(1).

Iyer, B., & Henderson, J. C. (2010). Preparing for the future: Understanding the seven capabilities of cloud computing. *MIS Quarterly Executive, 9*(2), 117–131.

Iyer, B., & Henderson, J. C. (2012). Business value from clouds: Learning from users. *MIS Quarterly Executive, 11*(1), 51–60.

Jamil, E. (2009). What really is SOA. A comparison with cloud computing, Web 2.0, SaaS, WOA, Web Services, PaaS and others. *White Paper, Soalib Incorporated*.

Kerin, R. A., Varadarajan, P. R., & Peterson, R. A. (1992). First-mover advantage: A synthesis, conceptual framework, and research propositions. *Journal of marketing, 56*(4), 33–52.

Kushida, K. E., Murray, J., & Zysman, J. (2011). Diffusing the cloud: Cloud computing and implications for public policy. *Journal of Industry, Competition and Trade, 11*(3), 209–237.

Lacity, M. C., & Willcocks, L. P. (2013). Outsourcing business processes for innovation. *MIT Sloan Management Review, 54*(3), 63.

Lacity, M. C., & Willcocks, L. P. (2016). A new approach to automating services. *MIT Sloan Management Review, 58*(1), 41.

Leavitt, N. (2009). Is cloud computing really ready for prime time? *Computer, 42*(1), 15–20.

Lee, R. P., & Chen, Q. (2009). The immediate impact of new product introductions on stock price: The role of firm resources and size. *Journal of Product Innovation Management, 26*(1), 97–107.

Levinthal, D. A., & March, J. G. (1993). The myopia of learning. *Strategic Management Journal, 14*(S2), 95–112.

Low, C., Chen, Y., & Wu, M. (2011). Understanding the determinants of cloud computing adoption. *Industrial management & data systems, 111*(7), 1006–1023.

March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization science, 2*(1), 71–87.

Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems, 51*(1), 176–189.

Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

Miller, D., & Chen, M.-J. (1994). Sources and consequences of competitive inertia: A study of the US airline industry. *Administrative Science Quarterly*, 1–23.

Mircea, M., Ghilic-Micu, B., & Stoica, M. (2011). Combining business intelligence with cloud computing to delivery agility in actual economy. *Journal of Economic Computation and Economic Cybernetics Studies, 45*(1), 39–54.

Misra, S. C., & Mondal, A. (2011). Identification of a company's suitability for the adoption of cloud computing and modelling its corresponding return on investment. *Mathematical and Computer Modelling, 53*(3), 504–521.

Oliveira, T., Thomas, M., & Espadanal, M. (2014). Assessing the determinants of cloud computing adoption: An analysis of the manufacturing and services sectors. *Information & Management, 51*(5), 497–510.

Owens, D. (2010). Securing elasticity in the cloud. *Communications of the ACM, 53*(6), 46–51.

Poslad, S. (2011). *Ubiquitous computing: smart devices, environments and interactions*, Wiley.

Rai, A., Pavlou, P. A., Im, G., & Du, S. (2012). Interfirm IT capability profiles and communications for cocreating relational value: Evidence from the logistics industry. *MIS Quarterly, 36*(1), 233–262.

Sako, M. (2012). Business models for strategy and innovation. *Communications of the ACM, 55*(7).

Sambamurthy, V., Bharadwaj, A., & Grover, V. (2003). Shaping agility through digital options: Reconceptualizing the role of information technology in contemporary firms. *MIS Quarterly*, 237–263.

Shane, S., & Venkataraman, S. (2000). The promise of entrepreneurship as a field of research. *Academy of Management Review, 25*(1), 217–226.

Steininger, D. M. (2019). Linking information systems and entrepreneurship: A review and agenda for IT-associated and digital entrepreneurship research. *Information Systems Journal, 29*(2), 363–407.

Subramani, M. (2004). How do suppliers benefit from information technology use in supply chain relationships? *MIS Quarterly*, 45–73.

Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: Towards a cloud definition. *ACM SIGCOMM Computer Communication Review, 39*(1), 50–55.

Yoo, C. S. (2011). Cloud computing: Architectural and policy implications. *Review of Industrial Organization, 38*(4), 405–421.

Youseff, L., Butrico, M., & Silva, D. D. (2008). Toward a unified ontology of cloud computing. In *Paper presented at the Grid Computing Environments Workshop, 2008. GCE '08.*

Zahra, S. A., Nielsen, A. P., & Bogner, W. C. (1999). Corporate entrepreneurship, knowledge, and competence development. *Entrepreneurship: Theory and Practice, 23*(3), 169–169.

Analyzing Usage Data in Enterprise Cloud Software: An Action Design Research Approach

Philipp Hoffmann, Kai Spohrer, and Armin Heinzl

Abstract The shift from on-premise to cloud enterprise software has fundamentally changed the interactions between software vendors and users. Since enterprise software users are now working directly on an infrastructure that is provided or monitored by the software vendor, enterprise cloud software providers are technically able to measure nearly every interaction of each individual user with their cloud products. The novel insights into actual usage that can thereby be gained provide an opportunity for requirements engineering to improve and effectively extend enterprise cloud products while they are being used. Even though academic literature has been proposing ideas and conceptualizations of leveraging usage data in requirements engineering for nearly a decade, there are no functioning prototypes that implement such ideas. Drawing on an exploratory case study at one of the world's leading cloud software vendors, we conceptualize an Action Design Research project that fills this gap. The project aims to establish a software prototype that supports requirements engineering activities to incrementally improve enterprise cloud software in the post-delivery phase based on actual usage data.

1 Introduction

The shift from on-premise to cloud enterprise software products has fundamentally changed the interactions between vendors and users. When on-premise software products are developed, there is rarely a direct interaction between the enterprise software vendor and the users of the software product. Independent of the respective systems sourcing mode, the software product is deployed in the client organization and maintained under the rule of the client's IT department. Thus, the client's IT

P. Hoffmann (✉) · K. Spohrer · A. Heinzl
University of Mannheim, Mannheim, Germany
e-mail: hoffmann@uni-mannheim.de

K. Spohrer
e-mail: spohrer@uni-mannheim.de

A. Heinzl
e-mail: heinzl@uni-mannheim.de

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_11

department acts as a mediator between the enterprise software vendor and its software product users so that users do not directly interact with the software vendor. This has led to a persistent disconnect between enterprise software vendors and the actual users of their software on the client side, the so-called vendor-user gap.

During the software development phase, enterprise software vendors try to bridge this gap by applying classical methods of requirement engineering (RE), such as focus groups, lead users, and workshops (Liu et al. 2017; Maalej et al. 2016; Olsson and Bosch 2015). Although these methods aim to transfer requirements from both client IT departments and respective software users, they rely only on the attitudes and needs of few selected users. Consequently, this approach has led to an imperfect solicitation of requirements, even if RE methods were applied rigorously. Not meeting the requirements of users is still one of the key risks in enterprise software development (Mathiassen et al. 2007).

With the shift from on-premise to on-demand or cloud software, two fundamental changes have taken place: First, relying on agile development practices, enterprise software vendors established shorter development cycles (Hoda et al. 2013) and most enterprise cloud software products are now updated more frequently for all users at the same time. These changes have led to shorter innovation cycles with incremental updates of cloud products that often consist only of a few innovative features (Choudhary 2007). In contrast to on-premise solutions, enterprise cloud products usually start with basic core functionalities and evolve with additional features while users are already using the product. This has shifted much product development time from the pre-delivery to the post-delivery phase. Second, the software users are now working directly on an infrastructure that is provided or monitored by the software vendor. Consequently, enterprise cloud software providers are technically able to measure nearly every interaction of each individual user with their cloud products. This offers unprecedented insights into the actual usage of their software products.

The availability of new data and the emergence of novel insights into actual usage may provide an opportunity to better understand how to improve and extend enterprise cloud products while they are being used. In connection with new forms of feedback data (e.g. social media posts, app store reviews, usage polls etc.), prior work on requirements engineering (RE) has referred to the integration of the massive amounts of newly available user input into RE as *data-driven RE* (e.g. Maalej et al. 2016). Yet, it has remained unclear how the novel sources of usage data can be incorporated into a continuous development process of enterprise cloud products and how they are able to support enterprise software vendors in providing products that better fit their users' needs. Considering the characteristics of usage data, data-driven RE constitutes a big data analytics problem and scholars envision that analytics are likely to constitute a performance enhancer (Grover et al. 2018) by leading to improved requirements that eventually translate into better enterprise software products. Conceptualizations of such ideas or even technological solutions in enterprise cloud software development have, however, remained fairly unexplored. Thus, the time is ready to explore how software artifacts can utilize the continuous availability of usage data of enterprise cloud software users in order to improve RE in enterprise cloud software development (ECSD).

Our paper develops a conceptualization of an Action Design Research (ADR) project (Sein et al. 2011) that aims to establish a software prototype that supports RE activities on usage data in the post-delivery phase of enterprise cloud software. Our concepts are based on an explorative case study at an enterprise cloud software vendor. In the words of Henderson and Clark (1990), we foster incremental innovation regarding software products: By focusing on usage data in enterprise software development for the incremental improvement of the software, we reinforce the software's core concepts and keep the linkages between them unchanged (Henderson and Clark 1990).

Therefore, we focus on the following research question:

> RQ: How should an ADR-based software artifact be designed to make use of a continuous analysis of enterprise cloud software usage to improve requirements engineering (RE) in enterprise cloud software development (ECSD)?

Our manuscript is organized as follows: We first delineate data-driven RE from classical RE with regard to previous research and provide a literature review of the few extant approaches to leverage usage data for RE. We report insights from an initial empirical investigation of the problem based on an exploratory case study at a large enterprise cloud software vendor. Our first findings indicate how usage data are currently leveraged in a piecemeal fashion to support single steps in requirements elicitation techniques by Mathiassen et al. (2007), namely, requirements discovery, requirements prioritization, requirements experimentation, and requirements specification. Although little support for any of these activities could be found, we use our initial findings to conceptualize our ADR project following Sein et al. (2011) to develop and evaluate a software artefact that improves ECSD leveraging usage data and contemporary analytical techniques. At the end of our paper, we outline a number of specific further research avenues based on gathered usage data.

2 Data-Driven Versus Traditional Requirements Engineering

Academic literature often views RE from a normative and deterministic perspective (e.g. Davis and Hickey 2002; Sommerville 2007) describing RE as a "staged sequence of activities and/or task objectives" (Chakraborty et al. 2010). From a non-normative perspective, Pohl (2010) outlines a RE framework encompassing the core activities of eliciting, negotiating, and documenting desired properties and necessary constraints of a software, as well as the frequent validation and continuous management of the same. Compared to traditional RE, where the activity of requirements elicitation is mostly conducted with a few inquiry techniques, such as interviews, workshops, focus groups, survey questionnaires, and field observations (Liu et al. 2017; Maalej et al. 2016; Olsson and Bosch 2015), data-driven RE extends the set of applied techniques by the extensive analysis of user activities. Although data-driven RE relies on the same high-level tasks as other approaches to RE, it emphasizes the user as the central stakeholder within a large and distributed user base. This is fundamentally different from traditional RE in enterprise software development where often only user representatives are involved in RE and the vast majority of users remains outside an enterprise software vendor's reach.

Only a few studies are referring directly to data-driven RE: Maalej et al. (2016) propose a practitioners' approach to data-driven RE distinguishing novel sources of explicit and implicit (big) data and the need of systematically analyzing and aggregating those. Q-Rapids, a data-driven approach to quality RE, was introduced and discussed in previous work (e.g. Franch et al. 2017) and Czarnecki (2018) describes data-driven RE in the specific context of cyber-physical systems.

We build on the definition of data-driven RE by Hoffmann et al. (2019). According to their conceptualization, user input is an umbrella term for any type of reflection of a user's behavior and experience with a software. While previous work identified different categories of novel sources of user input, this line of work has yielded a number of diverging definitions and terminologies to describe user input. Compared to previous research, we conceptualize user input in the requirements elicitation phase more holistically as **feedback data**, describing subjective user perceptions of a software reflected in the user's explicit articulations, and **usage data**, reflecting the users' behaviors and interactions with a software (Hoffmann et al. 2019). The enhanced user input in data-driven RE adds the following characteristics to classical RE (Hoffmann et al. 2019):

- **Continuity**: Requirements are continuously re-elicited and re-prioritized, also during the post-development phase, as opposed to a more occasional elicitation and prioritization in traditional RE. This emphasizes continuous software improvement and constant adaptation to user needs and preferences.
- **Automation**: For the increasing volumes of user input, it is no longer suitable to analyze the respective input data manually and to deduce requirements manually. Furthermore, since new user input becomes increasingly available through software, data-driven RE requires automated acquisition, processing, and smart interpretation of software requirements.
- **Context-awareness**: The user input is no longer restricted to the explicit expression of users' needs and preferences. Those traditional sources of feedback can be enriched with information about the actual context of the usage of a software.

Form a technical perspective, feedback data analysis is mostly relying on text analytics and natural language processing techniques (e.g. Panichella et al. 2015; Iacob and Harrison 2013), whereas usage data analysis is often utilizing process mining approaches (e.g. Dąbrowski et al. 2017). We define data- driven RE by extending a traditional RE definition by Maalej et al. (2016) as follows (see Fig. 1): "Data-driven RE enriches the methods of collecting and analyzing user input in traditional RE with the automated and continuous analysis of novel feedback sources and with the analysis of context-aware usage data to identify, prioritize, document and manage requirements for a software product" (Hoffmann et al. 2019, p. 4).

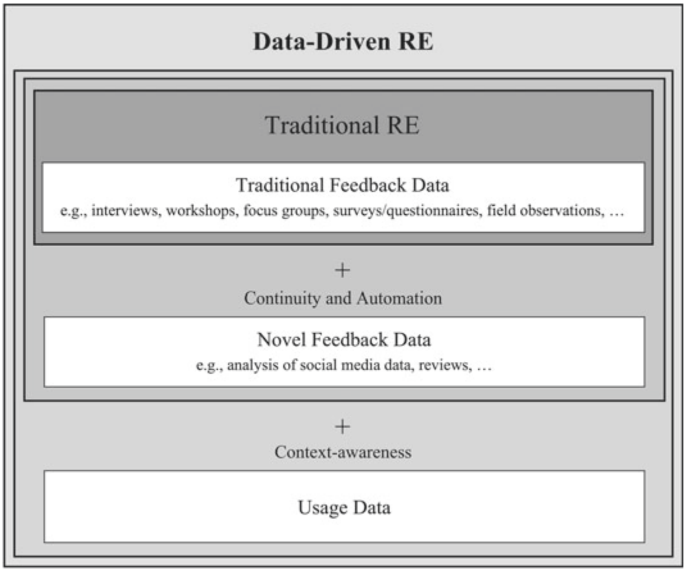

Fig. 1 Data-driven RE compared to traditional RE

3 Related Work in Data-Driven Requirements Engineering

Research on leveraging usage data for RE can be grouped broadly into three partially overlapping categories regarding the type of usage data they address: (1) Sensor data in cyber-physical systems (e.g. Czarnecki 2018), (2) system and hardware status data as well as quality and performance data of user clients (e.g. Stade et al. 2017) and (3) monitoring usage data on feature level (e.g. Oriol et al. 2018). In this paper, we focus on the third category and use a categorization regarding the intended use of the collected usage data on a software feature level. We follow Bosch's (2000) definition of a software feature as "a logical unit of behavior specified by a set of functional and non-functional requirements" (Bosch 2000, p. 194). For functional requirements elicitation and improvement in ECSD on a feature level, we focus on research addressing at least the third stage of the "Post-Deployment Data Usage Framework" by Olsson and Bosch (2013) as depicted in Fig. 2 and the two steps beyond, namely, "feature usage", "feature improvement" and "new feature development".

In 2013, Olsson and Bosch conducted a case study regarding the use of post-deployment data, i.e. software usage data, as a basis for product improvement and new product development in software-intensive embedded systems (Olsson and Bosch 2013). They derived the staged "Post-Deployment Data Usage Framework" depicted in Fig. 2. At the time the study was conducted, none of the participating companies leveraged post-deployment product data for improving the current software version or for innovating new functionalities. The data was only used for diagnostic purposes (i.e. troubleshooting and maintenance activities) as well as to monitor the operation

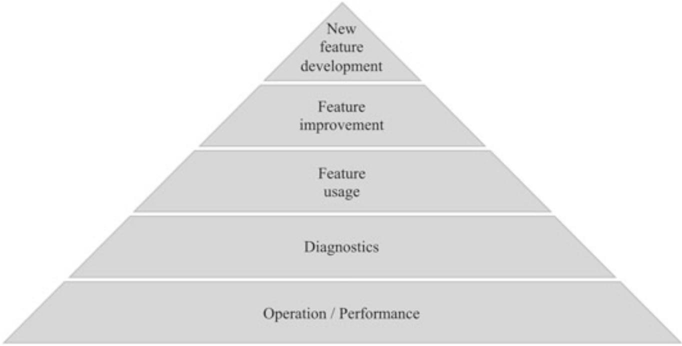

Fig. 2 Post-deployment data usage framework (Olsson and Bosch 2013)

and performance of the system. Fabijan et al. (2015) conducted a literature review on currently recognized techniques for collecting customer data and derived a categorization according to their characteristics in terms of the development stage where the technique is deployed as well as their challenges and limitations. Regarding usage data collection techniques, they identified “operational and event data” in the development phase as well as “A/B testing” in the post-development phase. They argue that utilizing those usage data collection techniques leads to designs and results that are subject to statistical assumptions. Furthermore, they sought to confirm hypotheses rather than explore stakeholder opinions (Fabijan et al. 2015).

Initial ideas of leveraging usage data for RE can be found in Robinson (2010) who introduced a comprehensive, four-layer service framework to focus on the verification and validation of requirements. The proposed two-component monitoring architecture is based on an event listener, recording the stream of usage events, and a requirements analyzer, reviewing the events to determine if the software meets the requirements defined by a predefined and evolving model of the software. Furthermore, Robinson (2010) derived consequential research questions regarding event-acquisition and event-interpretation in the proposed approach.

Bosch (2012) derived from a single case study in a Software as a Service (SaaS) company an “innovation experiment system” approach for software development. It is based on a loop between developing new software functionalities, the measurement of usage and other performance metrics, and the reintegration of the collected data and findings in the next iteration cycle. The loop is conducted in three phases of software development (i.e. pre-deployment, non-commercial deployment, and commercial deployment), and in three scopes for experimentation (i.e. optimization, new features, and new products). Through A/B testing of alternative feature implementations in the commercial deployment phase, usage data is leveraged in the development cycles.

Further research by Olsson and Bosch (2014) identified the “open loop problem” between customer feedback and product management decisions. It describes situations in which, due to inaccurate and non-timely feedback from customers, decisions are made based on opinions rather than data. The authors addressed the “hypothesis experiment data-driven development" (HYPEX) model aiming to support companies in running feature experiments to shorten the feedback loops. Based on the same problem statement, Olsson and Bosch (2015) developed a conceptual model of "qualitative/quantitative customer-driven development" (QCD) to combine feedback data from customers in early stages of development with customer usage data in later stages of development.

Bosch and Olsson (2016) introduced a conceptual model for evidence-based engineering of autonomous systems to validate any new development or change to the system from the perspective of the value that it delivers, including three levels of development feedback loops. In the first level of the model, called the "R&D loop", the software engineers can continuously validate and determine if the functionality of the software provides the intended and expected value to the customers or the company itself by including experimentation (e.g. A/B testing). The second ("dynamic system loop") and third level of feedback loop ("static system loop") are leading towards self-optimizing systems measuring the effectiveness of a development effort in relation to a predefined value function and shifting "development effort from the R&D teams to the system itself" (Bosch and Olsson 2016, p. 30). However, since already the first loop is not addressed by working prototypes in research at the moment, we delimit our ADR project also to this first loop to reduce the complexity of our project. Furthermore, because our research is embedded in ADR research methodology at an enterprise software vendor, we rely on the possibility to embed our software artefact in a working enterprise software environment and we see the continuous evolution of autonomous systems in ECSD as too far in the future for an ADR research approach at the moment.

In addition to the presented high-level conceptual models for leveraging usage data in data-driven RE, Dąbrowski et al. (2017) sketched an approach of using process mining for RE. Based on a hypothetical example they discuss a "goal-driven process mining" system approach to analyze usage data for confirming and validating hypotheses based on received feedback data. From system logs, the proposed system extracts user's processes to detect misalignments between designed and executed processes by examining whether assumptions about user preferences and the system design specifications satisfy the system goals. A first implementation of leveraging usage data for data-driven RE is presented by Oriol et al. (2018) introducing the "Feedback Acquisition and Monitoring Enabler" (FAME) framework for simultaneous collection of feedback data and usage data (click streams and navigation paths) to support requirements elicitation and prioritization. In stark contrast to earlier research, they implemented and validated their approach in a German software company. However, only feedback related usage data were analyzed, and the analysis was conducted manually by a company representative in a single workshop. Although it was useful to demonstrate the feasibility and the usefulness of leveraging usage data for RE, it seems to be necessary to develop automatic (or semi-automatic) means to analyze the usage data (Stade et al. 2017) since insights from usage data that were not related to corresponding feedback data were neglected in the approach.

In summary, the review of current research reveals that implementations or design guidelines of data-driven RE with regard to the top-three stages of "Post-Deployment

Data Usage Framework" by Olsson and Bosch (2013) (i.e. feature usage, feature improvement and new feature development) for RE are not in place for ECSD. Only context supporting solutions for requirements elicitation on feedback data could be found in Oriol et al. (2018). Furthermore, Stade et al. (2017) discovered in a case study very first experiences of a software company using run-time data derived from system logs, hardware logs as well as end-user interactions by using Google Analytics for data-driven RE. However, their case company was not able leverage the data for RE to identify and understand the needs of the users to evolve their software (Stade et al. 2017) based on those relatively easy to implement sources of data.

4 Methodology

It is our intention to utilize findings from a multiple case study to set up a normative concept for an ADR research approach at a participating company in order to create a testable IT artifact. In this environment, utilizing usage data in terms of a fully data-driven RE approach has not yet been covered and to our knowledge, no ADR research regarding leveraging usage data in RE has been conducted in ECSD. Our research can be deemed to be well suited for the ADR methodology by Sein et al. (2011) since we want to combine theory generation that contributes to the body of knowledge on data-driven RE with the solution of a relevant organizational problem (Baskerville and Wood-Harper 1998). In the following section, we describe the context of a large enterprise cloud software vendor that provides the basis for our study and give an overview of our case study design and ADR.

4.1 Participating Enterprise Cloud Software Vendor

In the last decade, the participating enterprise cloud software vendor started a transition from an on-premise enterprise software vendor towards a cloud software vendor. It is one of the world's largest players in the enterprise cloud software market and a suitable industry-representative in multiple ways due to the large software product portfolio ensuring the coverage of different types of software architecture, varying targeted industries, and diverse sizes and complexities of software products. We gained access to the development organization of the software vendor and were able to gather data in RE for various products to get a holistic overview over their RE activities.

4.2 Multiple Case Study

At the participating company, we conducted an explorative multiple case study to gather insights into the current and planned use of usage data for RE as well as to understand current RE activities and practices in enterprise cloud development following Yin (2018). Since no comprehensive understanding of those RE activities could be found in academic literature, our case study is explorative by nature. Within three sampled cases, we conducted seven semi-structured interviews and examined internal documentations, dashboards, and system reporting tools. The interviewees were selected at the interface between the customers and the development teams. Our interviews were conducted mostly face-to-face or otherwise via video conference and were scheduled for 60 min. After a short introduction, the interviewees were asked how they currently integrated usage data into their RE activities and what further projects of integrating usage data into RE they were planning. Like in other large development organizations working with agile development methods, two specific job roles conducted those RE activities: Product owners acted as the functional leads in each team, guiding the development effort via the management of the product backlog. In contrast, product managers were responsible for direct customer and user interactions in particular, capturing requirements as well as transferring them to the product owner of the corresponding development teams.

4.3 Action Design Research

Since its beginnings, information systems (IS) research had aimed to make theoretical contributions as well as to solve problems in practice (Benbasat and Zmud 1999). In this regard, design science aims to develop and evaluate innovative IT artifacts to create design knowledge for an identified class of problems (Hevner et al. 2004). The key proposition of ADR is that the IT artifact is at the core of IS research (Sein et al. 2011). However, compared to other design science approaches where evaluation takes places after the development (e.g. Vaishnavi et al. 2004), ADR follows the “technology as structure” view by Orlikowski and Iacono (2001) where the structures of the organizational domain are introduced into the development and use of the IT artifact so that ADR makes organizational intervention necessary. Therefore, the continuous evaluation of developed artifacts becomes the key focus or our research (Sein et al. 2011).

ADR research is based on four stages containing seven principles “that capture the underlying assumptions, beliefs, and values” (Sein et al. 2011, p. 40): (1) The **problem formulation** is based on an encountered problem in practice or on a problem anticipated by researchers. It conceptualizes a research opportunity based on existing academic literature. It is important to ensure the generalizability from the specific problem in the participating organization toward a broader class of problems. (2) **Building, intervention, and evaluation** is an iterative phase from creating the first IT artifact which is afterwards shaped by subsequent design cycles and the organizational use. The constant evaluation of the IT artifact should lead to a final design after this stage. Furthermore, the end of this stage indicates the "locus of innovation which may come from the artifact design or the organizational intervention" (Sein et al. 2011, p. 42). (3) The **reflection and learning** stage shifts the focus from the lessons learned of the IT artifact to solve a specific organizational problem towards a broader class of problems. It parallels the first and second stage and ensures that the project contributes to the body of academic knowledge (Sein et al. 2011). (4) The **formalization of learning** formulates the aforementioned contribution to the theories used by the initial stage characterized as design principles and refinements of those theories.

5 Results

In the following, we present our case study results from the participating company which provided the basis for our ADR research approach.

5.1 Case Study Findings

Our three cases at the participating company differ in the use of leveraging usage data for RE, their degree of context-aware collection of usage data, the continuity of the data collection and the degree of automation in terms of the collection and analysis of the usage data. Table 1 presents an overview of the three cases at the participating software vendor regarding the use of software usage data in RE (cf. Hoffmann et al. 2019). Feature usage describes how often a software feature is used by a user, measured by pre-defined API calls within the software code or specific data base activities. Measuring failure refers to the collection of error records, e.g. though response codes of an API, when users encounter an error using a feature in operation. Furthermore, we added for the sake of completeness also the category of latency or

Table 1 Cases with examples of leveraging usage data in RE at the software vendor

		Product A	**Product B**	**Product C**
	Number of customers	Few	Moderate	Many
	Point in time of initial release	Young	Young	Established
	Type of software	Application	Platform	Application
Usage data	Feature usage	Yes	Yes	Yes
	Measuring failure	No	Yes	Yes
	Latency or computation load	No	Yes	Yes

computation load even though it addresses lower stages of Olsson and Bosch's (2013) "Data Usage Framework". This category incorporates the measurement of statistics, e.g. regarding the hardware load of the cloud software or the speed of loading user interfaces.

We structured our detailed findings within the four-dimensional categorization of requirement development techniques by Mathiassen et al. (2007):

- **Requirements discovery** techniques are user-centric activities and facilitate the identification and prediction of emerging requirements (Mathiassen et al. 2007). In our cases, error data is analyzed to measure and track failures in executing features during user operations (cases B, C). Furthermore, we found one product unit experimenting with click streams to measure the time of using process-oriented software features (end-to-end workflows) and identify workarounds of users (case A).
- **Requirements prioritization** addresses resource-centric analyses and decision support to focus on the elaboration of the relevant requirements (Mathiassen et al. 2007). In all cases, feature usage is analyzed by single specific measurement API calls within the code or by analyzing database calls (cases A, B, C). It allows to derive statements about the importance of a feature in case of improvements or failures.
- **Requirements experimentation** techniques apply software-centric designs as a key means for communicating with and iteratively involving users (Mathiassen et al. 2007). Whereas previous literature focuses often on requirements experimentation (e.g. Bosch 2012), we found no evidence of leveraging usage data for A/B-Tests or canarying (e.g. Harman et al. 2013), i.e. for testing software features deployed to a limited group of given users compared to another group of users.
- **Requirements specification** describes the documentation-centric abstraction and textual/graphical representation of a requirement (Mathiassen et al. 2007). Within our cases, we could not find any implementation of data-driven requirements specification based on usage data.

In summary, our three cases at our participating company offer a variety of approaches to leverage usage data within the development. However, a holistic data-driven RE approach incorporating the context-aware, continuous and automated elicitation of requirements was not yet discovered. This calls for developing a novel concept of a usage-data fueled RE approach.

5.2 Normative Concept of an Action Design Research Approach to Data-Driven RE

According to Sein et al. (2011), ADR deals with two apparently incompatible challenges: "(1) Addressing a problem situation encountered by a specific organizational setting by intervening and evaluating; and (2) constructing and evaluating an IT artifact that addresses the class of problems typified by the encountered situation" (Sein et al. 2011, p. 40). We encountered the first challenge during the presented case study at the participating partner where an enterprise cloud software vendor tries to leverage the new opportunity to analyze in-depth cloud usage data. In the following, we want to utilize those findings to conceptualize a normative ADR research approach to address the second challenge of constructing and evaluating an IT artefact that can support RE.

Stage 1: Problem Formulation

We encountered the problem of the vendor-user gap during the presented empirical investigation at the participating enterprise software vendor and discovered its first approaches to overcome this gap by leveraging usage data from enterprise cloud products for RE.

To the knowledge of the authors, there is no holistic approach that concurrently addresses feature usage, feature improvement and new feature development (Olsson and Bosch 2013) as well as the definitional criteria for data-driven RE, neither in industry nor in the literature. Therefore, our initial scope is to implement a working prototype in one component of an enterprise cloud software product. Within that component, we aim to implement various measure points in a few software functions to continuously monitor the users' actions within the software that are then used to derive requirements.

Although our software artefact is developed and evaluated in one specific product, we assume that the diverse organizational contexts can help elevate the generalizability of our findings. To ensure an adequate context of our research at the software vendor, we reviewed company internal documentations of the methodologies and processes of RE and interviewed the central process manager for RE. We were able to understand the specific problems of the organization with the new evolving usage data and initiated a joint project to implement and evaluate a solution.

The problem of leveraging cloud software usage data to overcome the vendor-user gap at the participating company is an instance of a class of problems in ECSD. Since no working prototype for usage data addressing feature usage, feature improvement and new feature development (Olsson and Bosch 2013) for requirements elicitation could be found, we conclude that post-deployment usage data remains an untapped resource for RE in most companies.

Stage 2: Building, Intervention and Evaluation

In the second stage of ADR, the initial design of the IT artifact is generated which is then, in an iterative process, "shaped by organizational use and subsequent design cycles" (Sein et al. 2011, p. 41).

In a first step, we realize a set of fully measurable functions (in terms of usage data) within an enterprise cloud software at the participating software vendor. In this context, we have to specify the level of detail of the different measuring points within the software code, starting with a relatively granular implementation since previous work has shown that coarse granularity of the data might be an obstacle to derive meaningful insights from usage data (e.g. Stade et al. 2017). The measure points should contain, in a first step, the unique ID of the measure point, an anonymized session ID of the individual user as well as the current timestamp. During the iterations of the design cycles, we are going to evaluate how deep the measurability should be to balance the need for detail with the prevention of potentially excessive data storage.

In a second step, a matching of the usage data to the development data (e.g. user stories) of the corresponding software features has to be conducted in terms of which usage data corresponds to which parts of the development data. We aim for an automated matching.

The third step includes the analytical tasks, especially which (bigdata) analytics techniques should be applied to detect valuable insights. We aim to structure those RE insights in four categories, derived from literature: Oh et al. (2013) distinguish **buggy features**, i.e. features that are not working according to their specification ("functional bug") from **missing features**, where software users are expecting more functionality ("functional demand"). Furthermore, we differentiate **unused features** (e.g. Oriol et al. 2018), i.e. features that are implemented but not used by the users, from **misused features** where features are used differently than in their specification.

The last step addresses the aggregation of the data as well as their presentation to the different job roles. Since our research will be conducted in close interaction with product managers and the product owners of the chosen component, we expect different needs with regards to the job role (c.f. Bick et al. 2017). Furthermore, the two distinct views of those roles will help us to address all four requirements elicitation techniques (i.e. requirements discovery, requirements experimentation, requirements prioritization, requirements specification) since the different roles are focusing on different RE techniques. We aim to categorize the usage data corresponding to software features and classify them from the requirements discovery perspective (c.f. Mathiassen et al. 2007) as buggy features, missing features, unused features or misused features. Moreover, we aim to leverage usage data for Requirements experimentation, requirements prioritization as well as requirements specification to address all four categories of requirements elicitation techniques (Mathiassen et al. 2007) and present a holistic automated, continuous and context-aware data-driven approach.

We plan to work in IT-dominant "Building, Intervention and Evaluation"- design cycles since a more mature artifact is needed for testing of the artifact in an ECSD organization. During the development of the prototype, we aim to evaluate our IT artifact continuously with the corresponding product managers and product owners, as well as with the software engineers in the development teams in terms of new valuable insights for requirements elicitation.

Stage 3: Reflection and Learning

In this stage, the artifact moves from the stage of just solving a specific instance of a practical problem towards a "broader class of problems" (Sein et al. 2011). Academic literature has been proposing ideas and conceptualizations for leveraging usage data in RE for nearly a decade. Against this backdrop, we expect that a first, research-driven prototype implementation that aims to address feature usage, feature improvement and new feature development greatly contributes to the academic knowledge.

Stage 4: Formalization of Learning

Our objective is to generalize our findings. We aim to derive design principles of how much data is needed and to what level of detail usage data has to be measured to leverage RE processes. Furthermore, we plan to find a good automated presentation of the data and findings to the product owners and product managers. This includes analytical considerations of the patterns to find in the usage data to derive RE implications.

6 Contributions

The contributions of this manuscript are twofold: On one hand, we present first empirical insights of leveraging usage data in RE within one of the world's leading enterprise cloud software vendors. On the other hand, we provide a high-level conceptualization of a normative ADR approach to develop a holistic data-driven IT artifact that allows for generalization to contribute to the academic knowledge base.

Our empirical insights indicate that first approaches of utilizing usage data in RE are already in place in industry. We only found approaches for requirements discovery and prioritization, whereas we found only little application of data-driven RE techniques for requirements experimentation and specification. Although prior research mainly focused on requirements experimentation (e.g. Bosch 2012), we could not empirically find evidence of leveraging usage data for A/B-Tests or canarying (e.g. Harman et al. 2013) within ECSD.

We offer a normative conceptualization of an ADR approach to derive a software artifact that attempts to utilize usage data for data-driven RE in ECSD. Although not in the context of ECSD, previous research has already identified the need to leverage usage data in RE for cloud software as a whole. However, no working prototypes that correspond to the data-driven RE definition could be found in the literature so far and existing approaches focus mainly on requirements experimentation (e.g. Olsson and Bosch 2014) or pure conceptual ideas (e.g. Dąbrowski et al. 2017). Furthermore, an ADR approach to derive a holistic data-driven prototype has not yet been covered by academic literature. Following the ADR methodology, we are able to derive generalizations from the specific solution of a problem at the participating organization towards a larger class of problems in RE beyond a simple prototype. Therefore, we expect two further theoretical contributions: First, we aim to derive design principles for a software artifact how to discover, prioritize, experiment and specify requirements from usage data addressing feature usage, feature improvement and new feature development (Olsson and Bosch 2013) in RE. We strive to elaborate principles for how to measure feature usage at the right level of granularity, how to select analytical methods as well as how to derive usage data patterns to detect missing, misused, unused, and buggy software features. Second, we aspire to discover how the additional information provided by usage data in RE has to be presented to different job roles within the software development process as well as if the presentation has to differ regarding the type of the requirement. We expect an organizational change of the process of RE in ECSD when shifting towards a continuous, automated, and more fact-based RE by focusing on the actual user of the software.

Based on our initial research, we perceive a large practical impact in providing a tool to address the user-vendor gap in the enterprise software development. Since direct user interaction has always been a challenge in ECSD, we expect that an automated, continuous and context-aware approach to leverage usage data in RE can lead to more user-centric enterprise software. The novel approach is likely to address RE-related problems earlier and more precisely. Furthermore, it has the potential to prioritize requirements better according to fact based decisions rather than intuitions and speculations.

7 Limitations

Our project is subject to certain limitations: First and foremost, our research is currently in progress and, therefore, the instantiation of the IT artifact is not yet available. So far, we can only present a normative ADR conceptualization of how we want to implement and evaluate an IT artifact at the participating company.

Second, since our presented case study acted as a starting point for our ADR project, it was only conducted at a single company. A broader case basis derived from different companies would have increased the validity of our empirical findings. However, due to a variety of acquisitions of other software vendors during the last years, our case company offers already a wide range of different cases.

Third, we focus on usage data and do not integrate feedback data in this research project. Previous research highlights the value of combining feedback data with usage data (e.g. Oriol et al. 2018). However, we concluded during our case study at the participating software vendor that we have to reduce the complexity of our research project in a first step. Nonetheless, only the additional integration of feedback data will make our software artifact fully consistent with the data-driven RE approach in the future.

Fourth, we have to ensure legally compliant handling of gathered usage data following the General Data Protection Regulation. It may be necessary to aggregate data to address privacy regulations whereas a more detailed view on the data could provide deeper insights for RE.

8 Conclusion and Further Research Opportunities

In summary, based on our literature review, previous research on data-driven RE focused most on motivating ideas and designing specific prototypes regarding the idea of data-driven RE only on a conceptional level. We presented first empirical insights of which approaches of leveraging usage data for RE within ECSD are already being used in one of the world's leading enterprise cloud software vendors.

Moreover, our research project presents a first normative ADR project to develop a software artefact to leverage usage data for RE in ECSD in line with the definition of data-driven RE. We aim to categorize the usage data corresponding to software features and classify them from the requirements discovery perspective (c.f. Mathiassen et al. 2007) as buggy features, missing features, unused features and misused features. Furthermore, we also plan to integrate requirements experimentation, requirements prioritization as well as requirements specification (Mathiassen et al. 2007) into our approach towards an automated, continuous and context-aware data-driven approach.

With the gathered usage data, we see further research opportunities: Research can derive learning curves of new users in enterprise cloud software from the data by measuring the growing "feature vocabulary" of the users as well as how fast the processing times through different features decrease. Research can address the question which kinds of feature and user interface designs have an influence on the complexity of a software as well as how different kinds of onboarding trainings for new user affect the learning curves in different ways.

References

Baskerville, R. L., & Wood-Harper, A. T. (1998). Diversity in information systems action research methods. *European Journal of Information Systems, 7*(2), 90–107. https://doi.org/10.1057/palgrave.ejis.3000298.

Benbasat, I., & Zmud, R. W. (1999). Empirical research in information systems: The practice of relevance. *MIS Quarterly, 23*(1), 3–16. https://doi.org/10.2307/249403.

Bick, S., Spohrer, K., Hoda, R., Scheerer, A., & Heinzl, A. (2017). Coordination challenges in large-scale software development: A case study of planning misalignment in hybrid settings. *IEEE Transactions on Software Engineering, 44*(10), 932–950. https://doi.org/10.1109/TSE.2017.2730870.

Bosch, J. (2000). *Design and use of software architectures: Adopting and evolving a product-line approach*. Pearson Education.

Bosch, J. (2012). Building products as innovation experiment systems. In *International Conference of Software Business* (pp. 27–39), Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-30746-1_3.

Bosch, J., & Olsson, H. H. (2016). Data-driven continuous evolution of smart systems. In *2016 IEEE/ACM 11th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)* (pp. 28–34). https://doi.org/10.1109/seams.2016.011.

Chakraborty, S., Sarker, S., & Sarker, S. (2010). An exploration into the process of requirements elicitation: A grounded approach. *Journal of the association for information systems* (11:4).

Choudhary, V. (2007). Software as a service: Implications for investment in software development. In *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)* (pp. 209a–209a), IEEE. https://doi.org/10.1109/hicss.2007.493.

Czarnecki, K. (2018). Requirements engineering in the age of societal-scale cyber-physical systems: The case of automated driving. In *2018 IEEE 26th International Requirements Engineering Conference (RE)* (pp. 3–4), IEEE. https://doi.org/10.1109/re.2018.00-57.

Dąbrowski, J., Kifetew, F. M., Muñante, D., Letier, E., Siena, A., & Susi, A. (2017). Discovering requirements through goal-driven process mining. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)* (pp. 199–203). https://doi.org/10.1109/rew.2017.61.

Davis, A. M., & Hickey, A. M. (2002). Requirements researchers: Do we practice what we preach? *Requirements Engineering, 7*(2), 107–111. https://doi.org/10.1007/s007660200007.

Fabijan, A., Olsson, H. H., & Bosch, J. (2015). Customer Feedback and Data Collection Techniques in Software R&D: A Literature Review. In J. M. Fernandes, R. J. Machado, & K. Wnuk (Eds.), *Software business* (No. 210, pp. 139–153).

Franch, X., Ayala, C., López, L., Martínez-Fernández, S., Rodríguez, P., Gómez, C., & Jedlitschka, A., et al. (2017). Data-driven requirements engineering in agile projects: The Q-rapids approach. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)* (pp. 411–414). https://doi.org/10.1109/rew.2017.85.

Grover, V., Chiang, R. H. L., Liang, T.-P., & Zhang, D. (2018). Creating strategic business value from big data analytics: A research framework. *Journal of Management Information Systems, 35*(2), 388–423. https://doi.org/10.1080/07421222.2018.1451951.

Harman, M., Lakhotia, K., Singer, J., White, D. R., & Yoo, S. (2013). Cloud engineering is search based software engineering too. *Journal of Systems and Software, 86*(9), 2225–2241. https://doi.org/10.1016/j.jss.2012.10.027.

Henderson, R. M., & Clark, K. B. (1990). Architectural Innovation: The reconfiguration of existing product technologies and the failure of established firms. *Administrative Science Quarterly, 35*(1), 9–30. https://doi.org/10.2307/2393549.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly, 28*(1), 75–105. https://doi.org/10.2307/25148625.

Hoffmann, P., Mateja, D., Spohrer, K., & Heinzl, A. (2019). Bridging the vendor-user gap in enterprise cloud software development through data-driven requirements engineering. In *Proceedings of the 40th International Conference on Information Systems (ICIS)*, Munich.

Hoda, R., Noble, J., & Marshall, S. (2013). Self-organizing roles on agile software development teams. *IEEE Transactions on Software Engineering, 39*(3), 422–444. https://doi.org/10.1109/TSE.2012.30.

Iacob, C., & Harrison, R. (2013). Retrieving and analyzing mobile apps feature requests from online reviews. In *Proceedings of the 10th Working Conference on Mining Software Repositories* (pp. 41–44), IEEE.

Liu, L., Zhou, Q., Liu, J., & Cao, Z. (2017). Requirements cybernetics: Elicitation based on user behavioral data. *Journal of Systems and Software, 124,* 187–194. https://doi.org/10.1016/j.jss.2015.12.030.

Maalej, W., Nayebi, M., Johann, T., & Ruhe, G. (2016). Toward data-driven requirements engineering. *IEEE Software, 33*(1), 48–54. https://doi.org/10.1109/MS.2015.153.

Mathiassen, L., Rossi, M., Saarinen, T., & Tuunanen, T. (2007). A contingency model for requirements development. *Journal of the Association for Information Systems, 8*(11), 569–597.

Oh, J., Kim, D., Lee, U., Lee, J. G., & Song, J. (2013). Facilitating developer-user interactions with mobile app review Digests. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems* (pp. 1809–1814), ACM. https://doi.org/10.1145/2468356.2468681.

Olsson, H. H., & Bosch, J. (2013). Towards data-driven product development: A multiple case study on post-deployment data usage in software-intensive embedded systems. In *International Conference on Lean Enterprise Software and Systems* (pp. 152–164), Springer, Berlin, Heidelberg.

Olsson, H. H., & Bosch, J. (2014). From opinions to data-driven Software R&D: A multi-case study on how to close the 'Open Loop' problem. In *2014 40th EUROMICRO Conference on Software Engineering and Advanced Applications* (pp. 9–16), IEEE. https://doi.org/10.1109/seaa.2014.75.

Olsson, H. H., & Bosch, J. (2015). Towards continuous customer validation: A conceptual model for combining qualitative customer feedback with quantitative customer observation. In J. M. Fernandes, R. J. Machado, & K. Wnuk (Eds.), *Software business* (No. 210, pp. 154–166).

Oriol, M., Stade, M., Fotrousi, F., Nadal, S., Varga, J., Seyff, N., & Abello, A., et al. (2018). FAME: Supporting continuous requirements elicitation by combining user feedback and monitoring. In *2018 IEEE 26th International Requirements Engineering Conference (RE)* (pp. 217–227), IEEE. https://doi.org/10.1109/re.2018.00030.

Orlikowski, W. J., & Iacono, C. S. (2001). Research commentary: Desperately seeking the 'IT' in IT research—A call to theorizing the IT artifact. *Information Systems Research, 12*(2), 121–134. https://doi.org/10.1287/isre.12.2.121.9700.

Panichella, S., Di Sorbo, A., Guzman, E., Visaggio, C. A., Canfora, G., & Gall, H. C. (2015). How can I improve my App? Classifying user reviews for software maintenance and evolution. In *2015 IEEE International Conference on Software Maintenance and Evolution* (pp. 281–290). https://doi.org/10.1109/icsm.2015.7332474.

Pohl, K. (2010). *Requirements engineering: fundamentals, principles, and techniques*, 1st edn. Springer Publishing Company.

Robinson, W. (2010). A roadmap for comprehensive requirements modeling. *IEEE Computer, 43*(5), 64–72. https://doi.org/10.1109/MC.2009.373.

Sein, M., Henfridsson, O., Purao, S., Rossi, M., & Lindgren, R. (2011). Action design research. *Management Information Systems Quarterly, 35*(1), 37–56. https://doi.org/10.2307/23043488.

Sommerville, I. (2007). *Software engineering*, 8th edn. Addison Wesley.

Stade, M., Oriol, M., Cabrera, O., Fotrousi, F., Schaniel, R., Seyff, N., & Schmidt, O. (2017). Providing a user forum is not enough: First experiences of a software company with CrowdRE. In *IEEE 25th International Requirements Engineering Conference Workshops (REW)* (pp. 164–169). https://doi.org/10.1109/rew.2017.21.

Vaishnavi, V., Kuechler, W., & Petter, S. (Eds.). (2004). Design science research in information systems. Created in 2004 and updated until 2015 by Vaishnavi, V. and Kuechler, W.; last updated by Vaishnavi, V. and Petter, S. 2019. http://www.desrist.org/design-research-in-information-systems/.

Yin, R. (2018). *Case study research and applications*, 6th edn, Sage.

Balancing Risks and Opportunities in Cloud-Based Outsourcing

How and Why Software Outsourcing Projects Drift—An Actor-Network-Theoretic Investigation of Control Processes

Thomas L. Huber, Jens Dibbern, and Thomas Fischer

Abstract This study seeks to explain the perplexing phenomenon that many software outsourcing projects drift, i.e., they enter into a creeping process of targeting emergent goals often at the expense of losing sight of initial goals. Such drift is difficult to reconcile with the traditional logic of control found in the literature. According to this logic, clients should be able to ensure goal achievement through close monitoring. If drift occurs despite rigid control, this suggests that within the control process forces are at work that divert controls from their initial objectives. To better understand these forces in the control process and how they relate to drift, we contrast the logic of control with concepts and assumptions from actor-network theory (ANT). ANT allows us to understand the process of designing, enacting, and adapting controls as one of creating and changing actor-networks. Our longitudinal case study of four software outsourcing projects reveals that drift processes differ depending on three interconnected changes in the actor-networks, i.e., changes in who partakes in the (re-) negotiation of control mechanisms, what specific control mechanisms are (re-) defined, and how they are inscribed in the software artifact and the software task.

1 Introduction

In recent years many organizations have replaced their in-house information systems (IS) with solutions and services provided by external vendors (Lacity et al. 2009). While this trend towards software outsourcing is fueled by the promise of saving costs and enhancing quality, software outsourcing projects often drift, i.e., they enter into a creeping process of "deviating from their planned purpose for a variety of

T. L. Huber (✉) · J. Dibbern · T. Fischer
Paris, France
e-mail: huber@essec.edu

J. Dibbern
e-mail: jens.dibbern@iwi.unibe.ch

T. Fischer
e-mail: Thomas-A.Fischer@bekb.ch

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_12

reasons often outside anyone's influence" (Ciborra 2000, p. 4). Such drift, where software outsourcing projects take a life of their own, is difficult to reconcile with the traditional logic of control (LOC) in IS outsourcing research. According to this logic, clients select and enact control mechanisms that ensure goal achievement (Eisenhardt 1985; Ouchi 1979). In particular, clients will design contracts aimed at aligning vendor behaviors with their goals (Rustagi et al. 2008; Tiwana 2010; Tiwana and Keil 2009). If vendors still show undesirable behaviors, then controllers should be able to recognize such deviations, adapt controls accordingly, and bring the project back on track (Choudhury and Sabherwal 2003). This logic implies that in the face of formally defined and adapted controls, drift should not occur. And yet, software outsourcing projects often take on unpredictable paths where corrective actions are taken that move the project into new directions (Cullen et al. 2005). In this article, we seek to examine this mystery of drift by addressing the following research question: *How and why do software outsourcing projects drift, and how are differences in drift related to differences in the process of controlling such projects.*

As a first step, we unpack why the traditional logic of control struggles to explain drift. First, LOC's clear conceptual separation between controller (i.e. client) and controllee (i.e. vendor) (Eisenhardt 1985; Ouchi 1979) undermines the fact that in the context of software outsourcing, the definition and adaptation of control usually results from a process of negotiation between client and vendor (Huber et al. 2013), who may also enact control jointly (Gregory et al. 2013). Second, LOC conceives of controllers as rather rational actors that select and enact optimal control portfolios. This clashes with the fact that in software outsourcing projects client and vendor often lack the necessary knowledge for such informed control decisions and therefore control mechanisms are often incomplete (Cram et al. 2016; Wiener et al. 2016). Along the same lines, more recent research has found that control in software outsourcing corresponds more closely to a dynamic trial-and-error process than to the ideal of rational choice (Choudhury and Sabherwal 2003; Gregory et al. 2013; Huber et al. 2013; Remus and Wiener 2012). While the idea of trial-and-error change is in line with one key aspect of drift—i.e. control changes in unpredictable ways—research on the dynamics of control has not studied the other key aspect of drift. i.e., the relation to the complete or partial deviation from the planned purpose of a project.

To understand this relationship, we contrast the logic of control with actor-network theory (ANT)—a socio-technical lens focused on understanding the complex social processes related to technology development. ANT is useful to study drift because it places special emphasis on complex negotiation processes between human actors with limited rationality as well as the role on non-human actors such as contracts or artifacts that may take on agency (Ciborra 2000; Latour 1996; Tjornehoj and Mathiassen 2008). Contrasting LOC with ANT provides us with a dialectical theoretical basis that conceptualizes control as an open-ended process of creating and changing heterogeneous actor-networks. Guided by this conceptualization we conducted an exploratory, longitudinal multiple-case study of four software outsourcing projects. Our inductively derived findings provide evidence for drift processes to occur in each of the four projects, but in different forms. Specifically, projects either show unbalanced drift, where initial goals are given up in favor of emergent goals—often without explicit notice, or balanced drift where initial goals and emergent goals are reconciled. We show that such differences in drift depend on how actor-networks form along three interconnected changes of human and non-human actors, i.e., which stakeholders partake in the negotiation of control mechanisms, what specific control mechanisms they (re-) define, and what kind of changes they inscribe in the software artifact and the software task. These results contribute to the understanding of sources of control change, the role of controllers, and the relationship between control and performance. Next, we present our theoretical background, followed by methods, findings, and implications.

2 The Logic of Control in Research on Software Outsourcing

Control is defined as any attempt to affect behaviors to achieve goals (Cram et al. 2016; Wiener et al. 2016). Such attempts are typically categorized into different formal (e.g., behavior control, outcome control) and informal control modes (e.g., clan control) that are exercised via control mechanisms (Cram et al. 2016; Kirsch 2004; Wiener et al. 2016). The majority of prior research on control in software outsourcing projects adopts a view which we refer to as the logic of control. The first core assumption of this view is that managers act as controllers that plan and decide whereas subordinates act as controllees that follow the controller's lead (*actor asymmetry assumption*) (Ciborra 2000; Eisenhardt 1985). The second core assumption of LOC is that controllers make rational control choices (*rational choice assumption*) (Eisenhardt 1985). This implies that controllers can anticipate the consequences of their actions so that they will select control mechanisms that optimally align the behaviors of controllees with their own goals. Following such formally defined, enacted and adapted control mechanisms, software outsourcing projects should move towards reaching project goals. However, recent research (Cullen et al. 2005) has shown that even if corrective actions are taken to keep outsourced projects on track, projects may take on unpredictable paths moving towards emergent goals which can lead the project farther and farther away from initial goals. Such 'controlled' drift invites a new perspective on control that goes beyond the traditional LOC.

Research in the tradition of LOC has adopted concepts and assumptions that are likely too narrowly framed to explain drift. First, research on control in software outsourcing widely adopted LOC's actor asymmetry assumption by conceptualizing the client as controller and the vendor as controllee and by analyzing which formal and informal control mechanisms clients select to control vendors (Gopal and Gosain 2010; Rustagi et al. 2008; Tiwana 2010; Tiwana and Keil 2009). For example, prior research has analyzed under which environmental conditions, clients rely more heavily on formal control mechanisms such as contracts (Rustagi et al. 2008). However, recent research has cast doubt on the actor asymmetry assumption in software outsourcing because the boundaries between controller and controllee appear blurred in this context (Gregory et al. 2013). This work has shown that clients do not select and enact control unilaterally, instead, client and vendor select and enact control jointly (Gregory et al. 2013). Specifically, client and vendor jointly negotiate desirable project goals that are then used to track and steer goal achievement (Gregory et al. 2013; Huber et al. 2013). Moreover, the actors from client and vendor that participate in the selection and enactment of control are not necessarily confined to managers, instead, many more human actors such as developers and users may participate (Dibbern et al. 2004; Moody et al. 2016). Such complex negotiations between many actors have been identified as a potential source of drift (Ciborra 2000; Tjornehoj and Mathiassen 2008)—but the narrow actor asymmetry assumption conceals such complexities.

Second, research in the spirit of LOC widely adopted the rational choice assumption and accordingly predicted a positive relationship between ex ante control choice and performance. However, empirical support for LOC's prediction of a positive relationship between ex ante control choice and performance remained weak: While a few studies did confirm the predicted positive relationship (Gopal and Gosain 2010; Remus and Wiener 2012; Srivastava and Teo 2012), other studies did not (Kim et al. 2013; Tiwana and Keil 2009). In fact, two recent literature reviews concluded that evidence on the relationship between control and performance is inconclusive—particularly in software outsourcing (Wiener et al. 2016) and when it comes to the simultaneous achievement of multiple project goals (Cram et al. 2016). Acknowledging these difficulties of making optimal ex ante control choices, nascent research has shifted attention to the dynamics of control (Choudhury and Sabherwal 2003; Gregory et al. 2013; Huber et al. 2013). This research emphasizes that at the beginning of a project, those formulating control lack the required deep knowledge to define anything other than high-level goals (Huber et al. 2013). Because such an open formulation lacks details and conceals potential contradictions between different project goals, client and vendor jointly elaborate and advance control over time (Choudhury and Sabherwal 2003; Gregory et al. 2013; Huber et al. 2013; Remus and Wiener 2012). As a consequence, control corresponds more closely to a trial-and-error process than to the ideal image of optimal ex ante choice (Cram et al. 2016). Even though this research has softened the rational choice assumption by acknowledging that not all control mechanisms instantaneously work optimal, the untested assumption still appeared to be that controllers instantly recognize goal deviations (or contextual changes) and immediately adjust control so that drift should not be an issue.

Overall, the majority of prior research on controlling software outsourcing projects followed the basic tenets of LOC but the conceptual and theoretical assumptions of this logic appear too restrictive to fully account for the phenomenon of drift. Therefore, we next contrast LOC with ANT (Latour 1996)—a theoretical perspective that was suggested to be particularly suited to study drift (Ciborra 2000).

3 Enriching the Logic of Control with Actor-Network Theory

To better understand control and drift in modern technology creation processes, such as software outsourcing, Ciborra (2000) has pointed to the value of enriching traditional perspectives such as LOC with assumptions and concepts from ANT (Latour 1996). ANT is a socio-technical method that guides researchers in their endeavor to understand the complex processes related to the development of technologies (Latour 1999). ANT is useful to understand control and drift because it acknowledges that control mechanisms "are there and relevant" but that there are important social processes beyond the "means-ends chains" expressed in rational control (Ciborra 1999; Elbanna 2006, p.167). Specifically, ANT urges researchers to acknowledge that not only managers but a wide variety of actors equally shape projects (*actor symmetry assumption*) (Ciborra 2000; Latour 1996). Actors can be either human or non-human (Latour 1996). A *human actor* is any individual that acts (Latour 1996). In software outsourcing a wide variety of human actors from both the client and vendor side can be involved, including but not limited to managers, software experts, and users (Dibbern et al. 2004). In addition to acknowledging the influence of a wide variety of human actors, ANT points to the important influence of non-human actors (Latour 1996). *Non-human actors* are artifacts that serve as sources of action (Latour 1996). In the context of software development and outsourcing, such artifacts may include contracts, technologies (e.g., the software product), or tasks (e.g., programming) (Dibbern et al. 2004; Lacity et al. 2010; Sarker et al. 2006).

A key idea of ANT is that the different human and non-human actors do not influence behaviors in isolation but jointly as a network (Hanseth et al. 2004; Latour 1996). These networks that link together an act with all of its influencing factors are called *actor-networks* and the process of creating and changing them is called translation (Latour 1996; Law 1987). In translation processes actors reconfigure their actor-network, e.g. by adding or removing human and non-human actors, and by changing the nature of connections between them (Latour 1996; Law 1987). From this perspective, the above described observation, where in the context of software outsourcing, varying actors may participate in the selection and enactment of control, can be modelled as differences in the composition of actor-networks: Each human actor involved in selecting controls (such as managers, users, software experts) would be represented as a separate node. Likewise, the artifacts that these actors define to enact control (such as contracts or project plans) and that they create to comply with defined controls (such as software features or software modules) would also be represented as nodes that are added to the actor-network.

The key concept to understand how an actor-network shapes behaviors towards a certain direction is *inscription* (Latour 1996). Inscription refers to the way artifacts in the actor-network embody behavioral patterns (Hanseth and Monteiro 1997). Artifacts embody behavioral patterns because actors inscribe their interests into them (Hanseth and Monteiro 1997). For example, client and vendor inscribe their interests in the form of negotiated project goals into a contract and therefore the contract embodies behavioral patterns to achieve those goals. ANT holds that the more difficult an inscription can be reversed, the more likely future behaviors will unfold as programmed (Law 1987). For example, the interests that a manager inscribed into a project plan may be easier to reverse than those inscribed by a software designer into the software architecture.

Because a large variety of human and non-human actors can participate in these processes, ANT assumes that their outcome is non-deterministic in nature (*open-endedness assumption*) (Hanseth and Monteiro 1997; Latour 1996; Sarker et al. 2006). Applied to the context of software outsourcing, this means that projects may move into different directions depending on who participates in the process of negotiating contracts and what is being inscribed in the contract, the software artifact, and the task. Since such inscriptions can be irreversible, they can exert an obdurate influence on a project moving it either away from or towards its initial goals and either towards emergent goals and against emergent goals. If projects only move into one direction either towards emergent goals or initial goals (ignoring or losing sight of the other), we refer to this movement as 'unbalanced drift', if, however, they tend to reconcile initial and emergent goals, we refer to this as 'balanced drift' (Ciborra 2000).

Table 1 contrasts the traditional LOC perspective that has dominated research on managing software outsourcing with the ANT perspective that resonates with empirical complexities captured in more recent studies. This contrast serves as the dialectical theoretical basis of our study: We conceptualize recurrent changes in the design and enactment of control as processes of creating and changing actor-networks. Then, we analyze whether and how different ways of creating and changing actor-networks either encourage or discourage drift. Table 2 provides definitions and illustrative codes of the core concepts adopted in this study.

Table 1 Different perspectives on managing software outsourcing projects

Characteristic	The logic of control	This study	Actor-network theory
Relevant actor(s)	Only human actors—especially the controller	Relevant human actors (esp. from client and vendor) and non-human actors (esp. control mechanisms, software artifact, and task)	All conceivable human and non-human actors
Focus of analysis	Control mechanisms selected and enacted by the controller	Changes in the design and enactment of control conceptualized as changes in the network of human and non-human actors	Changes in the complete actor-network of human and non-human actors that shape an act

(continued)

Table 1 (continued)

Characteristic	The logic of control	This study	Actor-network theory
Main assumptions	Actor asymmetry: Controllers dominate controllees Rational choice: Controllers select and enact control mechanisms that optimally serve their interests	Key human and non-human actors from client and vendor shape projects equally The selected and enacted control mechanisms can move projects into varying directions	Actor symmetry: Different actors shape social processes equally Open-endedness: The outcome of social processes is non-deterministic
Classic references	Eisenhardt (1985), Kirsch (1996), Ouchi (1979)	n.a.	Latour (1996), Law (1987)
IS references	Goo et al. (2009), Gopal and Gosain (2010), Gopal and Koka (2012), Rustagi et al. (2008), Tiwana (2010), Tiwana and Keil (2009)	n.a.	Ciborra (2000), Hanseth et al. (2004), Hanseth and Monteiro (1997), Tjornehoj and Mathiassen (2008)
Criticism	Overemphasis of agency of controller	n.a.	Overemphasis of agency of non-human actors

Table 2 Key concepts adopted from actor-network theory

Concept	Description	Illustrative codes
Actant/ actor	An actant is someone or something that acts, to which activity is granted, or which is the source of action (Latour 1996)	• Human actors: Managers, developers, users, business analysts • Non-human actors: Control mechanisms, contracts, technologies, tasks
Actor-network	The network that links together an act with all of the human and non-human actants that influence the act (Latour 1996; Monteiro 2000)	• A network of contracts and project plans created by the client to steer the vendor • A network of legacy technologies and contractual extensions that creates a tendency for vendors to act in a distinct way
Control process	The open-ended process of creating, changing, and wielding actor-networks (Latour 1996; Law 1987)	• Managers enroll new user groups, then they managers and users jointly extend the contract, and then force the vendor to redesign the architecture of the system • Software experts extend the contract to force the vendor to design a highly flexible technological architecture and then demand additional customizations through frequent contract extensions

(continued)

Table 2 (continued)

Concept	Description	Illustrative codes
Inscription	The way artifacts in the actor-network embody behavioral patterns (Hanseth and Monteiro 1997; Law 1987)	• A contract defines mandatory goals • A rule defines desirable behaviors • The underlying architecture of the software makes adapting the system difficult
Drift	The unintended and gradual movement towards emergent directions or goals that seems to be driven by outside forces (Ciborra 2000)	• Unbalanced Drift: The project drifts away from cost goals in favor of quality goals • Balanced Drift: The project addresses emergent quality goals while also aiming at the initial goal of cost efficiency

4 Method

4.1 Research Approach and Case Selection

We chose a multiple-case study approach to explore the dynamics of control and drift. In line with canonical procedures of exploratory research, we selected cases on theoretical grounds (Eisenhardt 1989; Yin 2009). We chose four software outsourcing projects from a globally operating bank that we refer to as "Bank". Software outsourcing in the financial services industry is a particularly suitable context for our empirical exploration because it is a highly regulated industry in which client companies are forced to define legally-binding project goals right from the beginning of a project and these controls are usually refined over time (Gewald and Dibbern 2009).

The selected projects had—according to the responsible manager of Bank—varying degrees of success. This increased the likelihood of observing differences in deviations from initial goals. In order to observe changes in control and drift over time, all projects had to be ongoing, allowing us to gather data in several rounds to conduct a retrospective analysis of changes over a period of several years (Yin 2009). To ease comparability between cases, we sought to ensure that the selected cases all started with similar initial context conditions. The same business context, i.e. Human Resources (HR) in banking, was one element of homogeneity; others were project size, development methodology, and type of outsourced task, i.e. delivery of software products and associated services. Table 3 summarizes the selected projects.

4.2 Data Collection

We collected data retrospectively in two phases. Phase one lasted from September to October 2009 and phase two from July 2011 to January 2012. In both phases, we conducted interviews with the key individuals responsible for managing and

Table 3 Descriptive case information

	Candidate	Highpot	Alumni	Payslip
Similar case characteristics	Each selected case is *similar in size* (i.e., it ranges from 0.5 to 1.5 million US dollars *per year*), uses the *same software development* methodology (a slightly adapted waterfall model), and mainly consists *of software application services*			
Client	Big, globally acting German bank. All projects are from the HR department			
Vendor	The initial vendor Gamma (UK) was taken over by its competitor Delta (USA) during the project	Vendor Beta located in the USA	Vendor Epsilon located in the UK	Vendor Alpha located in the USA with a branch in India
Timeframe of investigation	2006–2011	2009–2011	2006–2011	2008–2011
Software system	The system supports screening and management of internal and external applicants for open positions at Bank	The system supports skill development and performance assessment of Bank's employees	The system supports the recruitment of graduates from universities for Bank	The system processes the payroll and self-managed reimbursements for Bank employees

executing the projects from both client and vendor, i.e. the client's senior managers (SM), project managers (PM), and application owners (AOs), and the vendor's key project manager (VM). This ensured insights from different perspectives allowing us to triangulate changes in the projects over time. Interviews were based on a semi-structured interview guideline, took between 45 and 90 min, were mostly conducted face-to-face, and were recorded and transcribed. To further triangulate and enrich our interview data, we did an extensive document analysis of contracts, operating level agreements (OLAs), service level agreements, project plans, project status reports, and audit documentation. Table 4 provides an overview of the collected data.

Retrospective data collection allowed us to capture relevant changes in control over several years while reducing the risk of data overload compared to continuous longitudinal data gathering (Poole et al. 2000). To alleviate the disadvantages of retrospective data collection, we asked interviewees questions about particular events instead of their general view on the project and we triangulated this information with evidence from documents. In case of discrepancies between subjective interview statements and objective documents, we contacted respondents for clarification. Only if information gleaned from one source (i.e., interview or document) was confirmed by another source it was included in our subsequent analysis (Kirsch 2004).

Table 4 Interviews and documents

	Interviews in Phase 1	Interviews in Phase 2	Total number of interviews	Main documents
Canidate	Client: 4 Vendor: 1	Client: 1 Vendor: 0	Client: 5 Vendor: 1 Total: 6	Contract incl. all four schedules, OLA, other project documentation
Highpot	Client: 2 Vendor: 1	Client: 6 Vendor: 0	Client: 8 Vendor: 1 Total: 9	Contract incl. all 13 schedules, OLA, other project documentation
Alumni	Client: 3 Vendor: 1	Client: 3 Vendor: 0	Client: 6 Vendor: 1 Total: 7	Contract incl. all schedules, OLA, other project documentation
Payslip	Client: 5 Vendor: 1	Client: 2 Vendor: 1	Client: 7 Vendor: 2 Total: 9	Contract incl. all eight schedules, OLA, other project documentation
Cross-project	Client: 2	Client: 3	Total: 5	OLA template, other project documentation
			Sum: 36	

4.3 Data Analysis

We followed an iterative approach to data analysis and theory building that involved moving back and forth between data and theory using open, axial, and selective coding (Charmaz 2006; Corbin and Strauss 1990). The coding was done using the software package NVivo 9. The concepts that emerged from our analysis were iteratively refined to become more robust and reliable (Charmaz 2006). Next, we describe the four stages of our analysis.

Stage 1—Understanding the case story. First, we used the interviews with higher-level managers to establish the high-level goals, as well as the context of each project. Then, we used interviews with project managers to understand the history of each project. Hence, at the end of this stage we had a basic understanding of the goals of each project, and the chronological flow of events, which we documented in a graphical timeline and in a first case write-up.

Stage 2—Narrowing down control changes and identifying episodes. The second stage of analysis was based on interviews with additional key informants from the project level who described the chronology of changes. In these interviews we asked further questions on how and why control changed over time. During this stage, both the documents and the graphical timeline for each project helped us to verify the consistency of these subjective accounts and to gather a richer picture of the case history (second write-up) (Miles and Huberman 1994). Each case story was then broken down into a number of control episodes to create distinct analytical units (Langley 1999). Each control episode consists of several temporally-ordered moments. The first moment is usually the negotiation of a new control mechanism, followed by several moments in which the new mechanism creates inscriptions in other non-human artifacts (i.e., software artifact and software task) with tangible consequences for outcomes. To ensure external validity of our understanding of these dynamics, we conducted a feedback presentation with Bank managers. The discrepancies discovered in these presentations were addressed by going back to the data and discussing intensively with key informants. Hence, at the end of this stage we had a verified understanding of the key dynamics of each case on the level of control episodes.

Stage 3—Concept refinement. In this stage we compared all episodes to refine our a priori concepts (see Table 2). First, we analyzed differences in the composition of the network of actors that negotiated control mechanisms. During this step the distinction between narrow and broad negotiation networks emerged. In narrow negotiation networks the variety of actors that represent distinct interests in the negotiation of control is small, whereas broad negotiation networks enroll additional actors with distinct interests that were previously marginalized. Second, we strived for a rich and differentiated picture of the nature of control changes. This required us to "move beyond current conceptualizations" of formal control (Kirsch 2004, p. 392) because the classic control-theoretic notion of formal control was not able to capture important differences in the content of formal control mechanisms. Specifically, contracts that started with the definition of rather generic goals (i.e., high-level project goal) were adapted over time to become more concrete and elaborate. To capture such differences in formal control we classified control mechanisms into a set of categories that is relevant to the context of this research called *specific formal control mechanisms*. Specific formal control mechanisms refine the description of an existing goal or a procedure by adding information that is tailored to a particular project situation. For example, if clients prescribe desired behaviors or outcomes in a more precise manner, add details to high-level goals, or introduce an exception to a general rule, we coded these as specific formal controls. Specific controls showed systematic differences regarding the ways through which specific requirements should be implemented (i.e., configuration or modification) and the system should be maintained (i.e., with or without out-of-standard steps). These differences are captured in four types of specific formal control (see Table 5).

Then, we examined the follow-up changes in the actor-networks. This analysis step unveiled that different types of specific controls affected the non-human actors software artifact and software task differently. Software artifacts were either changed by setting parameters in the package (configuration), by modifying the code (modification), or by modifying the code to create a new configurable feature (generalizable modification). Software tasks either remained stable or their standardization changed. We coded the need to perform additional steps for operating, maintaining,

Table 5 Types of more specific formal control mechanisms

Types	Description	Examples codes	Observed at…
Rule-based exception	Control mechanisms that specify additional, out-of-standard software requirements or additional steps for carrying out the software task in terms of developing, fixing, operating, and improving the system	• Vendor obliged to implement additional requirements without restriction to configuration-based changes • Vendor obliged to operate additional security resources in a way exclusive to the client (e.g., cameras, access control systems) • Vendor obliged to perform additional quality control before each release	Highpot: E2, E3, E4 Canidate: E1
Rule-based legalization	Control mechanisms that specifically entitle the client to request additional modifications from the vendor	• Exemption clause entitling the client to request additional customizations or bug fixes	Canidate E2,
Rule-based configuration	Control mechanisms that direct the vendor to implement more specific requirements through adapting an existing resource (i.e.,. a software, infrastructure, or employee skill set) by setting parameters (i.e., settings for a software functionality or redistribution of existing skills) instead of adapting through modifying resources (e.g., client-specific new features, dedicated infrastructures, or new skills)	• Client bans code-based modifications in favor of configuration-based changes • Contract changed so as to specifically restrict software changes to configuration • Vendor obliged to operate a dedicated IT resource (e.g., backup environment) but is allowed to follow standard procedures • Vendor obliged to decompose data entry and approval without deviating from standard activities related to data entry and approval	Alumni: E2 Payslip: E1, E2, E3 Canidate: E3
Design-based configuration	Control mechanisms that direct the vendor to first generalize idiosyncratic requirements, and then modify the system in such a way that idiosyncratic requirements can be implemented through configuration-based changes	• Vendor contracted to platformize the software • Vendor contracted to ensure new component is reusable and configurable	Alumni: E1, E3

Table 6 Capturing dynamics in outcomes—Drift and no Drift

Definition	Illustrative codes
Unbalanced Drift: The unintended movement away of software outsourcing projects from initial to emergent goals (adapted from Ciborra 2000)	• Intermittent departure from cost goals in favor of quality goals • Escalating departure from cost goals in favor of quality goals
Balanced Drift: The simultaneous movement of software outsourcing projects towards reaching both initial and emergent goals (adapted from Ciborra 2000)	• Constrained departure so that low costs are sustained while quality goals are not overly sacrificed • Stopped departure from cost goals in favor of quality goals Prevented departure so that demanding cost and quality goals are simultaneously achieved

fixing or improving the system (e.g., out-of-standard maintenance procedure) and the need to consider additional interdependencies (e.g., tasks carried out by the vendor need approval from the client) as decreases in standardization of the software task. Finally, we compared the dynamics in outcomes across episodes as "unbalanced drift" and "balanced drift" (see Table 6 for definitions). Across episodes, projects moved episodically towards successfully achieving both initial cost and emergent quality goals (balanced drift) or they focused on only one goal by either suppressing emergent goals in favor of staying with initial goals or giving up (i.e. loosing sight of) initial goals in favor of emergent goals (unbalanced drift). Thereby, cost goals refer to minimizing the time and effort spent for developing, fixing, operating, and improving the system, while quality goals refer to maximizing the degree to which the software satisfies the client's (emergent) requirements, complies with client's guidelines, operates at adequate performance, and is delivered as envisioned by the users (Moody et al. 2016; Subramanyam et al. 2012).

Stage 4—Uncovering patterns and developing a process model. To explore relationships between our core concepts, we used axial coding (Charmaz 2006; Corbin and Strauss 1990) and comparative analysis techniques, including replication logic, memo writing, tables, and visual maps (Charmaz 2006; Miles and Huberman 1994). To identify long-term dynamics, we also sorted the episodes chronologically and constructed process models that depict the dynamics of each case in terms of our key concepts. This phase of constantly comparing and integrating findings across episodes, cases, and over time resulted in our distinction if different drift processes, i.e. balanced versus unbalanced drift, which are characterized by different ways in which actor networks are being formed.

5 Results

Each of our four cases of outsourced software services started with similar initial situations. In all four projects, the vendors promised to deliver high-quality enterprise software at low costs. The respective service offers comprised delivering, fixing, operating, and improving the software that was supposed to be delivered as software-as-a-service (Saas). In all projects, the business model of the vendor rested on the premise of offering similar software services to a large number of clients. Thus, vendors had an interest to realize economies of scale and scope by standardizing software services. This interest was in line with the client's key objective in all four projects, namely saving costs and keeping future maintenance costs low. Accordingly, cost savings and maintainability were the key contractually agreed project goals.

In all four projects, however, it soon turned out that that the HR department of the Bank also had its idiosyncratic needs as expressed by the actual users. These unique requirements were generally underestimated by both the client and the vendor. They emerged as new goals that partly stayed in strong contrast to the original value proposition of relying in standard services. Notably, this general customization-versus-standardization trade-off, as it became salient in the projects, led to different responses in the projects over time. In fact, we observed differences in drift processes across projects and over time, that can be captured by two trajectories. The first trajectory reflects *unbalanced drift*, which occurred when projects moved towards one particular goal—either gradually moving towards emerging quality goals and away from initial cost goals or taking the trajectory towards emergent quality goals, but then reverted back to initial cost goals. The second trajectory reflect balanced drift, when projects moved towards reaching two rather opposing goals simultaneously, i.e. both emergent quality goals and initial cost goals. The rich case stories that follow illustrate how the client and vendor reacted to emergent goals in different ways over time. Specifically, we noted that how controls (i.e. contracts) where adapted and implemented over time played a central role for explaining whether drift occurred in a rather balanced or unbalanced way. The case stories show that external events triggered (re-) negotiations of control mechanisms in which different human actors were involved. The (re-) negotiations led to agreements between the parties that were inscribed in specific formal control mechanisms. The specific control mechanisms then trigger different adaptations of the software artifact and/or the software task that reflect either balanced or unbalanced drift.

5.1 The Highpot Case: Unbalanced Drift—Moving Away from Initial Goals

In 2009, Bank decided to outsource the provision of a software solution for skill and performance evaluation of their employees to Beta. The major objective was to harmonize the process of skill and performance evaluation across all international subsidiaries based on Beta's standard web-based solution. Thereby, Bank sought to lower long-term service costs by tailoring Beta's software solution to Bank's needs by way of configuration only. These high-level project goals were added to the contract for which contractual documents from previous partnerships served as templates.

Episode 1. Shortly after the kick-off, it turned out that Bank had misjudged the flexibility of the application. This became problematic when the future users of the system asked for a new complex feature—the "superior view" (AO). Superiors should be able to document their "true" opinion about a given employee that would be concealed from that employee. While such a feature was a contractually-defined goal, the respective contract clause was missing crucial information about the specific processes that needed to be mapped in Beta's software in order to implement it. To overcome this problem, representatives of future end users that were part of Bank's project team engaged in extensive negotiations with their counterparts from Beta. In these negotiations, the parties came to understand how exactly the "superior view" could be implemented. Bank rigorously documented the processes that needed to be mapped in the system, which allowed the parties to elicit the detailed functional requirements that were to be implemented as the "superior view". Bank's project managers then converted these functional requirements into a contractual clause that Beta was expected to implement. However, Beta soon realized that it was not possible to implement the feature through parameterization because the software did not support role-based access privileges. Hence, Beta started to modify the existing codebase of the software. This modification caused the software to deviate from the standard and this had follow-up consequences for the software task: For those parts of the software that had been changed on a code-basis, configuration through mere parametrization was not feasible anymore. Instead, changes to the software required to take effortful "*by hand measures*" (PM). Thus, moving away from the software standard had led to follow-up destandardization of the software task. While this change helped to meet the quality objectives of the project, it made software maintenance much more costly and time-consuming. Thus, the achievement of the cost goal was put at risk in favor of quality goals.

Episode 2. Shortly after, it turned out that Beta was not compliant with Bank's data center security standards. Although Beta was contractually obliged to compliance, the contract lacked the details of how exactly this could be achieved. Accordingly, Bank's project managers and compliance experts entered into negotiations with Beta's project managers in order to specify the missing details. Client managers insisted on a list of very specific measures to improve data center security (e.g. the exact location of a camera to track each person entering and leaving the datacenter and eligibility criteria for employees maintaining Bank's software). This list was then added to the contract to elaborate the already existing policy acceptance clause. While this specification of control served Bank's interest of improving security, the specific control mechanism also exerted a pressure on Beta to adapt the software task: Beta now was no longer able to follow its standard maintenance procedures but instead had to define Bank-specific roles with highly distinct layers of access rights—whereas in the standard all employees had the same unlimited access rights.

Thus, security related quality goals were better achieved but security services for Bank became more complicated leading to increases in service costs.

Episode 3. Shortly after that it turned out that Beta's solution was noncompliant with some of Bank's group-wide IT policies. In the contract, Bank and Beta had deliberately left open the relevant policy acceptance clause, because Bank knew how challenging it was for vendors to comply with all policies. And indeed, Beta was only willing to accept policies that would not entail additional code-based modifications:

> Bank has a pile of policies to all sorts of topics and they want each vendor to approve the whole pile. However, when you are a vendor of software services, you cannot do that. That is not how we work. We are neither a classical ERP-vendor nor a software engineering company… If a policy is very special we cannot implement it through configuration. (VM)

This implied that some more idiosyncratic security functionalities were never implemented. However, while Bank was willing to dispense some minor policies, Bank insisted that Beta accepted the "core" policies. Subsequently, the policy acceptance clause was elaborated accordingly, i.e. a list enumerating the specific policies Beta had to comply with was negotiated and became an addendum to the contract. To comply with the accepted policies, Beta had to conduct additional steps for maintaining Bank's software instance that diverged from its standard maintenance approach. Thus, while the specification of control served the purpose of quality, it also made the software task for Beta less standardized. This made the maintenance of Beta's solution for Bank more effortful, thus again sacrificing cost goals for a gain in quality.

Episode 4. The cumulated software changes over the course of the project had moved the software service away from the standard. This caused "*a tremendous problem*" (AO) when Beta published a new release that overwrote a number of Bank-specific modifications. Most importantly, the "*access rules*" (AO) for the "superior's view", mentioned above, were mistakenly overwritten in the new release so that all of a sudden all Bank employees had "*access to [confidential and hidden] comments of their superiors*" (AO) on their performance. When Bank realized this it caused quite a stir and Beta had to shut down the system immediately resulting in five weeks of unscheduled downtime. To ensure that future releases would not cause similar problems, Bank introduced a number of control mechanisms. First, Bank significantly tightened the release procedure Beta had to follow by introducing a new "key operating procedure" that obliged Beta to follow a more restrictive release process that forbid Beta to publish a new release without having tested it on a new Bank test platform and without approval of Bank employees. Second, Bank was upgraded to Beta's "*critical account program*" that incorporates a stricter "*quality assurance process*", a "*new deployment process*" and new, demanding service levels that measured Beta's performance in implementing the new release implementation procedures. Thus, Bank prescribed in much more detail which steps Beta had to take when new releases were published, and in addition, started to measure Beta's performance in implementing these procedures. Moreover, Bank developed a scorecard that required Beta to make their performance "*more transparent*" on a number of outcome dimensions, like "*policy compliance and time-to-market*" (AO) to enable Bank to detect problems earlier.

However, these specific control mechansisms did not address the problems' root cause—Bank's users "*continued to insist on the [Bank-specific] superior view*" (PM)—which now required code-based fixing. The underlying software task became even more complex since now Beta had to follow a highly customer-specific release procedure. Thus, on the one hand, the significantly tightened release procedure and the new scorecard enabled Bank to more quickly respond to problems that would compromise service quality; on the other hand, the measures taken forced Beta to further deviate from its standard maintenance approach, i.e. maintainability suffered, and put the cost goals at risk. Thus, the drift away from initial cost goals continued.

5.2 The Candidate Case: Unbalanced Drift—Moving Away and then Back to Initial Goals

In 2006, Bank subscribed to Gamma's software solution to support its recruitment process. The outsourcing included the provision and hosting of the software, tailoring, and maintenance. The new system should provide Bank with basically the same functionality as the prior "home-brewed" solution, but at lower costs and with instant updates. This should be achieved through configuring Gamma's solution.

Episode 1. When the project started, the contract lacked detailed functional requirements needed to tailor the software to the peculiarities of Bank's recruitment process. The parties had only elicited the desired high-level functional requirements. For instance, the contract rather generally expressed the need to connect Bank's internal HR system with Gamma's system. To further specify requirements, Bank let user representatives directly negotiate with Gamma in workshops, phone calls, and web conferences. In these negotiations the parties created a shared understanding about the variety of requirements that needed to be addressed. For example, it was exactly specified to which internal HR systems, the software needed to connect. The specific requirements were subsequently formalized as deliverables in the contract but they turned out to be difficult to implement through configuration. Gamma's software developers were left with no other option than to implement the specific deliverables through code-based modifications rather than configuration:

> Gamma had to change the codebase [of Bank's software instance] an awful lot… in case of the more complicated requirements it was more likely to be customized [through code-based changes] rather than configured. (PM)

At first, the code-based changes to the software artifact seemed to generate desirable quality results. However, the code-based changes also made the maintenance of the software less standardized. Instead of automatically processing the data received by Bank, Gamma now needed to manually transform it. Since such manual transformations are error-prone, Bank reviewed the data, and only after Bank gave approval, was Gamma allowed to continue data processing. Thus, the code-based changes that were cherished at first became a burden because the associated destandardization of the software task made operating the system costlier. The parties had departed from cost goals for a gain in quality:

Q: So it must be pretty hard to maintain the software I guess?

A: Exactly. Particularly now [after the modifications]. [VM]

Episode 2. This situation became even more problematic when the solution went live in Germany, India, the US, and the UK. The code-based modifications implemented previously had to be re-engineered for these additional regions. For example, software connectors to internal HR systems had to be rewritten to integrate with the HR systems of the new countries. While the vendor was contractually obliged to make Bank participate in the regular release cycle, the parties soon realized that the code-based modifications had accidentally accumulated to the degree to which new releases of the standard software could not be deployed anymore because Bank's software instance was simply too different from the standard. Bank realized that the dropout from the regular release cycle was both parties' fault: Bank had lured the future users of the system with the promise of addressing their "*very specific requirements*" (SM) but Gamma had also failed to anticipate the associated downsides:

> We have forced Gamma to [make changes in the code base] – but unfortunately Gamma allowed that. They should have said "No". (PM)

In its negotiations with Gamma about how to cope with the "dropout from the regular release cylce" (PM), Bank acknowledged the joint responsibility for this problem: Without consulting maintenance experts, Bank renounced its right of participating in the regular release cycle. However, Bank was not willing to renounce the promise of addressing specific user requirements. This was reflected in the negotiation result: Gamma and Bank agreed on an exception clause which exonerated Gamma from its contractual obligation to make Bank part of the regular release cycle if, in return, Gamma would implement 30 minor bug fixes or additional customizations whenever Gamma provided its other customers with a new regular release. This exception clause became the basis for evaluating the vendor. The exception clause with the explicit option to demand additional customizations legalized the end users' interest for highly specific requirements, i.e. it ensured that Bank's quality goals were met. However, each code-based customization had to be acquired in exchange for dropping a bug-fix that was part of the standard update. For this reason, the exception clause began to act like an independent force that "*pulled [Bank's software instance] farther and farther away from the standard*" (PM).

> The tool needed to be aligned with our [Bank's] processes… and once we had realized that this was not the case we sleepwalked into a [code-based] customization.

The destandardization of the software artifact through code-based changes had ripple-effects for the software task which also became less standardized: "*This made updates increasingly difficult since it wasn't standard software anymore* (AO). Rather than to follow the standard maintenance procedure, Gamma was "*trying to run and fix the system on a day-to-day basis*" (AO) which required increased oversight from Bank.

Thus, the code-based changes exerted a pressure that made fixing and improving the system more complicated and effortful. As a consequence, "*frustration levels got to such a point that it became a difficult relationship*" (SM). Users became dissatisfied and the exception clause encouraged them to address their dissatisfaction by requesting additional code-based changes such as "*customer-specific bug fixes and other customizations*" (PM):

> It's just one of those things that against original intentions became custom. We went down the route of customizations—as of request of Bank. (VM)

Again, the further the software moved away from the standard, the less standardized the software task became which made maintaining the system more effortful and time-consuming:

> The newer version of our system didn´t have the same fixes and changes as the standard system. Gamma had to re-customize or re-fix them. (SM)

Episode 3. Despite increasing dissatisfaction with the effortful maintenance of Gamma's software, Bank and Gamma seemed to have stuck with the situation until Gamma was taken over by Delta—the company Bank specifically opted against in the vendor selection process. This takeover was accompanied by drastic employee layoffs and Delta announced that it would stop supporting Gamma's software in two years. Despite the difficulties related to maintaining Gamma's software, this was against Bank's intention which had planned to continue using the software. With the drastic employee layoff not only a lot of Bank-specific knowledge got lost but also the trusted relationship "*went down like hell*" (PM). This event led Bank to initiate negotiations with the vendor to finally address the root cause(s) of difficult maintenance. In these negotiations, Bank decided to actively involve their maintenance experts which had been excluded from the previous direct negotiations between end user representatives and Gamma; and Gamma called in managers from Delta. The negotiation outcome was an extensive adaptation of control: Bank intensified the use of existing formal procedures ("*tightened the thumbscrews*" (PM)) by reducing its payment due to Gamma's failure to roll-out regular releases. Moreover, Bank immediately stopped the global rollout of the Gamma application so that fewer country specific requirements had to be fulfilled. Most importantly, however, Bank's maintenance experts and Delta's managers both worked towards replacing the exception clause with a new "*build process*" that approximated Gamma's standard build process (VM). Both had a shared interest in resorting to the initial goal of lowering service costs. The new process banned further code-based modifications and instead restricted customizations to "*configuration only*" (VM). These new control mechanisms decoupled changes in the software from changes in the software task and, consequently, maintenance stopped to become less standardized. The drift towards higher customization at the price of increasingly effortful maintenance stopped.

5.3 *The Payslip Case: Balanced Drift by Constraining Moving Away from Initial Goals*

In 2008, Bank mandated Alpha to take over the payroll processing for their Indian branch at the earliest possible date. The reason for this urgency was that Bank's previous Indian payroll vendor, Pre-Alpha, had been convicted of embezzling the payments of several customers. As Bank had already outsourced its Indian payroll for several years, they were no longer able to process it in-house. Therefore, quickly replacing Pre-Alpha's with Alpha's software, was the primary objective in the early project phase; later more idiosyncratic features came to the fore:

> We had two goals. The payroll had to be running. This works even without segregation of duties, etc. Therefore, all those security issues were initially given a low priority… later we took care of them. (PM)

Episode 1. Due to the rapid switch from one vendor to the other, the outsourcing relationship started with a note-of-understanding as the sole contractual basis. This note-of-understanding broadly obliged Alpha to quickly support Bank's core payroll process. However, the subtleties of this process remained unclear:

> [The Indian payroll] is not only about salary statements. They [Indian employees] also have a diverse benefit systems. For instance, they can reimburse the cost for drugs via the payroll. It is a cafeteria model that had to be captured in the process. (PM)

To further specify the payroll process, Bank's and Alpha's payroll experts started to negotiate:

> [The] functional experts talked to the other functional experts on the other side who understood their needs better…They used to more or less talk to each other every day or every other day in this initial phase. (VM)

In these discussions, the functional experts created a shared understanding about the specifics of the core payroll process. Because the note-of-understanding prioritized speed, the functional experts agreed to refrain from considering time-consuming code-based modifications and only considered configuration-based changes. This agreement was then elaborated in the first contract by spelling-out the detailed steps of Bank's core payroll process and how they related to the (configurable) modules of Alpha's software. This enabled the vendor to map Bank's processes in the system and process the first payroll in time:

> We worked it up very well with Alpha… They were on time. (SM)

Episode 2. After the payroll processing had gone operational, Bank turned its attention to ensuring that its idiosyncratic security needs would be fulfilled. Due to the "*hectic nature*" (PM) at the beginning of the project, the parties had only included a rather general security policy acceptance clause in the initial note-of-understanding.

> At that time…the contract needed to be reformulated to fulfill our policies. This touched on issues like disaster recovery and segregation of duties. (PM)

For disaster recovery it turned out that Bank's generic security policy was not feasible in that specific case. In particular, the general security policy required a vendor to operate dedicated backup servers in another country—but the vendor had all its operations in India. Thus, closely following the disaster recovery policy would have made operating the system unnecessarily effortful. Bank acknowledged this problem:

> …the costs for a dedicated backup environment are as high as the yearly fee [that Alpha received from Bank]. (PM)

To arrive at an adequate solution for this problem, Bank did not insist on the generic disaster recovery policy but was willing to negotiate a solution that would not lead to unnecessary increases in service costs. For this purpose, Bank actively involved its maintenance experts because they knew "*what this rule [disaster recovery] is all about*" (PM). In these negotiations, the parties came to a shared understanding that Alpha would not have to operate a dedicated backup environment in a country other than India and agreed on a way to adequately implement the standard disaster recovery clause by reconfiguring their server infrastructure:

> …the neighboring country of India is [the politically unstable country] Pakistan. Moreover, you can easily put great distances between two servers within India. (PM)

> Then we kind of gave a detailed explanation and then finally Bank agreed to supply a dedicated server and dedicated backup tapes to us while we initiated a dedicated backup environment for Bank [at one of our other Indian sites]. (VM)

This solution was formalized in the contract, i.e. Alpha was allowed to operate the backup server at one of its existing sites in India following its standard service approach. Bank covered the acquisition costs for the server on which the software was mirrored fully automatically. This detailed regulation made sure that Alpha subsequently was able to comply with the spirit of Bank's disaster recovery policy. Thus, it ensured the achievement of quality goals while also ensuring that costs would not drift away.

Episode 3. Then the challenge was was that the standard segregation of duties clause, as it was part of the contract, did not define how segregation of duties can be achieved:

> …It was about making sure that the vendor really operationalizes the respective contractual clauses, that the vendor would really do what we demanded. (PM)

Because the embezzlement of payments was the reason to switch from Pre-Alpha to Alpha, it was of major importance for Bank that Alpha would meet Bank's requirements regarding a "*clear separation between data entry and approval*" (PM). Therefore, Bank's audit experts joined the discussions between Bank's and Alpha's managers to create the knowledge about how data entry and approval could be adequately separated. Capitalizing on this knowledge they elaborated the segregation of duties clause in the contract by defining that the payroll process be broken down into subtasks and who exactly would be responsible for each subtask. However, the parties were careful to ensure that the payroll task was only decomposed into subtasks with separated responsibilities—the activities to perform the task remained unchanged so that Alpha could continue to follow its standard service approach by just reconfiguring the task responsibilities. Due to the project's "*fraud history*" (PM), security requirements were top priority and the segregration of duties, thus, seen as an important increase in system quality.

5.4 The Alumni Case: Balanced Drift—Reconciling Initial and Emergent Goals

In 2000, Bank decided to automate the global graduate recruiting process using Epsilon's software. Bank expected substantial cost savings through the automation but it was also important that the solution would support Bank's recruiting idiosyncrasies. Epsilon, a small start-up company with a subscription-based graduate recruiting solution that was still in its infancy, attracted Bank as one of its first large customers. In the contract negotiations, Epsilon agreed to re-engineer its solution fundamentally to comply with Bank's business processes.

Episode 1. After the initial re-engineering of the software, Epsilon moved forward with the plan to offer the software service—that had become highly Bank-specific—to other customers. The initial contract explicitly permitted Epsilon this move. However, Bank was concerned that with the inclusion of other customers, its own interests could recede into the background. To reconcile these conflicting interests, Bank and Epsilon decided that Bank should have "*a permanent seat at the table*" and "*benefit from Epsilon's success on the market*" (AO). Thus, Bank's investment arm acquired a minority stake in Epsilon. Being a shareholder, the investment arm was able to more directly advocate in Bank's interests; however, Bank also better understood (and benefited from) Epsilon's interest to satisfy "*all other clients*" (VM). Accordingly, the parties were looking for an approach that would simultaneously satisfy both Bank's needs and Epsilon's need to standardize across clients. They agreed to "*platformize the software*" (VM) and recorded this in the project plan. The "*platformization*" (VM) involved separating-out reusable core components and to design a system that was able to represent the Bank-specific processes through generalized modules customizable through configuration. The design of a highly configurable platform made sure that Bank's specific recruitment process was fully represented in the system, while Epsilon could continue to follow its standard service approach.

Episode 2. To guarantee future system updates, Bank's instance of the system had to be migrated to this new platform in 2007. From Bank's perspective, however, the migration procedure lacked many details and therefore created significant uncertainty. To further specify the migration procedure, project personnel from Bank and Beta negotiated and "*discussed on the phone*" (VM) very precise steps how the migration should be conducted:

> …we have to be very, very prescriptive about what we want because they [the vendor] don´t think of operating in such a global environment…so often we are teaching them on how to get things done and [the migration procedure] is a classic example… [We] actually pulled all that together but the benefit of that is that then we sit in the driver's seat…and we could tell them exactly how and what we wanted. (AO)

In these negotiations, the parties continued to equally take the interests of both Bank and all other users into account, so that neither of the parties questioned the previously taken decision that all Bank-specific requirements would have to be implemented through configuration rather than modification. The subsequent elaboration of the migration procedure involved a detailed business plan stipulating Bank's needs, a communication plan specifying exactly who at Bank had to be informed about which migration activities, and an elaborate test plans and establishing pass/fail criteria for those tests. The subsequent exercise of the migration procedure made sure that Bank's idiosyncratic recruitment process was still supported—but without any code-based modifications. This helped to achieve the quality goals without compromising costs.

Episode 3. Earlier in the project the Bank had introduced a change procedure specifying how changes requested by Bank had to be formally processed. Naturally, the specific contents of each change were not known. Hence, whenever the change procedure was exercised, the parties performed formal project steps to further clarify the contents of each change:

> What usually happens—we have procedures that we go through but often Bank might ask us, they have a query, they have something that needs to be fixed and they will contact me and we have kind of an informal discussion about what we want. (VM)

Once such a specification document was approved, it became an addendum to the contract defining new or elaborating existing deliverables for Epsilon. Such changes were always bearing the danger that Bank's software instance would move away from the standard. However, because Bank had a stake in Epsilon, it was in Bank's genuine interest that not only Bank would be satisfied by the changes but also the other users. Since it was only minority stake, Bank was not able to push-through changes against Beta's will. Therefore, all change requests were restricted to configuration-based changes which were unproblematic from the vantage point of standardization. In fact, they made sure that the idiosyncrasies of Bank's recruitment process were always effectively represented in the system without increasing maintenance effort.

Episode 4. The last specification of formal control was triggered when Bank simulated an attack on Epsilon's system (penetration test) to monitor whether Epsilon's software had security loopholes. Epsilon was contractually obliged to comply with certain security standards. The simulated attack, however, revealed a number of security loopholes:

> We gave the attackers some information and they tried to break into the system…and they succeeded. A couple of hours after the penetration test started, one of the attackers called us and said that he would control the system console now. (PM)

To ensure that Epsilon would comply with Bank's security standards in the future, the parties aimed at elaborating the contract's security clause to include specific guidelines, "*how these problems should be resolved*" (PM). Bank and Epsilon's project personnel negotiated how to best address the security issues. This was formalized in an "*action plan*" (PM) that elaborated the already existing security clause by exactly specifying which security issues had to be resolved. Now the vendor was urged to "*solve the problems… [which] required software development*" (PM). Yet, no discussion sparked as to whether this software change was covered by the contract. This was because of the way Epsilon implemented the change: Similar to when the software was initially "platformized", Epsilon made sure to not only address Bank's specific security requirements but to generalize these issues such that they could "*offer a secure solution to the market*" (PM). Epsilon decided to make the software change part of its standard functionality—Bank carried some of the associated costs indirectly through its minority stake. Thus, the change ensured that Bank's security needs were met while Epsilon could continue to rely on its standard maintenance approach—quality goals were met without compromising costs.

5.5 Cross-Case Analysis

The case stories showed that projects with similar starting conditions moved into different directions. Highpot constantly drifted away from cost goals for gains in quality. Candidate also drifted but eventually this drift was stopped. In Payslip the parties never lost sight of cost goals when pursuing quality goals so that drift was constrained from the beginning. In Alumni drift was prevented in a way that simultaneously addressed cost and quality goals. We next show how these differences in drift across cases and over time can be explained by three systematic differences in the control process: (1) The composition of the negotiation network, (2) the negotiated type of specific control, and (3) the consequent coupled versus decoupled relationship between the software artifact and task. Table 7 provides a summary of evidence.

5.6 The Composition of the Negotiation Network

The first key difference in control processes pertains to the composition of the negotiation network, i.e. whether it is narrow or broad. In narrow negotiation networks the variety of actors that represent distinct interests in the negotiation of control is small—so that some actors with valid interests may not be enrolled. This meant that actors that championed quality interests such as project managers or end users did participate in the negotiation, whereas actors with vested cost interests such as maintenance personnel did not (see Canidate E1 & E2, Highpot E1-E4). For example, at Highpot actors with a strong focus on quality (i.e., end user representatives and audit experts) consistently participated in negotiations and insisted that the software system would have to satisfy their specific requirements—thereby upstaging cost goals.

Table 7 Summary of evidence

Episode	Negotition network	Specific control	(De-) coupling	Destandardization	Drift
Drift pattern					
Highpot E1	Narrow	Rule-based exception	Coupling	Yes: S & T	Yes: Intermittent
Highpot E2	Narrow	Rule-based exception	Coupling	Yes: T	Yes: Intermittent
Highpot E3	Narrow	Rule-based exception	Coupling	Yes: T	Yes: Intermittent
Highpot E4	Narrow	Rule-based exception	Coupling	Yes: S & T	Yes: Intermittent
Canidate E1	Narrow	Rule-based exception	Coupling	Yes: S & T	Yes: Intermittent
Canidate E2	Narrow	Rule-based legalization	Coupling	Yes: S ↔ T	Yes: Escalating
No Drift pattern					
Canidate E3	Broader	Rule-based configuration	Decoupling	No	No: Stopped
Payslip E1	Broader	Rule-based configuration	Decoupling	No	No: Impeded
Payslip E2	Broader	Rule-based configuration	Decoupling	No	No: Impeded
Payslip E3	Broader	Rule-based configuration	Decoupling	No	No: Impeded
Alumni E1	Broad	Design-based configuration	Decoupling	No	No: Prevented
Alumni E2	Broad	Design-based configuration	Decoupling	No	No: Prevented
Alumni E3	Broad	Design-based configuration	Decoupling	No	No: Prevented
Alumni E4	Broad	Design-based configuration	Decoupling	No	No: Prevented

By contrast, broad negotiation networks enroll additional actors with distinct interests that were previously marginalized. This meant that not only actors that championed Bank's quality interests participated in the negotiation but also actors that championed Bank's cost interests and/or actors that championed the interests of all other users—including users from other clients (see Canidate E3, Alumni E1-E4, and Payslip E1-E3). For example, at Candidate the parties focused on quality goals at the expense of cost goals for a long time (see Candidate E1 and E2)—but after opening negotiations for actors with vested cost interests, these interests came to bear (see Candidate E3). In both the Alumni and the Payslip case, negotiation networks were broad right from the beginning and remained broad over time, so that the parties always had an eye on both cost and quality goals (see Alumni E1-E4, Payslip E1-E3).

5.7 Types of Specific Control

Negotiation networks of different composition tend to define different types of specific control. Narrow negotiation networks define two types, i.e. rule-based exceptions and rule-based legalizations (see Canidate E1 & E2, Highpot E1-E4). *Rule-based exception* clauses specify one-time customizations of the software or the vendor's processes. *Rule-based legalization* clauses permanently entitle the client to request customizations. In line with the one-sided quality focus of narrow negotiation networks, these two types of specific control fixate on enabling Bank-specific customizations but do not bother about long-term service costs.

By contrast, broad negotiation networks tend to define rule-based configurations and design-based configurations (see Canidate E3, Alumni E1-E4, and Payslip E1-E3). *Rule-based configuration* clauses restrict customization requests so that they are only approved if they can be implemented by setting parameters in the software system. *Design-based configuration* clauses direct the vendor to generalize idiosyncratic requirements of Bank in such a way that can be implemented by setting parameters. In line with the more balanced focus of broad negotiation networks, these two types of specific control direct projects to reconcile cost and quality goals.

5.8 Coupling Versus Decoupling Software Artifact and Software Task

Different types of specific control mechanisms entail different follow-up inscriptions in the software artifact and the software task. *Rule-based legalizations* and *rule-based exceptions* both inscribe a *close coupling* between the software artifact and the task, so that changes in the software artifact destandardize the follow-up activities of fixing, operating, and improving the system (see Canidate E1 & E2, Highpot E1-E4). By contrast, *rule-based configuration* and *design-based configuration* both *decouple* changes in the software artifact from changes in the task so that changes in the software artifact do not destandardize the software task (see Canidate E3, Alumni E1-E4, and Payslip E1-E3).

While both rule-based legalizations and rule-based exceptions exert a destandardization pressure, the interplay of software artifact and software task that underlie this pressure and the consequences for drift slightly differs between the two. First, rule-based exceptions directly inscribed additional steps for carrying out the software task (see Highpot E2 & E3) or they inscribed code-based software modifications that indirectly made the software task less standardized (see Canidate E1, Highpot E1 &

E4). In these cases drift manifested as an intermittent departure (see Canidate E1, Highpot E1-E4). Second, rule-based legalizations, permanently inscribed an opportunity for Bank to request additional customizations; and those customizations were not restricted to configuration-based changes (see Candidate E2). This inscribed a closely-coupled relationship of reinforcing de-standardization between the software task and the software artifact: Whenever the legalization clause was exercised, the software was tailored through additional code-based modifications and as a consequence, the software task increasingly moved away from standard procedures. This resulted in quality problems that sparked dissatisfied users to demand additional changes. To enforce this interest, users leveraged the legalization clause that had permanently inscribed an open gate for software modifications. Accordingly, users formulated additional change requests and the vendor implemented them through additional code-based software modifications. This further de-standardized the software task. Thus, drift, escalated, i.e., the departure from cost goals occurred frequently and in a self-reinforcing manner (see Canidate E2).

While both rule-based configurations and design-based configurations exert a standardization pressure, the interplay of artifact and task that underlie this pressure and the consequences for drift slightly differ between the two. First, rule-based configurations decoupled software changes from task changes because if the software was only changed by adjusting parameters, system maintenance could continue to closely follow the standard (see Payslip E1-E3, Candidate E3)—drift was constrained. Second, design-based configuration clauses also decoupled software changes from task changes because actors sought for similarities between various requirements and abstracted them into common, configurable properties For example, at Alumni, the client contracted the vendor to platformize the software right from the beginning and also made sure that subsequent idiosyncratic requirements were implemented through configuration—so that they became a reusable and configurable component of the standard software. As a consequence, vendor personnel could continue to follow its standard maintenance approach. This ensured high levels of software quality through severe customizations while allowing the realization of economies of scale and scope and hence cost efficient software provision. Thus, drift was prevented.

Importantly, to effectively prevent drifts it appears imperative that the specific control mechanism affects the nature of the relationship between task and artifact (i.e., decoupling)—rather than affecting task and artifact independently. For example, at Highpot the parties introduced a rule that obliged the vendor to continuously make key performance metrics available to the client, and the client evaluated vendor performance with a new scorecard—enabling Bank to detect problems earlier and react faster. At the same time, customizations through modification-based changes in the software artifact were still possible. This undermined efforts to stop the departure from cost goals because it continued to make quality assurance and updating more effortful. Thus, drift was not prevented (see Highpot E4).

5.9 Explaining Drift

By comparing the identified patterns of drift, drift may be explained as a process of translation that occurs through interactions between the three key actors that together form an actor-network of an outsourced software development project. The first refers to the human actors, i.e. client and vendor representatives involved in (re-)negotiating controls (i.e. the contract). This negotiation network can be composed differently which has important implications for the kind of controls that are being negotiated. These specific controls represent the second type of actor. The (re-)defined controls take on the role of actors as they serve as reference points for future actions, i.e. the actual development work. Any change in the contract should subsequently be followed by both client and vendor personnel. The actual actions taken are the software development tasks, which may be carried out differently based on what kind of specific controls were renegotiated. Moreover, once the tasks are carried out they result in changes in the actual software artifact, which serves as a building block for future changes in the artifact. Thus, changes in the software task and the software artifact are interrelated and together form the third type of actor. They represent actors because, again, they prescribe future actions in terms of routines being followed and material that is worked on.

Depending on how these three types of actors are formed, drift takes on different forms, which most generally can be described as unbalanced versus balanced drift. Such formation of the actor network has its starting point in the composition of the negotiation network, which could be formed rather *narrowly* with emphasis on a group of stakeholders, hence representing rather one-sidedly oriented project goals, or *broadly* including different stakeholders with multi-sided project goals that appear to contradict each other, such as cost and quality goals. Narrow negotiation networks tend to systematically marginalize actors with vested interests in one goal (i.e. cost savings) in favor of another (i.e. quality), whereas in broader negotiation networks a diverse set of actors can bring their often different interests to bear. These differences in the composition of negotiation networks translate into different types of specific control mechanisms: Narrow negotiation networks tend to define mechanisms in favor of one particular goal, i.e., rule-based legalization and rule-based exception. Broad negotiation networks tend to specify mechanisms that reconcile different goals (i.e., design-based configuration and rule-based configuration). These different types of specific control mechanisms entail different follow-up inscriptions in the software artifact and the software task. Rule-based legalizations and rule-based exceptions both inscribe a close coupling between the software artifact and the task. This triggers a process of destandardization. For example, exceptions allow the software to move away from the standard which in turn results in the software task to become less standardized as well, and vice versa.

This recursive process causes projects to drift away from one goal, such as cost efficiency, towards another goal, such as quality, which we refer to as *unbalanced drift*. By contrast, rule-based configuration and design-based configuration decouple changes in the software artifact from changes in the task. In this case, changes in the software towards reaching one particular goal, such as quality, do not undermine the other goal, such as costs, because these changes have no ripple-effects for the follow-up activities of operating, improving, and maintaining the software. Thus, the software task remains unchanged. For example, if a software is configured to meet a change request, this does not affect the nature of follow-up maintenance of the software such that cost and quality goals are approached simultaneously. Accordingly, such drift is oriented towards achieving multiple goals (i.e. cost efficiency and quality), which we refer to as *balanced drift*.

6 Discussion

This study asked how and why outsourced software development projects often take on unexpected routes in spite of the fact that they are closely controlled based on contractual agreements between client and vendor. We argue that the actual adaptations of control over time are critical for explaining such drift and we frame this adaptation process of controls as one of changing actor networks. The heart of actor network theory lies in viewing processes of change as transformations of actor networks, i.e. changing constellations of actors and their relationships, whereas actors can be both human and non-human. This nicely resonates with the fact that outsourcing relationships may not just be viewed as relationships between humans (i.e. client and vendor representatives) but also as constituted in artifacts such as the contractual agreements between client and vendor, the actual tasks carried out through project personnel, and the software artifact being created. Based on our analysis of four outsourced software development projects over time, we unveiled that the respective actor networks evolved differently over time reflecting two different forms of drift. The first refers to unbalanced drift, where projects either increasingly left their intended trajectory moving towards emergent goals or deliberately suppressed emerging goals reverting to the initial trajectory. The second refers to balanced drift, where emergent goals are reconciled with initial goals. This occurs either in an extenuated form where the recognition of emergent goals is constrained by following initial goals, or, on equal footing, where initial and emergent goals are sought to complement each other (i.e., in the spirit of ambidexterity). We observed some of these specific variants of drift only in particular cases. Nevertheless, based on our within case analysis (i.e. analysis across episodes of control change) and across case analysis, we derived a general process-theoretic logic that views differences in control processes of outsourced projects as transformations of actor networks reflecting either unbalanced or balanced drift. Next we outline the implications of our findings for research on control of software development project in general, as well as outsourced IT projects, specifically.

6.1 Theoretical Implications

To motivate our study we argued that the phenomenon of drift in IT outsourcing opens up the opportunity to view control from a new perspective. Rather than following the traditional logic of control that emphasizes the role of choosing and adapting controls for goal achievement, we borrowed and contextualized alternative assumptions and concepts from ANT. Using ANT for scaffolding allowed us to understand drift as a process of transformation that enfolds along the network of relevant actors that together account for changing directions in outsourced software development projects. Such actor networks include both human ones (i.e. client and vendor representatives, such as managers, user representatives, or technical experts) and non-human ones (e.g. the specific controls defined in the contract and their implementations in terms of tasks carried out and software artifacts produced). These actors and their relationships change over time and hence shape an open-ended process of controlling software outsourcing projects which has important consequences for how and whether initial goals and emergent goals are being addressed over time, i.e. either in a balanced o unbalanced way. Three major theoretical implications accrued from our ANT perspective on control in outsourced software projects.

Novel conception of the controller. First, our findings unveil that the way in which controls (i.e. contracts) are adapted over time can differ considerably from the conception of a rational actor that makes optimal control choices (Eisenhardt 1985; Ouchi 1979; Rustagi et al. 2008; Tiwana and Keil 2009). Viewing the controller as a rational actor implies that controllers do not only anticipate the consequences of their actions, but that they also instantly recognize goal deviations and perform immediate and unrestricted control adaptations (Choudhury and Sabherwal 2003; Eisenhardt 1985; Ouchi 1979). Our findings challenge this conception and provide the ground to develop a controller conception closer to empirical reality: First, individual actors often seemed unaware that their control adaptations would have ripple-effects for the software artifact and/or the task and hence for goals that are being targeted by the project. Often times, individual controllers have difficulties or are not willing to fully anticipate the wider consequences of the contextual changes that their control choices cause. Specifically, if controls were negotiated by a narrowly scoped group of stakeholders, representing only particular interests (e.g. primarily the ones by the users), it was more likely that drift occurred in a rather unbalanced way, where initial goals (such as cost efficacy) increasingly got out of sight in favor of following emergent goals, or, where emergent goals arose but were rather suppressed by strictly following initial goals. Hence, the ability to recognize and follow emergent goals besides initial ones hinges on the (changing) composition of negotiation networks. Moreover, controllers were often not able to make immediate and unrestricted adjustments of control because the complex interactions between actors (i.e. control mechanisms, software artifact and task) created path-dependent dynamics in the sense that changes in the past shaped the options in the future. Our study unveils three forces that create such paths and thereby limit the ability to make immediate and unrestricted adjustments. First, control mechanisms often inscribe changes into the software artifact that are difficult to reverse. For example, to reverse deep changes to the code base of a software—inscribed by a previous control mechanism—the architecture may need to be fundamentally refactored and many of the components rewritten. This disavows the option to encourage behaviors to reverse deeply-inscribed software changes. For example, in the Candidate case actors were only willing to take this option after a strong external event called into question the previous modifications to the software. Second, this irreversibility problem is aggravated by control mechanisms that inscribe close coupling between the software artifact and the software task. This creates a self-reinforcing destandardization pressure on the interrelation between artifact and task such that destandardization of the software artifact results in destandardization of the software task, and vice versa. For example, if the software artifact is repeatedly modified on a code-basis such changes become difficult to reverse. Third, different types of specific control mechanisms favor some options of change over other others long after they have been set in place. Rule-based legalizations of software changes permanently institutionalize the entitlement to request code-based modifications. Thus, the "legal" option to reject such requests becomes immaterial. By contrast, rule-based and design-based configuration clauses systematically restrict customizations to those changes that can be implemented by changing parameters—making away with the legal basis for code-based modifications in the future.

Novel insights on the sources of control change. Our findings show that controller and controllee are not fixed entities; instead control is adapted in negotiation networks—the composition of which may change over time. Prior research has explained control selection and adaptation as an outcome of changes in the external controller context such as changes in the controller's environment or knowledge (Benaroch et al. 2016; Chen and Bharadwaj 2009; Gopal and Koka 2012; Gopal et al. 2003; Rustagi et al. 2008; Tiwana 2010; Tiwana and Keil 2009). Our findings extend this research by unveiling the changing composition of negotiation networks as a novel source of control change. Specifically, narrow negotiation networks tend to refine controls in the interest of one particular party. This leads to marginalize one goal (e.g., the initial goal of cost savings) in favor of another goal (e.g., emergent quality goals) which characterizes rather unbalanced drift. Only when negotiation networks were broadened to include actors who represented and acted in favor of opposing goals (i.e. initial and emergent ones) compromises where sought for to reconcile goals, which reflects balanced drift. Thus, control changes are grounded in the composition of negotiation networks. Notably.

Prior research framed the control context (e.g. the task) as an external factor that shapes control change but is not affected back by control (Eisenhardt 1985; Ouchi 1979). In contrast, our findings show that the control context (i.e., the software artifact and the software task) and changes in control (i.e. specific control mechanisms), mutually shape each other in co-evolutionary processes that systematically create different paths. This suggests, that the classic idea of control change being mainly driven by changes in the external context is incomplete—in fact systematic differences in the control context are often not external but the result of systematic differences in preceding control dynamics.

Novel insights on the relationship between control and performance. Prior cross-sectional research had difficulties in substantiating the expected positive link between formal control and goal achievement in the context of software outsourcing (Gopal and Gosain 2010; Srivastava and Teo 2012; Tiwana 2010; Tiwana and Keil 2009). Specifically, prior research found a prevalence of formal outcome control in software outsourcing but these higher levels of outcome control were not connected to higher performance. More recent research on the dynamics of control has focused on the drivers of control change but paid little attention to the consequences of control dynamics on goal achievement (Choudhury and Sabherwal 2003; Gregory et al. 2013; Huber et al. 2013; Remus and Wiener 2012). Our findings contributes to closing this gap in two important ways. First, they point to the importance of thinking beyond the dominant modes of formal control, i.e. outcome and process control (Ouchi 1979), to also consider control of the solution space reflecting the degree of freedom of choosing both the process and the outcome. As such, we identified the two IS-idiosyncratic alternatives of how to develop software, configuration versus modification, as rather opposing controls defining the solution space. While configuration only allows changes within the given set of parameters of a software, modification allows unrestricted changes. Notably, we found the narrowing of the solution space as an important condition for achieving balanced drift, i.e. addressing opposing goals, such as achieving quality and cost efficiency, while leaving the solution space open rather encouraged unbalanced drift. This means that change should occur within boundaries that may need to be redefined over time. This became evident by the interplay that we observed between rule-based configuration and design-based configuration. Once, the existing configuration space is insufficient to balance initial and emerged goals, it is necessary to redesign the configuration space through architectural innovation (e.g. through platformization of a software) so that the solution space is enriched based on which software can be configured. Controls that both enrich and constrain the solution space simultaneously, such as rule-based configuration and design-based configuration, provide the client and the vendor the opportunity to cope with emergent requirements while doing so efficiently. They do so by inscribing a decoupling between the software artifact and the software task so that cost-increasing, self-reinforcing processes of destandardization are prevented. Our findings hence take up recent calls for novel conceptualizations of control in the IS context (Choudhury and Sabherwal 2003; Moody et al. 2016; Wiener et al. 2016).

Our findings on the importance of considering control of the solution space as complementary to pure outcome and process control also nicely resonate with software engineering literature. A key idea of the product and software engineering literature is that there is a close link between the chosen architectural approach and the follow-up work processes of developing, fixing, operating, and improving the system (Baldwin and Clark 2000; Ulrich 1994). Specifically, compared to the architectural approach of integration, modularization lowers the effort for developing and maintaining the system while still allowing for customization (Baldwin and Clark 2000; Ulrich 1994). This power of modularity is rooted in its architectural logic that decouples different subsystems from each other while ensuring that all subsystems work together as a complex whole (D'Adderio and Pollock 2014). Our study shows how the principles of modular design can be actively instilled in software outsourcing projects and that doing so has important implications for controlling drift, i.e. allowing for balanced drift.

6.2 Future Research

Future research may draw on our findings to further refine our understanding of how different control processes influence drift. A longitudinal event-time series study could transform the processes that our study unveiled into event chains to further substantiate the patterns that we have uncovered. Such an event-time series study should also systematically incorporate contingency variables or different types of triggers (Thomas and Bostrom 2010) to extend the generalizability of our findings. For instance, we suspect that vendors with better technological capabilities will be better at engaging in generalizable modification, thus, we expect projects with technologically more capable vendors to engage in such processes more often, and to rather enter into balanced drift. Moreover, beyond capabilities the motivation of the vendor to take investments into design changes appear worthy of study. Such a study should develop fine-grained instruments to distinguish between short-term and long-term performance effects such as the short-term gains in quality and the long-term escalations of costs observed in this study. Moreover, such a study should try to incorporate additional time-bound dimensions of success. We would expect that similar short- versus long-term tradeoffs may exist with regard to cost efficiency versus innovativeness of the software service (Aubert et al. 2015), or the ability to flexibly switch between vendors versus the specific investments into an outsourcing relationship.

6.3 Practical Implications

Practical advice for how managers can prevent software outsourcing projects from drift has thus far been scarce. Our findings show that in order to sustain the big but potentially contradictory benefits of the outsourcing arrangement, managers need to make mindful decisions about who to involve in the negotiation of control mechanisms and what to specify in these mechanisms. Specifically, managers are advised to carefully identify all actors with valid stakes in the project and then to actively open negotiations for these actors. This will enable managers to define specific control mechanisms that decouple changes in the software artifact from changes in the task which is key to simultaneously address competing cost and quality goals. This study has shown through which strategies this can be achieved: Managers can offer direct contractual remunerations to the vendor company that cover the initially high extra efforts for generalization. In situations in which such extra efforts are very substantial, managers can arrange an agreement that the vendor receives a substantial initial payment but the client in return receives a share from potential future profits of the vendor that may arise from offering the generalized extra-functionality to other clients.

References

Aubert, B. A., Kishore, R., & Iriyama, A. (2015). Exploring and managing the "innovation through outsourcing" paradox. *The Journal of Strategic Information Systems, 24*(4), 255–269.

Baldwin, C. Y., & Clark, K. B. (2000). *Design rules: The power of modularity*. Cambridge: MIT Press.

Benaroch, M., Lichtenstein, Y., & Fink, L. (2016). Contract design choices and the balance of ex-ante and ex-post transaction costs in software development outsourcing. *MIS Quarterly, 40*(1), 57–82.

Charmaz, K. (2006). *Constructing grounded theory: A practical guide through qualitative research*. London: Sage.

Chen, Y., & Bharadwaj, A. (2009). An empirical analysis of contract structures in IT outsourcing. *Information Systems Research, 20*(4), 484–506.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. *Information Systems Research, 14*(3), 291–314.

Ciborra, C. (1999). A theory of information systems based on improvisation. In Robert Galliers Wendy (Ed.), *Rethinking management information systems: An interdisciplinary perspective, C.* Oxford (UK): Oxford Unviersity Press.

Ciborra, C. (2000). *From control to drift: The dynamics of corporate information infrastructures*, Oxford University Press.

Corbin, J., & Strauss, A. (1990). Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative sociology, 13*(1), 3–21.

Cram, W. A., Brohman, K., & Gallupe, R. B. (2016). Information systems control: A review and framework for emerging information systems processes. *Journal of the Association for Information Systems, 17*(4), 216–266.

Cullen, S., Seddon, P., & Willcocks, L. (2005). Managing outsourcing: The life cycle imperative. *Management Information Systems Quarterly Executive, 4*(1), 229–246.

D'Adderio, L., & Pollock, N. (2014). Performing modularity: Competing rules, performative struggles and the effect of organizational theories on the organization. *Organization Studies, 35*(12), 1813–1843.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2004). Information systems outsourcing: A survey and analysis of the literature. *ACM SIGMIS Database, 35*(4), 6–102.

Eisenhardt, K. M. (1985). Control: Organizational and economic approaches. *Management Science, 31*(2), 134–149.

Eisenhardt, K. (1989). Agency theory: An assessment and review. *Academy of Management Review, 14*(1), 57–74.

Elbanna, A. R. (2006). The validity of the improvisation argument in the implementation of rigid technology: The case of ERP systems. *Journal of Information Technology, 21*(3), 165–175.

Gewald, H., & Dibbern, J. (2009). Risks and benefits of business process outsourcing: A study of transaction services in the German banking industry. *Information & Management, 46*(4), 249–257.

Goo, J., Kishore, R., Rao, H., & Nam, K. (2009). The role of service level agreements in relational management of information technology outsourcing: An empirical study. *MIS Quarterly, 33*(1), 119–145.

Gopal, A., & Gosain, S. (2010). The role of organizational controls and boundary spanning in software development outsourcing: Implications for project performance. *Information Systems Research, 21*(4), 960–982.

Gopal, A., & Koka, B. R. (2012). The asymmetric benefits of relational flexibility: Evidence from software development outsourcing. *MIS Quarterly, 36*(2), 553–576.

Gopal, A., Sivaramakrishnan, K., Krishnan, M., & Mukhopadhyay, T. (2003). Contracts in offshore software development: An empirical analysis. *Management Science, 49*(12), 1671–1683.

Gregory, R. W., Beck, R., & Keil, M. (2013). Control balancing in information systems development offshoring projects. *MIS Quarterly, 37*(4), 1211–1232.

Hanseth, O., Aanestad, M., & Berg, M. (2004). Guest editors' introduction: Actor-network theory and information systems. What's so special? *Information Technology & People, 17*(2), 116–123.

Hanseth, O., & Monteiro, E. (1997). Inscribing behaviour in information infrastructure standards. *Accounting, Management and Information Technologies, 7*(4), 183–211.

Huber, T. L., Fischer, T. A., Dibbern, J., & Hirschheim, R. (2013). A process model of complementarity and substitution of contractual and relational governance in IS outsourcing. *Journal of Management Information Systems, 30*(3), 81–114.

Kim, H., Shin, B., & Lee, H. (2013). The mediating role of psychological contract breach in IS outsourcing: Inter-firm governance perspective. *European Journal of Information Systems, 22*(5), 529–547.

Kirsch, L. J. (1996). The management of complex tasks in organizations: Controlling the systems development process. *Organization Science* (7:1), 1–21.

Kirsch, L. (2004). Deploying common systems globally: The dynamics of control. *Information Systems Research, 15*(4), 374–395.

Lacity, M., Khan, S., & Willcocks, L. (2009). A review of the IT outsourcing literature: Insights for practice. *The Journal of Strategic Information Systems, 18*(3), 130–146.

Lacity, M. C., Khan, S., Yan, A., & Willcocks, L. P. (2010). A review of the IT outsourcing empirical literature and future research directions. *Journal of Information Technology, 25*(4), 395–433.

Langley, A. (1999). Strategies for theorizing from process data. *Academy of Management Review, 24*(4), 691–710.

Latour, B. (1996). On actor-network theory: A few clarifications. *Soziale welt, 47*(4), 369–381.

Latour, B. (1999). On recalling ANT. *The Sociological Review, 47*(S1), 15–25.

Law, J. (1987). Technology and heterogeneous engineering: The case of Portuguese expansion. In *The social construction of technological systems: new directions in the sociology and history of technology* (No. 1, pp. 1–134).

Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. London: Sage.

Monteiro, E. (2000). *Actor-network theory and information infrastructure* (pp. 71–83). Oxford: Oxford University Press.

Moody, G., Kirsch, L. J., Slaughter, S. A., Dunn, B. K., & Weng, Q. (2016). Facilitating the transformational: An exploration of control in cyberinfrastructure projects and the discovery of field control. *Information Systems Research, 27*(2), 324–346.

Ouchi, W. G. (1979). A conceptual framework for the design of organizational control mechanisms. *Management Science, 25*(9), 833–848.

Poole, M. S., Van de Ven, A. H., Dooley, K., & Holmes, M. E. (2000). *Organizational change and innovation processes: Theory and methods for research*. USA, New York: Oxford University Press.

Remus, U., & Wiener, M. (2012). The amount of control in offshore software development projects. *Journal of Global Information Management, 20*(4), 1–26.

Rustagi, S., King, W. R., & Kirsch, L. J. (2008). Predictors of formal control usage in IT outsourcing partnerships. *Information Systems Research, 19*(2), 126–143.

Sarker, S., Sarker, S., & Sidorova, A. (2006). Understanding business process change failure: An actor-network perspective. *Journal of Management Information Systems, 23*(1), 51–86.

Srivastava, S. C., & Teo, T. S. (2012). Contract performance in offshore systems development: Role of control mechanisms. *Journal of Management Information Systems, 29*(1), 115–158.

Subramanyam, R., Ramasubbu, N., & Krishnan, M. (2012). In search of efficient flexibility: Effects of software component granularity on development effort, defects, and customization effort. *Information Systems Research* (23:3-part-1), 787–803.

Thomas, D. M., & Bostrom, R. P. (2010). Vital signs for virtual teams: An empirically developed trigger model for technology adaptation interventions. *Management Information Systems Quarterly, 34*(1), 115–142.

Tiwana, A. (2010). Systems development ambidexterity: Explaining the complementary and substitutive roles of formal and informal controls. *Journal of Management Information Systems, 27*(2), 87–126.

Tiwana, A., & Keil, M. (2009). Control in internal and outsourced software projects. *Journal of Management Information Systems, 26*(3), 9–44.

Tjornehoj, G., & Mathiassen, L. (2008). Between control and drift: Negotiating improvement in a small software firm. *Information Technology & People, 21*(1), 69–90.

Ulrich, K. (1994). Fundamentals of product modularity. In S. Dasu, & C. Eastman (Eds.), *Management of design*, Springer, Heidelberg, pp. 219–231.

Wiener, M., Mähring, M., Remus, U., & Saunders, C. S. (2016). Control configuration and control enactment in information systems projects: Review and expanded theoretical framework. *MIS Quarterly, 40*(3), 741–774.

Yin, R. (2009). *Case study research: Design and methods*. London: Sage.

Cybersecurity Risk in IT Outsourcing—Challenges and Emerging Realities

Michel Benaroch

Abstract IT outsourcing (ITO) is a major contributor to cybersecurity risk exposure. When organizations outsource IT needs and/or cybersecurity functions, they explicitly or implicitly assume that ITO providers bear the responsibility for cybersecurity risk. In reality, ITO clients' risk profile changes and becomes a combination of their risks and a subset of their ITO provider risks. This paper discusses cybersecurity risk challenges that are exacerbated in the ITO context and a commonly made argument that ITO client-provider *trust* can improve the management of cybersecurity risk. The paper proceeds to contrast three views on how to build trust with ITO providers: *decision-theoretic view*, *transparency-based view*, and *market-based view*. It shows that the market-based view is most likely to emerge as the dominant model for client-provider trust. Market-based trust involves market mechanisms that reward and penalize ITO service providers for obtaining cybersecurity certifications from independent, trusted third-party agencies. Specifically, the same way firms that obtain cybersecurity certifications benefit from positive market reactions that create firm value, so do firms that experience cybersecurity incidents indicating failures of certified IT security suffer punitive market reactions that destroy firm value. The paper elaborates on the feasibility of market-based trust in the ITO context, and shows that it works in the context of cyber failures and IT insourcing. The paper concludes with a discussion of obstacles to widespread adoption of market-based trust by ITO players.

1 Introduction

There is growing evidence that IT outsourcing (ITO) is a major contributor to cybersecurity risk exposure. Reports of cybersecurity incidents linked to IT providers arrive regularly. Recent examples are Salesforce's multi-hour cloud meltdown due

M. Benaroch (✉)
Whitman School of Management, Syracuse University, Syracuse, USA
e-mail: mbenaroc@syr.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_13

to a database blunder that granted users access to all data (May 2019); CapitalOne's data breach due to a former Amazon Cloud Services employee who hacked over 100 million customers' data hosted on Amazon's cloud (July 2019); and, Google's cloud outage that took down YouTube, Gmail and Snapchat in parts of the U.S. (June 2019). There is broader evidence as well. Benaroch and Chernobai (2015) find that, in their sample, almost one third of cyber incidents compromising the confidentiality of data and IT assets originate with IT providers. Vasishta et al. (2018) study data breaches in the healthcare industry, showing that over 30% of breaches have happened at outsourcing business associates' site. In VERIS, Verizon's extensive repository of data breaches, about 20% of the incidents involve business partners, vendors, and other 3rd parties (Verizon 2017). By another account, 13% of all cyber insurance claims sampled in 2016 were caused by third-party providers (NetDiligence 2016).

ITO continues growing at a steady pace, as is its associated cybersecurity concerns. One contributor is the rising reliance on cloud-computing service providers (CSP). In cloud computing, clients' risk exposure grows as they move sensitive data to federated cloud environments that may be hosted with multiple providers and subproviders belonging to different legal entities in various jurisdictions (Akinrolabu and New 2017). Such a layered cloud supply chain will invariably pose greater risk than a single provider (Weber and Staiger 2014). Clients' loss of control exacerbates cybersecurity concerns. Cloud providers are hesitant, or just not able, to share information on suppliers, data location, architecture, and security details of infrastructures confidential to the cloud clients. Another contributor is the fast-growing reliance on managed security service (MSS) providers, also called cybersecurity as a service (CSaaS). A key driver is lack of in-house cybersecurity skills and expertise. According to multiple accounts, more companies are outsourcing than insurcing their cybersecurity needs in a host of areas, particularly provider risk management, data protection, and identify and access management.

Cybersecurity risk considerations became paramount in all forms of ITO. ITO clients may implicitly or explicitly expect ITO and MSS providers to assume some responsibility for cybersecurity risk. In reality companies cannot outsource their cybersecurity liability. Reputation-wise, companies are liable for the security of their data and systems, no matter what. More importantly, laws simply do not allow firms to outsource regulatory responsibility. This means that ITO clients must still actively monitor, document, and manage their cybersecurity risk exposure. This is evident from corporate annual statements to shareholders (see Box A). The fact corporations disclose annually in financial statements cybersecurity concerns relating to ITO and other service providers is evidence of the importance of these concerns to security analysts and shareholders. Some corporations concede that these concerns compromise the reliability of their financial reports and drive them to re-insource outsourced IT services (see Box A, Example 1).

Box A: Excerpts from annual financial reporting statements

Example 1

Green Brick Partners, Inc. 2015 [10-K, Management—Internal Control Opinion]… our assessment of the effectiveness of internal control over financial reporting … identified the following material weaknesses

The **Company utilizes an integrated** Enterprise Resource Planning (ERP) **software system by a third-party service organization** … **the Company was unable to obtain a Service Organization Control ("SOC") 1 Type 2 report …** As a result, the Company was unable to conclude that its service organization maintained effective controls over its information technology environment to (a) prevent unauthorized database and application access, and (b) maintain effective security administration and appropriate change management for the application maintained by the third-party service organization … This resulted in an inability to rely on the accuracy and completeness of data and key application reports obtained from the application at the third-party service organization …

The remediation efforts … include the following: **Management will move to an ERP self-host structure that involves hosting and managing the Company's ERP software system and underlying infrastructure internally** … This will allow management greater flexibility and control to design, implement, and test the **information technology general controls over security access and change management** …

Example 2

GRAVITY Co., Ltd. 2017 [20-F, Management—Internal Control Opinion]
Certain of our mobile games and services were provided through servers located overseas and operated by a third-party developer and/or licensee. During 2017, we implemented **general information technology controls** which were operated by third party developers for the adequate preparation of financial statements and effective internal controls. We directly accessed these controls and evaluated their effectiveness as of December 31, 2017, as no **Service Organization Controls ("SOC")** report from the third party developers and licensees was available.

This paper discusses options ITO client organizations have to deal with cybersecurity risk. It starts by offering more background on cybersecurity risk in ITO and explicating the main challenges. These challenges point to trust in ITO providers as a key success ingredient. The paper proceeds to review three alternative ways to establish such trust (Akinrolabu and New 2017; Dhillion et al. 2017; Yuen 2008). Our review suggests that trust anchored in independent cybersecurity certification and market-based reputation mechanisms is emerging as a dominant model for client-provider trust. The paper elaborates on market-based trust and its feasibility in the

ITO context. The idea simple: just as firms that obtain cybersecurity certifications benefit from positive market reactions that create firm value, so do firms that experience cybersecurity incidents indicating failures to live up to their cybersecurity certifications suffer punitive market reactions that destroy firm value. This dual relationship is what could hold ITO providers accountable for cybersecurity risk. To illustrate the latter part of this relationship, the paper reviews a recent empirical study providing evidence that market-based trust could work in the ITO context. The paper concludes with a discussion of the promise and obstacles to market-based trust in the ITO context.

2 Background

2.1 ITO and Cybersecurity Risk

While ITO continues to be popular for its ability to make enterprises more agile and cost effective, the associated cybersecurity concerns have been growing and taking on a more urgent priority. An IDG study (2016) found security requirements to be the top outsourcing concern for nearly two-thirds of organizations interviewed. Another study by the world-bank reports that 40% of companies consider security to be a primary barrier to adoption of cloud computing services. The study indicates the top concerns in the following order: data privacy (49%), compliance (17%), access controls (10%), lack of transparency (6%), lack of trust (8%), accountability and shared responsibility (4%), and incident response (2%). Cyber security concerns are greater for smaller businesses that outsource more of their IT needs. Giving an ITO service provider almost complete control over a firm's IT systems and service could have fatal business consequences, considering that one data breach could have a detrimental effect on the firm's reputation.

Contributing to the growing concerns over cybersecurity risk is the recent trend of relying on MSS providers. MSS providers offer a wide range of security services, such as managed services for firewalls, intrusion detection, virtual private networks, security monitoring, incident management and forensic analysis, vulnerability assessment, anti-virus and content filtering services, etc. As of 2017, half of all firms (49%) outsource part or all of their cyber security functions (Klahr et al. 2017). Again, this is more common among small and medium size firms and in certain sectors, for example, finance or insurance (60%), and professional, scientific or technical services (69%). The drivers behind outsourcing cybersecurity needs include cost savings, access to a staff with highly specialized skills and expertise, and (perceived) liability protection. Liu et al. (2017), for example, provide evidence that universities which outsource their cybersecurity function have a lower likelihood of suffering data breaches. MSS providers, by virtue of scale economics, can afford to invest more heavily in human resources and proprietary software tools. Moreover, pooling information and data from multiple clients permits MSS providers to identify patterns and detect cybersecurity risks more reliably and then take suitable countermeasures. Despite these argued benefits, there is continued debate over whether use of MSS providers lowers cybersecurity risk. This is due to a series of related issues such as moral hazard, difficulty building trust between a client and its MSS providers, and other hidden costs.

In summary, the two modes of outsourcing involve substantial cybersecurity concerns. In one mode, firms outsource IT functions with an *implicit* expectation that ITO providers assume responsibility for cybersecurity risk surrounding the outsourced functions. In another mode, firms outsource cybersecurity functions with an *explicit* expectation that MSS providers assume responsibility for cybersecurity risk associated with failures of the outsourced functions. Nonetheless, in both outsourcing modes cybersecurity risk considerations remain paramount for ITO client firms. When organizations outsource IT and cybersecurity functions, their risk profile changes and becomes a combination of their risks and a subset of their ITO provider risks, leading to many unknowns (Cayirci et al. 2014; Chan et al. 2012).

2.2 Challenges of Cybersecurity in ITO

IT insourcing and IT outsourcing are both subject to cybersecurity risks, but many of the risks are exacerbated in the ITO context because of the main challenges summarized below.

Strategic Imperative. Cybersecurity is no longer an operational concern and rather became one of the most important strategic imperatives for the enterprise (Ali et al. 2014). Firms can no longer overlook cybersecurity considerations in their discussions of new products and services, customer relationship management, cybersecurity liability class act lawsuits, and so on. This reality makes cybersecurity and data privacy among the most challenging issues in ITO contract negotiations.

Opaque Supply Chains. ITO increasingly involves complex, dynamic, and non-transparent supply chains. Cloud computing, for example, is an ecosystem with many more points of access and higher potential for cybersecurity failures. Data could be scattered across multiple data centers and clouds provided or managed by multiple third-party suppliers. It is not possible for ITO clients to determine how data is treated without having enough visibility into ITO sub-providers, sub-providers' location, compliance records, and processes (Raj 2011). ITO supply chain transparency is the extent to which information about ITO players and sourcing locations is readily available to supply chain partners. How transparent is the ITO supply chain determines how feasible it is for clients to assess and control cybersecurity risk in ITO (Akinrolabu and New 2017). An evaluation of the transparency of 25 top CSPs based on their published information finds that most have very limited visibility into their operations and supply chains.[1]

[1] Aknirolabu and New (2017) compared of 25 CSPs (SaaS providers) on eight transparency features (Architecture, Technology/Partners, Datacenter location, Security features, IT-related compliance

Quantifying Cybersecurity Risk. Most companies lack data for reliably deriving their own IT insourcing cybersecurity risk exposure based on the frequency and damage-magnitude of cyber incidents they may suffer. In ITO, this situation is aggravated. Insufficiency of data on ITO providers' vulnerability to cyber incidents of different types as well as the frequency and damage magnitude of each type of incidents makes it impossible to assess cybersecurity risk exposure. In addition, since cybersecurity risk stems also from ITO providers' partners along the supply chain, the nature of risk is more diverse and evolves at a rapid pace. These factors complicate any attempt to reliably quantify cybersecurity risk exposure in the ITO context.

Liability Asymmetry. ITO providers seek to limit or disclaim their liability. They are concerned about not paying damages that are disproportionate to the revenue received. By contrast, ITO clients are concerned that providers may not have the same incentives to protect their data and systems.

Growing Regulatory Demands. Cybersecurity regulations are growing at a rapid pace in the US, the UK, the EU, and elsewhere, as illustrated in Fig. 1 (Gozman and Lesley 2019). Cybersecurity regulations impose disclosure and compliance requirements on firms, for example, a requirement that firms inform customers with breached personal data within *n* hours of the breach discovery. Ensuring regulatory compliance becomes daunting for ITO providers. Data and services may be moving across supply chain partners operating in different regulatory environments. In particular, rapid evolution of the cybersecurity regulatory environment adds to the frustrations and near impossibility of ITO and cloud computing providers to satisfy all laws applicable to global customers in different jurisdictions (Cayirci 2015). For this reason, some CSPs and ITO clients decide not to comply or avoid doing business in some jurisdictions.

3 Cybersecurity Risk and Client-Provider Trust

How can firms address cybersecurity risks in ITO? Most cybersecurity risks inherent in ITO are not likely to be mitigated contractually. This means that many ITO clients are, knowingly or unknowingly, accepting cybersecurity risk exposure (Chan et al. 2012; Gadia 2011). An argument frequently made is that managing cybersecurity risk in ITO requires client-provider trust (Akinrolabu and New 2017; Dhillon and Syed 2017; Yuen 2008). However, there are multiple perspectives on how to achieve such trust. Figure 2 labels these as the decision-theoretic, transparency-based, and market-based perspectives. As we will show, the market-based perspective is emerging as the dominant alternative.

---

certifications, Advertised Service Level Agreement (SLA), Disaster recovery/ business continuity, Monitoring/Support). The results show that: (1) the CSPs in vertical markets, such as the finance/ERP sub-group, scored the lowest points; and, (2) CSPs in the online workspace sub-group were found to be the most transparent.

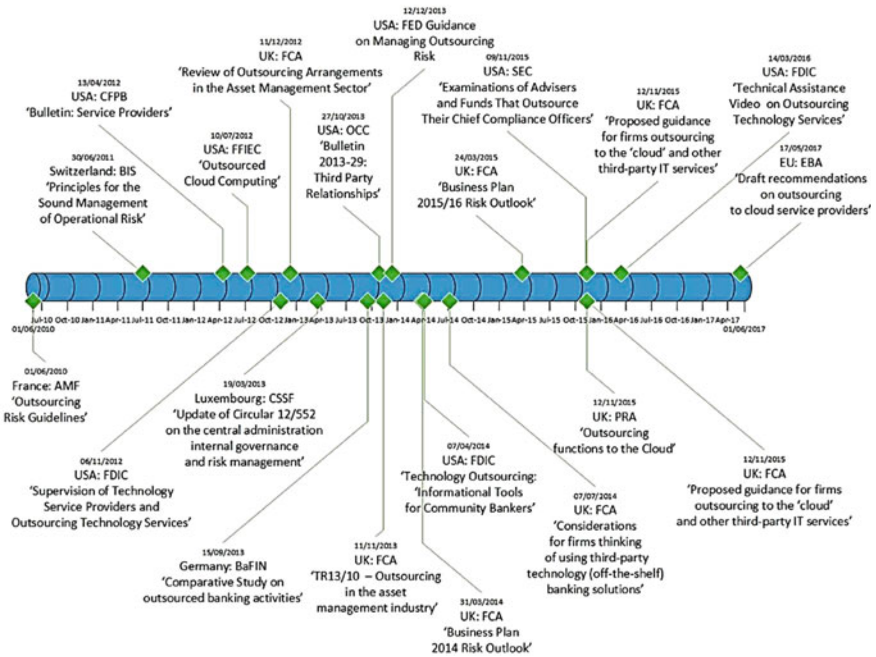

Fig. 1 Growing regulatory demands for cybersecurity (from [Gozman and Willcocks 2019])

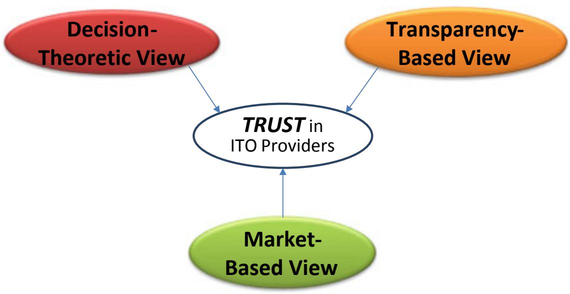

Fig. 2 Three perspectives on client-provider trust in ITO

3.1 Decision-Theoretic Perspective on Trust

This perspective is about ITO clients developing trust in their own decision to outsource, what, and to whom. This trust is anchored in a decision-theoretic calculation of risk exposure and a rational choice of how to reduce, avoid, or transfer this risk exposure. Such calculation require detailed data about: (1) the firm's and ITO provider's cybersecurity vulnerabilities, sources of threat, and assets subject to

Table 1 Data challenges with managing cybersecurity risk

Aspects of cybersecurity risk		Data challenges
Risk identification	Assets at risk	Clients have insufficient knowledge of their own internal operations Clients lack visibility into ITO providers' operations (information asymmetry)
	Potential threats	— "—
	Security Vulnerabilities	— "—
Risk analysis	Likelihood of events	Lack of statistical data on the occurrences rate of cybersecurity events ITO providers know more than clients (information asymmetry)
	Impact of events	Lack of statistical data on events' damage magnitude ITO providers know more than clients (information asymmetry)
Risk insurance	Coverage Specification	Unclear coverage limit, Exclusions and limited coverage, Low indemnity limits, and Overlap with existing insurance
	Premium Estimation	Correlated risks, (affordable) insurance premium

the threats; (2) the distributions of frequency and damage-magnitude of cybersecurity risk events; and, (3) contract terms and their pricing in the case of purchasing cybersecurity liability insurance (see Table 1) (Table 2).

Unfortunately, limited availability of these data restricts ITO clients' ability to quantify risk exposure and manage it using decision-theoretic strategies (Kopp et al. 2017). The risk reduction and risk avoidance strategies have little meaning when firms cannot reliably quantify ex-ante their cybersecurity risk exposure and their ITO providers' exposure.[2] In the strategy of risk transfer using cybersecurity liability insurance, the data challenge and magnified. First, pricing cyber risk and liability insurance policies is challenging because of the partial information insurers have in pricing risk and the fast-changing nature of that risk (Kopp et al. 2017). The combination of information asymmetries, difficulties in monitoring behaviors, and moral hazard problems that are typical of most insurance markets seem particularly binding in the case of cybersecurity risk. Second, cyber insurance policies typically do not cover indirect costs from cybersecurity incidents that manifest over the medium- to

[2]The risk reduction strategy involves taking steps that lower the underlying cost in case that risk events materialize (e.g., business continuity plans) and deploying security measures that reduce the likelihood of risk events occurrence (e.g., firewalls, encryption, security training, and role-based access rules). The risk avoidance strategy requires redesigning the way business activities are carried out and adapting or changing products and services. The risk transfer strategy involves the sale of risk to another party, primarily by buying cyber liability insurance in exchange.

Table 2 Competing cybersecurity risk standards

Standard	Audit focus	Strength	Sponsoring org.
Service Organization Control (SOC1/2)	Outsourcing service providers' financial reporting controls	Technology neutral	American Institute of CPAs
ISO 27001/2	Comprehensive list of 114 information security and data privacy controls	Technology neutral	International Organization for Standardization (ISO)
NIST 800-53 rev. 4	Security 18 control families meant to meet the Federal Information Security Management Act	Technology neutral	National Institute of Standards and Technology
Cloud Security Alliance (CSA)	Cloud-specific security	Dedicated to cloud security	Cloud Security Alliance
Cyber essentials plus	Five technical control categories: firewalls, secure configuration, user access control, malware protection, and patch management	Basic, technology neutral	UK-National Cyber Security Centre (NCSC)

long-term (e.g., reputational damage, lost customer relationship value, and increased cost of capital). Evidence suggests that policies typically impose restrictive liability exclusions and conditions that amount to serious coverage limits. This leaves clients with considerable risk exposure, primarily on the long tail of the cybersecurity loss distribution (see Fig. 3). Coverage limits are typically around $25 million, and do not exceed $300 million even for the largest financial firms (PwC 2015). Both these challenges are exacerbated in the ITO context. Blurry delineation of where ITO

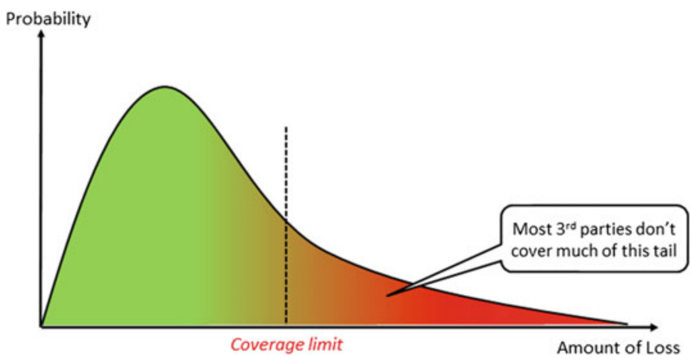

Fig. 3 Coverage limits and effective risk coverage

providers' responsibility for cybersecurity risk starts and ends makes it more difficult to agree on insurance coverage specifics and premium estimation.

3.2 Transparency-Based Perspective on Trust

If we cannot reliably calculate cybersecurity risk exposure, one alternative is to develop visibility into ITO providers' operations as a basis for trust. Supply chain transparency is the disclosure and transfer of credible, accurate and truthful information from one supplier to another through the chain of products and services down to the end user (Akinrolabu and New 2017). In ITO, transparency and improved visibility of the supply chain should allow ITO clients to verify that their trust in ITO providers is not misplaced (Vijayan 2015; Horvath and Agrawal 2015). Some hold that supply chains, especially those involving multinational companies, need to be inspected down to the second, third and fourth tiers. Others observe the dynamic boundaries of supply chains and difficulty of providers to coordinate beyond the 2nd tier (Wisner et al. 2008).

The reality is that businesses are not really making headway on visibility into IT providers' operations and their supply chain partners (Akinrolabu and New 2017). IT executives continue to cite supply chain visibility as a very high priority (Ali et al. 2014). Gozman and Willcocks (2019), in their study of ITO and regulation, identify lack of transparency and impaired control as one of three main challenges:

> … lack of visibility and control over SaaS arrangements is at the center of the regulatory dangers created through 'Shadow IT'. This challenge may also increase the difficulty of dealing with other challenges and increase the severity of breaches.

For true transparency-based trust, several unique challenges remain. Once challenge is the practical meaning of visibility—how should businesses approach and implement a visibility strategy? Akinrolabu and New (2017) identified three reasons for the vague information on supply chains: (1) ITO providers are not fully aware of their supply chain beyond the first tier; (2) many ITO clients are not capable, technically or cost-wise, to do an audit themselves; and, (3) how much should ITO providers be willing to share with their customers (i.e., ITO providers uncertain about the quantity and quality of technical information to share with customers). Another challenge is choosing where visibility is particularly important. In theory, transparency-based trust should work if every player in the supply chain has transparency into their directly connected parties and carries out its own audit of those parties. In practice, the efficacy of transparency-based trust hinges on the degree of information asymmetries between supply chain players. A third challenge is the cost to transparency. Even if the above requirements are met, more visibility means increased cost of doing business with providers that allow increased level of visibility into their security and supply chain practices (New and Brown 2012). More importantly, ITO clients may become more liable for cybersecurity risk the more they know about their ITO providers' operations and supply chains (New 2009).

3.3 Market-Based Perspective on Trust

The market-based perspective hinges on market mechanisms and independent evaluations establishing the reputation of ITO providers (O'Driscoll and Hoskins 2006). What constrains opportunistic behavior and exemplifies how markets self-regulate is one's reputations, or fear of its loss. Markets make such behavior costly. A chief function of markets is to provide information about service providers (sellers) through a variety of means, and the ultimate sanction is loss of reputation or brand.

Sometimes service providers hire trusted third-party experts to evaluate and certify their quality, where the evaluating experts' reputation substitutes for the service providers' reputation (O'Driscoll and Hoskins 2006). Examples are Underwriters' Laboratories (UL), which is hired to provide a seal of approval on products, and Dun & Bradstreet (D&B), which provides dependable credit information on thousands of businesses large and small. Both UL and D&B use evaluation standards that evolved through a trial and error process and have been institutionalized eventually. Evaluation standards are often established by regulatory bodies, especially when markets are too slow to develop reputation mechanisms and evaluation standards. This seems to be the case with cybersecurity risk and service providers, primarily because we continue to learn about the complex and dynamic nature of this risk.

For market-based trust to work, two factors must be addressed. One is balanced regulations. Lack of transparency increases demand for regulations, but at the same time serious information asymmetries between regulator and firms render regulations ineffective (O'Driscoll and Hoskins 2006). For example, the Security an Exchange Commission (SEC) requires public firms to disclose cyber incidents that materially damage their business, and yet over 60% of cyber incidents are not disclosed by firms (Coleman 2018; Croce 2019) but rather by customers, regulatory bodies, and whistleblowers. This example suggests that, without enforcement regulations can ineffective. In fact, regulations can become counterproductive, if they are excessive, as might be the case with the growing number of cybersecurity regulations (Gozman and Willcocks 2019).

Another factor that renders ex-ante regulation ineffective is failure to design effective evaluation standards (Kolstad et al. 1990). Going back to the SEC disclosure requirement of cyber incidents, even if firms do disclose when they suffer such incidence, is the SEC guidance specific enough on what details to include in the disclosure? We believe the answer is negative. We, hence, face the question: are there evaluation standards adequate for the ITO context, or are there regulatory frameworks for designing such standards? The mere emergence of multiple competing regulatory frameworks might be one strong indication that the answer is negative.

4 Market-Based Trust and Cybersecurity Certification

Of the three perspectives on trust, the market-based perspective is emerging as the dominant alternative in the ITO service delivery context. We elaborate next on this perspective. We focus on lessons from extant accounting research on the role of regulatory guidelines and evaluation standards, and the extent to which those lessons can apply in the cybersecurity risk and ITO context.

4.1 Financial Reporting Regulations and Control-Based Certification

Accounting is a field that studies extensively regulatory evaluation standards and market-based reputational mechanisms. Its focus is on internal controls over financial reporting processes public firms use to generate their annual financial statements.[3] This focus has grown rapidly upon the introduction of the Sarbanes-Oxley (SOX) Act in 2002, which was instituted to boost investor trust in public firms' financial reporting after several high-profile corporate scandals (e.g., Enron). SOX mandates firms to audit and disclose deficiencies in internal controls over financial reporting. Given this regulatory requirement, sponsoring agencies, such as the American Institute of CPAs, developed evaluation standards comprising lists of controls that need to be audited for SOX compliance. Firms' audits and disclosures of internal control deficiencies are certified by trusted public accounting firms (e.g., Deloitte, EY, KPMG).

Accounting research provides important insights particularly about the adverse market-based effects of internal control deficiencies. Secondary market data observed right after revelations of (disclosed) information about internal controls reflect how shareholders and security analysts react to the new information. It offers insight into a host of issues, including: penalties shareholders inflict on firms to hold them accountable for internal control deficiencies (e.g., drop in equity prices, increase in cost of capital, and higher audit fees), what types of internal control deficiencies matter more, and what role board corporate governance plays in ensuring the effectiveness of controls.

Overall, accounting research provides ample evidence that market-based trust works. Shareholders trust regulatory certifications by public accounting firms. And, firms work hard to avoid problems with their certified internal controls that would result in punitive market reactions. In sum, market-based trust holds firms accountable to their shareholders.

[3]*Internal controls* are "policies, procedures, practices, and organisational structures designed to provide reasonable assurance that business objectives will be achieved and *undesired events will be prevented or detected and corrected*" (ITGI 2007).

4.2 *Cybersecurity Regulations and Control-Based Certification*

Could market-based trust work vis-à-vis cybersecurity risk in the ITO context? The need and advantages are straightforward. Most ITO client firms are not capable or willing to evaluate and validate the IT security controls that ITO providers and their supply chain partners claim to have in place. At the same time, ITO providers need not, and do not want to be audited repeatedly by every client separately. Market-based trust and independent certifications of ITO providers' IT security controls seems a good way to fill-in the gap, as long as service providers' reputation can be established through visible controls the provider has implemented (ISACA and CSA 2015). Indeed, as evident from the sample excerpts from financial annual reports in Box A, third-party evaluation and certification of service providers' controls is becoming a viable option in terms of shareholders' trust in providers' financial reporting.

For cybersecurity risk, the focus is on certification of IT security controls in ITO providers' operational environment. *IT controls* are management and technical policies, procedures, standards and organizational structures prescribed to protect the confidentiality, integrity, and availability of a system and its information (ITGI 2007). It is common to distinguish two classes of IT controls (Bellino and Hunt 2007; ITGI 2007). *IT application controls* (ITAC) are embedded in, and apply to, individual IT systems. To ensure the reliability of application-level transactions, ITACs apply integrity and validity tests to inputs, data edits, and processing logic, among other things. *IT general controls* (ITGC), by contrast, are embedded in enterprise IT processes making up the IT management and production environment. As seen in Fig. 4, ITGCs "reside" on top of, and govern the portfolio of interdependent enterprise IT processes within which IT assets and IT services are developed, acquired, mobilized, deployed, maintained, operated, and delivered. As such, ITGCs apply to and affect all IT systems, IT services, and data assets. The *IT security controls* we are referring to are a subset of ITGCs focused on security, access controls, and data protection. They aim to provide assurance over the confidentiality, integrity, and availability of insourced and outsources systems and services.

Similar insights to those obtained by accounting research can and are starting to emerge regarding ITO providers' IT security controls. One study identifies IT security

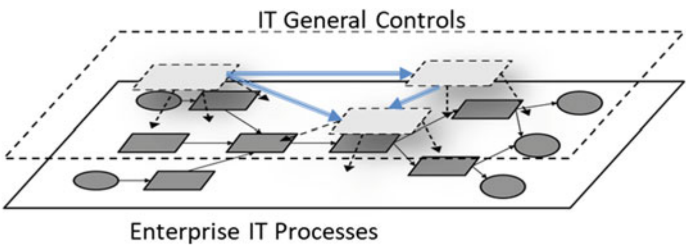

Fig. 4 IT general controls over a portfolio of interdependent enterprise IT processes

control deficiencies associated with data breaches in healthcare (Vasishta et al. 2018). Another study examines and maps cyber incidents in financial services to specific ITGCs deficiencies (Benaroch and Chernobai 2015). These studies demonstrate the connection between cybersecurity events and the existence of deficiencies in IT controls. Several other studies document a favorable stock market reaction to ITO providers announcing investments in certification of their IT security controls (Park et al. 2010; Kang 2014; Szubartowicz and Schryen 2018; Dean et al. 2019; Malliouris and Simpson 2019). These studies demonstrate the (positive) value shareholders ascribe to well-functioning, certified IT security controls. Lastly, a recent working paper studies empirically the ITGC deficiencies associated with cyber incidents and the negative market reaction to those cyber incidents (Benaroch 2018, 2019).

5 Market Value of ITGCs—Illustration for Cyber Failures

We proceed to use the study of Benaroch (2019) to illustrate that a prerequisite for market-based trust to work is met, namely: shareholders do care about deficiencies in ITGCs, of which IT security controls are a subset. Specifically, this section provides empirical evidence that market-based trust works in the context of cyber incidents in IT insourcing. It demonstrates the punitive stock market reaction to deficiencies in ITGCs that surface upon the occurrence of cyber incidents. As seen in Box B, firms routinely include in annual financial statements an assessment of their ITGCs, sometimes in association with cyber incidents and other adverse IT risk events they have suffered. This alone suggests the importance of information on ITGCs to security analysts and shareholders. On this basis, our illustration uses secondary market data to distil the value shareholders ascribe to ITGCs, particularly ITGC deficiencies, and their impact on firm performance, particularly firm equity prices.

Box B: excerpts linking ITGCs to cyber incidents

Example

Equifax, Inc. 2017 [10-Q, Note 5 in the Consolidated Financial Statements]

"…on September 7, 2017, we **announced a cybersecurity incident**. Our review of the circumstances and resulting impact on our internal controls over financial reporting (ICFR) identified **two significant *deficiencies in our IT General Controls environment*** … actions have already been and are being taken in the fourth quarter of 2017 to remediate these significant deficiencies."

The overarching theoretical assertion of the illustrations is that, first, deficiencies in ITGCs adversely affect market firm performance through shareholders' negative reaction to cyber failures and, second, this effect is negatively moderated by the IT competence level of the board of directors. The second part is motivated by the board

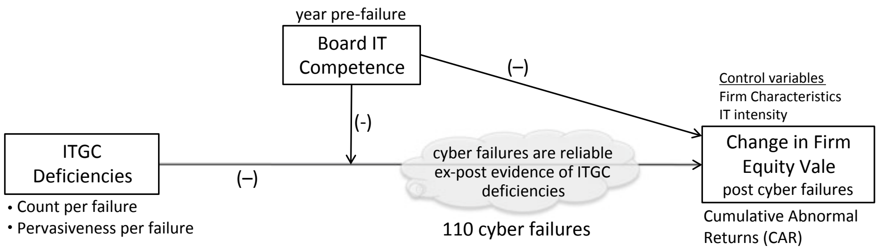

Fig. 5 Research model

of directors having the responsibility for monitoring corporate internal controls, of which ITGCs are a subset. Formally, the hypotheses tested are (see Fig. 5):

H1 Post-failure changes in firm equity prices are negatively associated with (a) the *count* of, and (b) the degree of *pervasiveness* of ITGC deficiencies associated with cyber failures.

H2 *Board IT competence* and the degree of *pervasiveness* of deficient ITGCs associated with cyber failures are complementary in their effects on post-failure changes in firm equity prices, such that changes in equity prices grow more negative when board IT competence is higher.

The hypotheses are tested using a sample of 110 cyber failures in public U.S. firms between 1992 and 2015. *Cyber failures* are defined broadly as internally or externally triggered events that accidently or maliciously compromise the confidentiality, integrity, or availability of data assets and/or functional IT assets that create, record, process, transport, store, and safeguard data assets (Benaroch et al. 2012). Based on public information (news releases, media articles, expert commentary, etc.) from a short window after a failure announcement, two IT-experienced individuals used an IT control framework called COBIT to identify and code *ITGC deficiencies* and underlying enterprise IT processes implicated in the failure. COBIT covers ITGCs over a total of 34 enterprise IT processes commonly found in IT's traditional responsibility areas (ITGI 2007). *Pervasiveness* of ITGC deficiencies is operationalized using a graph-based model of the input/output interdependencies among IT processes underlying ITGCs in COBIT. The model scores the criticality of each IT process to the proper functioning of the network of processes based on the notion of centrality from social network analysis. These criticality scores are specific to an idealized firm that implements all enterprise IT processes in COBIT. Adding up the criticality scores only for the IT processes underlying ITGC deficiencies implicated in a specific cyber failure provides a failure-specific pervasiveness measure. The remaining variables are defined following Benaroch and Chernobai (2017). *Board IT competence* is computed using corporate governance data in EDGAR, based on the percentage of board members with IT experience, on having a CIO on the board of directors, and on the number of board IT committees. Post-failure *changes in firm equity prices* are measured using an event study for a 4-day event window starting from the cyber failure announcement date.

The regression results are shown in Table 3. The variables for count and pervasiveness of ITGC deficiencies are highly correlated and so we have them in separate regression models. Model 1 includes only firm characteristics as control variables. Models 2–4 add the count, pervasiveness, and board IT competence variables, one at a time, showing negative and statistically significant variable coefficients and an increase in explanatory power (Adj. R-sq of 15.4% up to 20.1%). Models 5–8 add combinations of these variables as well as the interaction terms of count and pervasiveness with board IT competence. Again, these models show negative and statistically significant variable coefficients and increasing explanatory power (Adj. R-sq up to 25.1%). Consistent with the expectation, the interactions sharpen the cumulative negative effect in changes on post-failure equity prices. These results support both research hypotheses.

The takeaway from this illustration is that stock market investors do care about ITGC deficiencies. They care about the count and pervasiveness of ITGC deficiencies associated with cyber failures. This indicates that investors use public information to assess the ramifications, remediation difficulty, and persistence likelihood of ITGC deficiencies. Moreover, the results show that shareholders also care about the interaction between ITGC deficiencies and the board IT competence level. In fact, other research shows that, when the stock market penalizes firms for having ITGC deficiencies, those firms tend to respond by improving their board of directors' IT competence level, a factor that should improve the monitoring of ITGCs (Benaroch and Chernobai 2017). This reinforces the belief that market-based trust and reputational mechanisms do work in the IT context.

The lesson for the ITO context is that, market-based trust centering on IT security controls and their deficiencies seem to have adverse implications for ITO clients and ITO providers and, therefore, facilitate accountability for cybersecurity risk through stock market penalties.

6 Promise and Remaining Obstacles

The promise of market-based trust has increasing empirical support, but obstacles to adoption remain. Studies demonstrate that firms announcing completion of cybersecurity certifications, such as ISO27001, witness an appreciation of their stock prices (Park et al. 2010; Kang 2014; Szubartowicz and Schryen 2018; Dean et al. 2019; Mallliouris and Simpson 2019). The same way such certifications lead to positive market reactions that create firm value, so do cybersecurity incidents indicating failures of certified IT security controls would lead to punitive market reactions that destroy firm value. The study reviewed in Sect. 5 clearly demonstrates the latter relationship (Benaroch 2018, 2019). It is this dual market-based mechanism that should hold ITO providers accountable for cybersecurity risk.

However, there are also obstacles to widespread reliance on market-based trust. The rest of this section summarize the main obstacles, which center on the rate of cybersecurity certification by ITO service providers, the design and suitability

Table 3 Regression results

Dependent variable					CAR[0, + 2]				
	Exp. Sign	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Firm growth (Tobin's Q)		– 0.208*	– 0.171	– 0.169	– 0.199[a]	– 0.167	– 0.161	– 0.172	– 0.157
		(–1.93)	(–1.62)	(–1.59)	(–1.86)	(–1.59)	(–1.52)	(–1.65)	(–1.53)
Firm size		0.239[b]	0.216[b]	0.226[b]	0.262[b]	0.237[b]	0.249[b]	0.252[b]	0.240[b]
		(2.18)	(2.03)	(2.11)	(2.40)	(2.21)	(2.34)	(2.36)	(2.31)
Firm risk		– 0.199[a]	– 0.203[b]	– 0.159	– 0.196[a]	– 0.200[b]	– 0.156	– 0.187[a]	– 0.120
		(–1.98)	(–2.07)	(–1.58)	(–1.97)	(–2.05)	(–1.58)	(–1.93)	(–1.23)
IT intensity		– 0.160	– 0.166[a]	– 0.136	– 0.160	– 0.166[a]	– 0.137	– 0.164[a]	– 0.146
		(–1.58)	(–1.68)	(–1.37)	(–1.60)	(–1.69)	(–1.39)	(–1.69)	(–1.52)
Board IT competence	(–)				**–0.161[b]**	**–0.131[a]**	**–0.157[b]**	– 0.104	**–0.157[b]**
					(–1.84)	(–1.51)	(–1.83)	(–1.19)	(–1.88)
ITCD_Count	(–)		**–0.236[c]**			**–0.217[c]**		**–0.219[c]**	
			(–2.68)			(–2.46)		(–2.49)	
ITCD_Bet_Gephi_W	(–)			**–0.214[c]**			**–0.211[c]**		**–0.260[c]**
				(–2.39)			(–2.39)		(–2.95)
ITCD_Count ×	(–)							**–0.138[a]**	
Board IT competence								(–1.58)	
ITCD_Bet_W ×	(–)								**–0.225[c]**
Board IT competence									(–2.62)
N		110	110	110	110	110	110	110	110
R^2		0.185	0.238	0.228	0.211	0.254	0.252	0.272	0.299
adj. R^2		0.154	0.201	0.190	0.173	0.211	0.208	0.222	0.251

$N = 110$. Standardized beta coefficients; t statistics in parentheses. For directional hypotheses, significance levels are based on a one-tailed test,
[a] $p < 0.10$
[b] $p < 0.05$
[c] $p < 0.01$

of cybersecurity regulations and certification standards, and lack of theory of IT controls.

Optionality. While SOX compliance and certification is mandatory of all U.S. public companies, cybersecurity certification remains largely optional. Market-based trust could work only if every supply chain player is publicly traded, is cybersecurity certified, and requires a similar certification from its own supply chain partners. These requirements are not likely to be met as long as cybersecurity certification is not mandatory. However, there is evidence that players with some economic dominance might have the power to impose (mandate) certification standards that markets would recognize and adopt. For example, a recent study reports a favorable reaction of shareholders to ITO service providers investing in *Cyber Essentials Plus*, a certification program the U.K. government and National Cyber Security Centre mandate of firms bidding for government contracts involving the processing of sensitive and personal information (Malliouris and Simpson 2019).

Myriad of Regulatory Standards. A factor contributing to slow adoption of market-based trust is the myriad of cybersecurity regulations around the world (Yuen 2008).[4] The G7 has taken the first step toward standardized requirements for cyber risk. It developed a set of non-binding, high-level fundamental cybersecurity elements designed for financial sector private and public entities. Public authorities within and across jurisdictions can use the high-level elements to guide their public policy, regulatory, and supervisory efforts. In turn, the high-level elements are expected to be tailored by financial institutions themselves, to fit their operational and threat landscape as well as their legal and regulatory requirements. As we implied earlier, once regulations are instituted, they are operationalized and expanded into evaluation and certification standards by various sponsoring bodies. Sample standards for cybersecurity include *SOC1*/2, *ISO27001*, *NIST800-53*, and country-specific standards like UK's *G-Cloud* and Singapore's *MTCS*. Table 2 characterizes some of the better-known information security certification standards. All such standards seek visibility into service provider's IT security and data privacy controls aimed at ensuring the confidentiality, integrity, and availability of those providers' systems and services. ITO providers seeking participation in specific industry environments, such as cloud computing, are increasingly expected to adhere to specific cybersecurity standards. When their IT platform achieves certifications, it means that the platform has capabilities that meet specific security requirements. Unfortunately, the more standards exist, the harder it is for ITO providers to choose when and which exact certification standard(s) to pursue.

[4]The U.S. follows a sectoral law approach, where federal regulations on data protection are industry- or sector-specific. India is expanding sectoral laws to attain a more comprehensive data protection. The EU, by contrast, offers guidelines aimed at becoming a working multi-national standard (e.g., OECD Guidelines, EU Data Directive). The *EU Data Directive*, for example, prescribes eight principles for: (1) limiting collection and use of personal data, (2) access by individuals to their information, (3) accountability for compliance by data controllers (firms), (4) transparency of process, (5) security safeguards, (6) destruction or anonymizing of data no longer serving the original purpose for which it was collected, (7) and so on.

SOC1/2 Emerging Dominance. SOC1/2, Service Organization Control, appears to be more widely adopted by ITO and other service providers, probably because this standard is merely an extension of SOX (Weiss and Solomon 2016, p. 83):

> A goal of SOX is to maintain investor and public confidence through the accuracy of financial reporting. SOX essentially mandates establishing of adequate internal controls. Consider that many organizations outsource all sorts of activities that could have implications on SOX compliance. … Ensuring that adequate controls are in place is required regardless of whether the data is stored and processed in house or by an external party.

SOC1/2 is geared towards certifying financial reporting controls of publicly traded service providers. SOC1/2 is sponsored by the American Institute of CPAs with public accounting firms acting as the audit certifying bodies. SOC1/2 certification yields two common reports, depending on the client (ITO provider) desires and needs. SOC1 report informs auditors and shareholders about controls over financial reporting. SOC2 report informs knowledgeable users (e.g., clients, partners, regulators) also about controls for meeting information security handling objectives.[5] The big question is: Could it be that adoption of SOC1/2 is a lip service act vis-à-vis cybersecurity risk exposure? More comprehensive standards exist but one is yet to achieve dominance and broad market acceptance. Perhaps over time large government and industry bodies may use their economic power to impose standards that markets will adopt.

Overregulation and Cost. Instituting government regulation for market-based trust imposes significant costs to the detriment of shareholders, consumers, and taxpayers. This lesson comes from one relatively highly regulated area, namely: corporate governance oversight of financial reporting activities and SOX compliance (O'Driscoll and Hoskins 2006). SOX is estimated to have caused a loss of $1.4 trillion to shareholders in publicly traded companies that have decided to go private. SOX also increased the cost of issuing shares in the U.S., causing companies to switch their share issuing to London and Luxembourg. It is important to add that, ongoing SOX compliance is acknowledged to be costly, where up to 30% of the cost is attributed to auditing and monitoring ITGCs (IAA 2007). Can we expect similar costly effects of stringent cybersecurity regulations, for example, in the case of EU's Data Directive and other cybersecurity-related regulations?

Theory of IT Controls. There is no theory of IT controls (or internal controls) based on which cybersecurity certification standards can be designed, validated, and so on. As has been learned since the introduction of SOX in 2002, much of the work on frameworks of internal controls over financial reporting has been driven by accounting practice rather than academic research. The lack of theory on internal controls can be traced to this simple reality. It seems we are heading down the same

[5]The objectives are: *Security*—system is protected against unauthorized physical and logical access; *Availability*—system is available for business use and operations as required; *Processing integrity*—system processing is complete, accurate, timely, and authorized; *Confidentiality*—restricted information is protected and access is limited to authorized users; and, *Privacy*—personal information is collected, used, guarded, disclosed, and destroyed in conformity with the firm's privacy stated policy and generally accepted privacy principles issued by various standard-sponsoring organizations (e.g., AICPA).

road with IT security controls and cybersecurity certification standards, particularly the SOC1/2 standard and its sponsorship by public accounting firms. A theory of IT controls is necessary to address a host of fundamental questions, including: What is the universe of IT controls? How is it determined? What principles can guide the design of theoretically-sound cybersecurity standards (completeness, validity, internal consistency, and parsimoniousness)? What theory-based principles can guide the study of properties of IT controls? How can these principles help lower the cost of cybersecurity certification and the cost of compliance with cybersecurity standards? These are only a few of the open questions a theory of IT controls could help address.

In conclusion, market-based trust can ensure ITO service provider accountability for cybersecurity risk, as long as clients demand ITO providers to obtain suitable cybersecurity certification and as long as surfaced deficiencies in certified IT security controls have punitive market implications for ITO providers.

References

Akinrolabu, O., & New, S. (2017). Can improved transparency reduce supply chain risks in cloud computing? *Operations and Supply Chain Management, 10*(3), 130–140.

Ali S., Padmanabhan, V., & Dixon, J. (2014). Why Cybersecurity is a Strategic Issue: Is your business one hack away from disaster?" Bain and Company. (https://www.bain.com/insights/why-cybersecurity-is-a-strategic-issue/).

Bellino, C., & Hunt, S. (2007). *Auditing Application Controls.* The Institute of Internal Auditors (IIA).

Benaroch, M. (2018). Properties of IT control deficiencies at the root of cyber incidents: theoretical and empirical examination, in *Proceedings of the 12th ILAIS Conference.* Israel.

Benaroch, M. (2019). IT general control deficiencies and impact on firm IT capability and firm performance, in *Working paper, Whitman School of Management.* Syracuse University.

Benaroch, M., & Chernobai, A. (2015) Linking operational IT failures to IT control weaknesses, in *Proceedings of AMCIS'2015*. Puerto Rico.

Benaroch, M., & Chernobai, A. (2017). Operational IT failures, IT value-destruction, and board-level IT governance changes. *MIS Quarterly, 41*(3), 729–762.

Benaroch, M., Chernobai, A., & Goldstein, J. (2012). An internal control perspective on the market value consequences of IT operational risk events. *International Journal of Accounting Information Systems, 13*(4), 357–381.

Cayirci, E. (2015). Models for cloud risk assessment: A tutorial, in *Accountability and Security in the Cloud* (vol. 8937, pp 154–184) Berlin: Springer International Publishing. (http://link.springer.com/10.1007/978-3-319-17199-9).

Cayirci, E., Garaga, A., De Oliveira, A. S., & Roudier, Y. (2014). A cloud adoption risk assessment model, in *Proceedings—2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, UCC 2014* (pp. 08–913). (http://doi.org/10.1109/UCC.2014.148).

Chan, W., Leung, E., & Pili, H. (2012). Enterprise risk management for cloud computing. *Committee of Sponsoring Organizations of the Treadway Commission 4*.

Coleman, D. (2018). Nearly 65% of affected public companies did not report cybersecurity breaches to the SEC. *Audit Analytics Report.* (https://www.auditanalytics.com/blog/nearly-70-of-affected-public-companies-did-not-report-cybersecurity-breaches-to-the-sec/).

Croce, B. (2019). Majority of cybersecurity incidents go unreported to SEC, analysis finds. *Pensions and Investments.* (https://www.pionline.com/article/20190227/ONLINE/190229852/majority-of-cybersecurity-incidents-go-unreported-to-sec-analysis-finds).

Deane, J. K., Goldberg, D. M., Rakes, T. R., & Rees, L. P. (2019). The effect of information security certification announcements on the market value of the firm. *Information Technology and Management.* published online.
Dhillon, G., Syed, R., & Sá-Soares, F.de. (2017). Information security concerns in IT outsourcing: identifying (in) congruence between clients and vendors. *Information and Management* (54:4), 452–464.
Gadia, S. (2011). Cloud Computing Risk Assessment: A Case Study. *ISACA Journal* (4), 11–16. (http://www.isaca.org/Journal/Past-Issues/2011/Volume-4/Pages/Cloud-Computing-Risk-Assessment-A-Case-Study.aspx).
Gozman, D., & Willcocks, L. (2019). The emerging Cloud Dilemma: Balancing innovation with cross-border privacy and outsourcing regulations. *Journal of Business Research*, forthcoming.
Horvath, A. S., & Agrawal, R. (2015). *Trust in cloud computing, SoutheastCon 2015* (pp. 1–8). FL: Fort Lauderdale.
IDG. (2016). Data Centers in Flux: The IT Optimization Challenge, IDG Research Services. (https://www.insightcdct.com/getattachment/e900f48c-faa8-4d43-b9cf-07131e5cc713/Data-Centers-in-Flux-The-IT-Optimization-Challeng.aspx).
IIA. 2007. *Scoping Information Technology General Controls* (ITGC), The Institute of Internal Auditors.
ISACA & CSA. (2015). Cloud Computing Market Maturity, in *ANISACA Cloud Vision Series White Paper* (pp. 1–12).
ITGI. (20070. *COBIT 4.1 Framework*, IT Governance Institute, IL: Rolling Meadows.
Kang, H. S. (2014). An Analysis of Information Security Management System and Certification Standard for Information Security. *Journal of Security Engineering, 11*(6), 455–468.
Klahr, R., Shah, J. N., Sheriffs, P., Rossington, T., Pestell, G., Button, M., & Wang, V. (2017) Cyber security breaches survey 2017, in *Ipsos MORI Social Research Institute and the Institute for Criminal Justice Studies.* University of Portsmouth. (https://www.ipsos.com/sites/default/files/2017-04/sri-cybersecurity-breaches-survey-2017.pdf).
Kolstad, C., Ulen, T., & Johnson, G. (1990). Ex post liability for harm versus Ex ante safety regulation: substitutes or complements. *American Economic Review* (80:4).
Kopp, E., Kaffenberger, L., & Wilson, C. (2017). Cyber Risk, Market Failures, and Financial Stability, in *IMF Working Paper (WP/17/185), International Monetary Fund*.
Liu C.-W., Huang, P., & Lucas, H. (2017). IT centralization, security outsourcing, and cybersecurity breaches: evidence from the U.S. higher education, in *ICIS 2017 Proceedings*.
Malliouris, D. D., & Simpson, A. C. (2019). *The stock market impact of information security investments: The case of security standards*. Boston, MA: Workshop on Economics of Information Security.
NetDiligence. (2016). 2016 Cyber Claims Study. (https://netdiligence.com/wpcontent/uploads/2016/10/P02_NetDiligence-2016-Cyber-Claims-Study-ONLINE.pdf).
New, S. (2009). Supply chain traceability and product provenance: challenges for theory and practice, in E. Sweeney (Ed.), *Supply Chain Management and Logistics in a Volatile Global Environment.* Dublin: Blackhall Publishing Ltd, ISBN 9781842181775.
New, S., & Brown, D. (2012). The four challenges of supply chain transparency. *European Business Review*, 1–7. (http://www.europeanbusinessreview.com/?p=4082).
O'Driscoll, G. P., Jr., & Hoskins, L. (2006). The case for market-based regulation. *Cato Journal, 26*(3), 469–487.
Park, C.-S., Jang, S.-S., & Park, Y.-T. (2010). A Study of Effect of Information Security Management System [ISMS] certification on organization performance. *International Journal of Computer Science and Network Security* (10:3).
PwC (Price Waterhouse Coopers). (2015). Insurance 2020 and Beyond: Reaping the Dividendsof Cyber Resilience. Price Waterhouse Cooper Insurance.
Raj, S. (2011). Common Assurance Maturity Model, 1–2. (http://www.fstech.co.uk/fst/FSTech_Conference_2011/Common_Assurance_Maturity_Model_Raj_Samani.pdf).

Szubartowicz, E., & Schryen, G. (2018) Timing in Information Security: An Event Study on the Impact of Information Security Investment Announcements," Working paper. Germany: University Regensburg. (https://epub.uni-regensburg.de/37576/).

Vasishta, N. V., Gupta, M., Misra, S. K., Mulgund, P., & Sharman, R. (2018). Optimizing cyber-security program—evidence from data breaches in healthcare, in *13th Annual Symposium on Information Assurance (ASIA'18).* NY: Albany.

Verizon. (2017). Data Breach Investigations Report 2017. *Verizon Enterprise*. (http://www.verizonenterprise.com/resources/reports/rp_DBIR_2017_Report_en_xg.pdf).

Vijayan, J. (2015). Cloud Security: Transparency Is Crucial for Service Providers. (http://www.cio.com/article/2925773/cloud-security/cloudsecurity-transparency-is-crucial-for-service-providers.html).

Weber, R. H., & Staiger, D. N. (2014). Cloud computing: A cluster of complex liability issues. *Web Journal of Current Legal Issues* (20:1), 1–13. (http://webjcli.org/article/view/303/418).

Weiss, M., & Solomon, M. G. (2016) *Auditing IT Infrastructures for Compliance.* Jones and Bartlett Learning, LLC, an Ascend Learning Company.

Wisner, J. D., Tan, K. C., & Leong, G. K. (2008). *Principles of Supply Chain Management—A Balanced Approach*. Cengage Learning.

Yuen, S. (2008). Exporting trust with data: Audited self-regulation as a solution to cross-border data transfer protection concerns in the offshore outsourcing industry. *The Columbia Science and Technology Law Review (IX)*, 41–86.

Cloud Sourcing and Mitigating Concentration Risk in Financial Services

Daniel Gozman, Tim Machaiah, and Leslie Willcocks

Abstract With the increased adoption of cloud services, the resilience of cloud providers is paramount to not only the firm, but also to the stability of the financial sector. Method: We use a "mixed" method of research by using a combination of data ranging from the UK regulator regulatory data and information from the public domain, supported by interviews with technology risk specialists at the FCA. Conclusion: This research acknowledges the strategic role of information systems and recognises the key advantages that cloud providers can bring to financial firms. Most firms are keen to leverage these benefits, by adopting "cloud" into their future IT strategy. However, we find that this may lead to increased reliance on key service providers, thus leading to concentration risk. We also find that lack of supplier due-diligence and interoperability standards between providers can be significant contributors to this risk. This research then arrives at three aspects—availability concerns, cyber-attacks and contractual issues, which could constrain the ability of service providers to provision contracted services—that could potentially cause detrimental effects across the financial sector. Before concluding, we look at factors that could mitigate this risk and the increasing role of regulators, firms and service providers in this endeavour.

D. Gozman (✉)
University of Sydney, Sydney, Australia
e-mail: daniel.gozman@sydney.edu.au

D. Gozman · T. Machaiah
Henley Business School, Henley-on-Thames, UK

L. Willcocks
The London School of Economics and Political Science, London, UK
e-mail: l.p.willcocks@lse.ac.uk

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_14

1 Introduction

Information Systems (IS) plays a central part in financial services, allowing firms to not only perform effectively but also maintain their competitive advantage. This need to maintain competitiveness has increased since the financial crisis, wherein increased regulatory oversight and cost pressures have incentivized firms to leverage technology led services to deliver profitable revenue streams (Bolton 2008). Rapid advances in technology and processing power have also given rise to FinTech firms, which target niche areas of the traditional value chain (Gozman et al. 2018). By adopting the latest cloud-based solutions, new financial firms are able to take advantage of lower overheads that enable them to provide better services at a lower cost. At the same time, cloud based outsourcing arrangements have become increasingly scrutinized by regulators (Gozman and Willcock's 2019).

Traditional financial firms have historically relied on legacy interconnected platforms that hinder the ability to leverage the advantages of technological evolution. In this endeavor, financial firms have been looking at various forms of outsourcing strategies that can provide best of breed or even off the shelf products that can drive the firm's competitiveness (Hirschheim and Lacity 1993). By using applications that are hosted and managed by 3rd parties, financial firms can not only benefit from reduced costs, but also take advantage of the provider's capability to rapidly provision innovative solutions and platforms. These advantages could however lead to multiple regulated firms adopting a similar strategy which could lead to a situation, wherein the financial sector is reliant on a few key providers for delivery of critical services. This over-dependency can lead to concentration risk, wherein the failure of a service provider can lead to systemic implications from the viewpoint of regulatory stability (Hon and Millard 2016). In terms of addressing concentration risk, apart from the recognition of the risk by financial authorities such as the FCA[1] and IOSCO,[2] there is no evidence of a coordinated mechanism to develop frameworks that could mitigate this risk. One reason for this could be that the usage of cloud services for critical services such as "core banking" among the larger systemically important banks is not that widespread for reasons such as regulatory oversight and inertia. Nonetheless, the usage of cloud platforms by challenger banks and other FinTech firms is growing

This paper will first review relevant literature surrounding the concepts, benefits and risks of outsourcing. We then outline the cloud related drivers of concentration risk our study identified. Using these findings, we will highlight key actions that could be applied by managers, regulators and service providers to mitigate concentration risk. Finally, we offer some concluding comments to summarize the findings and key take-aways for managers.

---

[1] UK Financial Services Regulator—Financial Conduct Authority (FCA).

[2] Technical Committee of the International Organization of Securities Commissions (IOSCO).

2 Do Existing Studies and Industry Guidance Help?

In broad terms, outsourcing is the concept of using an external provider to deliver contracted goods and services. Although, there are a number of definitions of outsourcing in the literature, they all consistently refer to a common theme, referring to the usage of third-party agents to deliver contracted IT services. For example, Lacity and Hirschheim (1993) specify that "Outsourcing is the concept of conducting one or more organizational activities, using external agents". Quinn and Hilmer (1994) considers "outsourcing as an external acquisition of activities, including those traditionally considered an integral part of any firm, provided that they do not form part of the firm's core capabilities".

This focus on core capabilities is further emphasised in the literature with Manning et al. (2008) implying that "any function that is operationally non-strategic and not a core function should be outsourced to enable the organisation to focus on value-adding core functions". There are ambiguities in the literature on what is meant by a firm's core competency, with (Quinn and Hilmer 1994) considering core competencies to be those activities that the firm is continuously engaged in. On the other hand, Prahalad & Hamel (1990), state that "core competencies are those activities which provide long-term competitive advantage and must be kept in-house". In terms of the financial sector, IS enabled Services underpins the business strategy and can be a key driver of business growth (Ang and Straub 1998). It can hence be argued that IS in financial firms form part of their core competencies and hence, according to literature, should not be outsourced. However, according to (Braun and Winter 2005), financial services is mostly information based and involve repetitive tasks (such as payments) which can be operationalized; thus, making it easier to outsource those tasks. In addition, research by Willcocks and Lacity (1998) identifies that, outsourcing provides a great deal of flexibility in the deployment of IT services and makes it easier for financial firms to deal with business volatility. Financial firms can also benefit by using outsourcing to consolidate disparate systems and processes that arise as a result of mergers and re-organisations which occur quite frequently in this sector.

Over the last decade, there has been an evolution in outsourcing arrangements mainly categorised by the scope, business criticality and technology complexity (Verner and Abdullah 2012). With rapid advances in technology, new infrastructure provides such as Amazon and Google have introduced innovative business models such as cloud services, wherein applications are delivered as services, on demand, usually over the internet. Cloud service models include infrastructure (IaaS) or hosting provisioning, computing platforms (PaaS) and software as a service (SaaS). These services can be deployed either as a private cloud, which offers dedicated services solely for a firm; a public cloud, which is shared across multiple firms; and a hybrid cloud, which is a usually a combination of public and private cloud. Examples of public cloud infrastructure (IaaS) platforms include Amazon Web Services (AWS), Microsoft Azure and IBM's Bluemix. Furthermore, cloud-based application services such as Office 365 or Salesforce could either be installed in-house using a private cloud or delivered via SaaS over the internet.

It is also critical to understand that technology services could sometimes use "layers" of cloud services. For example, Temenos is a SaaS based core banking solution that counts Metro Bank and Schroders among its clients. The application platform in-turn is hosted on Microsoft Azure based IaaS solution. In such cases, customers usually have contracts only with the SaaS providers and not with the IaaS providers. Such an arrangement masks the underlying complexity such as the dependency of the SaaS service on other PaaS or IaaS cloud providers. Keahey et al. (2012) supports this argument by pointing out that cloud-based systems have levels of failure that are higher than traditional in-house services, because services are usually accessed over multiple layers of infrastructure with a high potential of network failure or delays.

There are multiple viewpoints in the literature with some arguing that some cloud uses should not be considered outsourcing (Millard and Walden 2013). However, there is a consistent view among regulatory authorities that usage of cloud computing is another form of outsourcing. This is consistent with the FCA's view which defines "outsourcing" as "an arrangement of any form between an investment firm and a service provider by which that service provider performs a process, a service or an activity which would otherwise be undertaken by the investment firm itself" (FCA 2016).

Outsourcing provides an organisation with benefits as well as risks. According to the literature, outsourcing helps financial firms offload the burden of legacy systems and enables them to rapidly deploy services to cater to varying business requirements (Hirschheim and Lacity 1993). Technologies such as cloud computing provides a flexible and scalable technology platform at lower costs (Armbrust et al. 2009). Outsourcing can also help firms reduce the risk of obsolescence or making frequent investments in new technologies. Kakabadse and Kakabadse (2000), suggests that outsourcing enables firms to capitalise on service provider's innovations and professional expertise. In terms of cost savings, Myllykoski and Ahokangas (2013), infers that outsourcing could help shift the cost structure from capital expenditure to operational expenses thus helping firms optimise their IT spending and improve overall agility. This agility could be in the form of capacity or flexibility which enables firms to switch suppliers when more advanced and cost effective technologies become available. (Quinn and Hilmer 1994). With respect to risks of outsourcing, most of the literature is around operational, governance and regulatory risk. López (2004) refers to two important challenges that outsourcing presents to financial firms. The first is the concern regarding confidentiality and protecting the privacy of customer's financial information; the other is the relatively high degree of regulatory requirements imposed on financial firms, which force firms to assume full responsibility for outsourced functions.

Another is concentration risk. when a limited number of outsourcing suppliers provide services to multiple regulated firms, it could lead to concentrated operational risks that may pose a systemic threat (Basel 2005). With 89% of banks using at least one cloud application (Capgemini 2015), this sentiment is also echoed by regulators across Europe, who raised the issue of a systemic failure across pan-EU banks who are reliant on a common global service provider.

As a mitigation strategy, the proposals in the literature include adequate contingency planning and ongoing regulatory supervision and risk assessments. Basel (2005) states that a combination of regulatory and market influences are necessary to address risks that arise due to massive reliance on external service providers. Another solution proposed by Armbrust (2009) is to use multiple cloud providers, without providing additional details on whether this is multiple providers per service – which could have significantly impact cost and thereby impractical. In addition, interoperability between cloud providers could be a challenge that needs to be addressed with common frameworks and technology standards. New regulations such as the European Unions' General Data Protection Regulation (GDPR), could also create an entry barrier to smaller firms, thus not only restricting competition, but also increasing a firm's dependency on large providers that could potentially create a situation, where cloud providers become too big to fail.

In summary, there is an acknowledgement that, some form of concentration risk is inevitable as financial firms look to consolidate legacy systems and look for improved efficiency and economies of scale (Hon and Millard 2016). However, there is a considerable gap in the literature regarding the true extent of the problem. One reason for this may be that concentration risk is not that widespread, because many financial firms are still using internal systems for critical services. This research tries to brideg this gap, by analysing the available data to identify sources and impact of concentration risk and determine mitigating actions that need to be considered, before the usage of cloud becomes widespread in financial systems.

3 Findings: Drivers of Concentration Risk

Failure of a single service provider that is used by many financial firms could potentially impact the stability of financial services. The complexity of this risk stems from the fact that 3rd party solutions could be "layered" with key dependencies on other providers and the wider network/internet infrastructure. For instance, Metro bank, Banco capital and Tandem Bank among others use Temenos as its banking platform. Temenos is hosted in Microsoft Azure which is also used by Clear Bank for infrastructure hosting. On the other hand, HSBC and NatWest among many others use Monitise (hosted on IBM) for digital banking. Lloyds Bank will also soon rely on IBM to offer critical services. In a hypothetical scenario, if either Microsoft or IBM suffers an availability, cyber or contractual issue, it can potentially affect all the downstream firms at the same time with an exponential increase of customer harm as shown below in Fig. 1.

Further examples of supplier concentration include Fiserv's Agility core banking solution used by Tesco Bank and Think Money among other firms. Dubai based Global Processing Services (GPS) is used by Monzo, Revoult and Starling for processing payments. Most the large retail and investment banks are reliant on FIS to

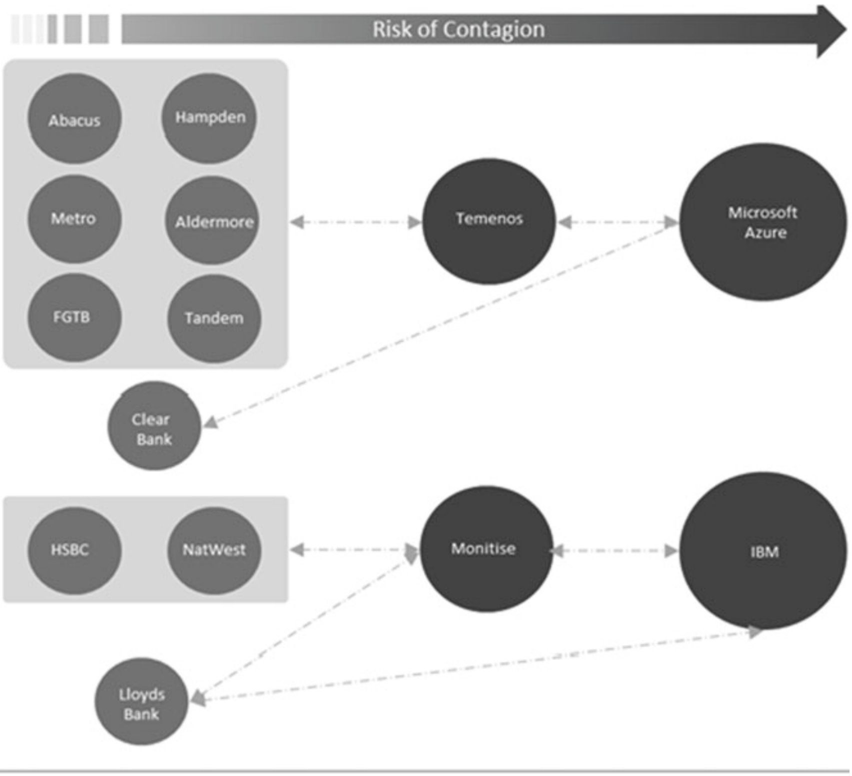

Fig. 1 Illustration of concentration risk

run its core banking and investment platforms. Ohpen (SaaS Platform—hosted on AWS), is used by Robeco (Global Asset Manager) and Robien (Dutch insurance firm) to administer their investment funds, insurance and savings account platform.

It is evident from our data analysis that many financial firms are in the process of evaluating cloud providers, with the intention of transitioning their services in a phased manner. These findings were also supported by most of our informants who supported this strategy since it would enable firms to have access to better resilience, scalability and security to deliver services effectively. According to a one of our interviewees, *"Greater consolidation means that the providers have greater resource to improve the resilience of their systems. Firms get access to world class infrastructure and processes at a reasonable cost, thus improving the quality and security of services delivered"*

However, there was a universal concern outlined by our interviewees that firms may not have conducted appropriate due diligence in selecting their suppliers and ensuring that necessary operational infrastructure is in place to ensure the confidentiality, integrity and availability of data. There were also concerns about interoperability and consolidation of technology providers that could contribute to concentration risk. Our study also identified managerial challenges arising from availability and performance issues, cyber-attacks and contractual problems. The drivers of concentration risk we identified are described in the following sections.

3.1 Driver 1—Increased Adoption and Reliance on 3rd Parties

Regulated financial institutions have historically relied on internal systems to deliver their core services. The reluctance to hand over responsibility to 3rd parties stems from the highly regulated nature of this industry, wherein system outages and data breaches could result in damages to the firm's reputation as well as incurring penalties from the regulator. However, cost pressures and changes in user behaviour such as access to on-demand services have prompted traditional institutions to look at more efficient and cost-effective means of delivering their core services. In the words of a technology risk specialist, *"Banks may find that their existing datacentre's, lack the processing capability required for compute intensive data analytics, parallel batch processing and on-demand real-time banking transactions using digital channels."*

In retail banks for example, there is a shift towards services that respond to user requests in real time especially with the advent of mobile applications. With this in mind, Lloyds Bank signed a 10-year cloud transformation agreement which is one of the largest deals within the financial sector (IBM 2017a). According to Morteza Mahjour, group CIO, *"The deal with IBM will allow the bank to become agile, scalable and provide up to date technology on demand"* .

Across the continent, Danske bank, a large Danish bank uses an IBM cloud-based solution to deliver a digital banking platform to over 3 million customers across Europe (IBM 2017a). This strategy was adopted to cater to growing demand of consumers being able to perform real time banking transactions.

HSBC has recently announced that the bank is adopting a cloud-first strategy to move away from legacy infrastructure to better support consumer demands. The firm has already begun trialing Google's public cloud to improve data analytics. In addition, the firm is working with big cloud providers to determine a hybrid approach to serve its 37 million customers (Dobinson 2017).

This shift is also clearly visible in the asset management sector, wherein the effort required to process data has increased substantially, leading asset managers to look for more sophisticated solutions. Even though most of the asset managers continue to use internally developed systems, there are numerous 3rd party providers offering solutions in this space, especially in areas of asset pricing, risk models and analytics, order management and execution systems. Drivers of this shift are regulatory pressures, the need to support legacy technology and cost pressures in an uncertain environment. In addition, the costs of the changing existing systems to comply with new regulations such as Retail Distribution Review (RDR) can be prohibitively expensive. For instance, once of the consequences of RDR is the unbundling of fees; a feature that some legacy systems are unable to cope with the new charging structures.

In the insurance sector, increased competition has increased the shift towards an internet and electronic trading model. This has incentivized brokers to use 3rd party software providers such as SSP to increase efficiencies across insurance distribution and bring additional capabilities for their customers.

Although, most of the firms cite scalability and resilience as the main reasons to adopt cloud services, regulators are not wholly convinced by this argument and believe that cost pressures play a big part in the firm's decision-making process. Traditional financial firms are faced with cost pressures due to declining margins, extended competition and increased regulatory oversight. The findings of the Dear Chairman Exercise (DCE) also emphasize this as a generic observation across all the participating firms, *"Firms appreciate that the use of third parties to deliver IT services is growing in scale and complexity. However, the criteria applied for review of third parties is often skewed by financial parameters rather than fully understanding the criticality of the supplier in supporting good customer and market outcomes."*

These cost pressures have incentivized firms to look at cost-effective solutions to convert capital expenses into operating expenses. Tesco Bank for example, found that an on-premise solution could cost around £3000 in capital expenses, compared to £66/month when the same solution was hosted on AWS (Cloud Pro 2017). These incentives however can lead to a perception that outsourcing is a cost cutting exercise that could be detrimental to service quality. According to a technology risk specialist, *"It is of paramount important to ensure that decisions to use 3rd parties are consistent with the institution's strategic plans and corporate objectives approved by the board of directors and senior management."*

3.2 Driver 2—Emergence of FinTech's

FinTech firms often focus on specific process or technology to disrupt some of the lucrative components of the financial value chain. Smartphone and technology adoption by the "Millennials", coupled with their expectation of instant gratification, convenience and security, have made traditional banking and payment systems archaic and unacceptable.

Since 2012, the Bank of England has granted licenses to 21 new providers aiming to rival the big four banks and in 2016, Atom—a digital only player, was launched (Intelligent Environments 2016). In retail and wholesale lending for example, Peer to Peer or P2P firms such as Ratesetter and Zopa used data analytics to develop the ability to lend money between individuals without bank involvement. In the investment space, robo-advisors are making inroads into traditional advisory services by transforming delivery of service and undercutting costs. By adopting automated analytics and customer identification mechanisms, such platforms can significantly reduce both the time and costs for customer on-boarding and retention.

Our informants believe that the viability and success of FinTech firms have been partly attributed to the technical advantage offered by cloud-based solutions, that are agile and can scale-up based on demand. This means that such firms can adapt to latest trends and technologies without needing to undergo expensive and time-consuming changes to legacy infrastructure. According to one technology risk specialist who is involved with FCA's Sandbox initiative, *"Adopting cloud-based solutions is the only way that mainstream financial institutions can compete with Fintech firms who rely on scalable and cost-efficient cloud solutions to analyse vast amounts of data in real time."*

Metro bank for example, uses T24, a SaaS based platform offered by Swiss based Temenos. The service offered on a "pay-per-use" basis enabled Metro bank to deploy the solution within 9 months (Temenos 2013). Tesco bank uses Fiserv's Signature software as its core banking platform along with AWS to host many its applications (Finnegan 2015). Tesco completed the transition to AWS in 8 months, with the new solution offering considerable cost benefits both in terms of initial investment as well as enabling faster time to market. Underlining this achievement Tesco Bank's head of transformation, Allan Brearley, quotes,

> The adoption of cloud technology allows us to respond to the needs of our customers more quickly and efficiently, while also offering the security standards our customers and regulators rightly expect from a financial services provider. (BBC News 2016)

Citing similar reasons, Robeco a large international asset manager adopted the SaaS based solution OhPen as its master platform across product offerings (Ohpen 2012). Instead of building its core banking application from scratch, Atom bank selected a commoditized SaaS based platform from FIS. This approach was also followed by Tandem bank and Think Money which aims offer personal current accounts with unique functionality for budgeting and personal fiscal management (Fiserv 2017). Saxo bank on the other hand as opted for Oracle Flexcube Managed Cloud Services for its new banking marketplace solution. The firm expects this solution to enable immediate transfers to bank accounts throughout the world at lower costs (Brusnahan 2015).

Apart from the challenger banks, other FinTech's such as "Robo Advice" firms also use major cloud providers to host their platforms. Nutmeg for example, runs the firm's operations on virtual private cloud on AWS. According to Nutmeg's CTO William Todd (IBS Intelligence 2015), *"This approach gives Nutmeg 'scalability and reliability through automation' as well as the 'overall control needed to meet all the regulatory criteria'."*

In a survey by PWC (2016), Fintech firms were highlighted as a threat to profit margins by 67% of global financial services with 59% of the firms acknowledging loss of market share. One of the key ways in which Fintech firms support the margin pressure point is by adopting cloud-based solutions that not only decreases upfront costs but leads to reduced ongoing operating costs. This adds additional pressure on traditional financial firms who look to adopt a similar approach in trying to compete with Fintech firms.

3.3 Driver 3—Complex Supplier and Subcontractor Management

Due to the interconnected nature of cloud providers, financial firms may not be aware of 3rd parties linked to their existing suppliers (subcontracted) that make up its services value chain. Due to this, firms may find that they may have an unacceptable level of reliance on certain suppliers. This concern was echoed by one of the interviewees, who quoted, *"Due to the effort and costs involved, firms are usually inclined to perform a supplier based only on their critical services which usually is around larger firms. This is because, firms do not think that the effort involved is commensurate to the loss incurred by smaller firms"*

Without a detailed value chain analysis, firms may not have the information to adequately spread their counterparty risk. This could also increase the risk of cyber-attacks, since the security of a firm is only as good as its weakest link. In addition, the DCE report also identifies shortcomings in this area, *"In majority of cases, third party's' services, whilst monitored operationally through service level agreements are not subject to end-to-end reviews of their control environment."*

Firms may also lack the legal and technical capability to ensure supplier contracts consider factors that could lead to concentration risk. These could include factors such a geographical concentration, reverse supplier concentration and BCP/DR end to end testing. Since interoperability between providers is still in the initial stages, financial firms need to understand up-front what the exit strategy will be, and build these into the service's design and cost, during the due diligence phase.

One of the main reasons that firms find cloud service attractive is that it is potentially reduce the risk of "single point of failure". Platforms such as Microsoft Azure are designed to run on multiple virtual systems across geographical locations thus aiming to ensure that the continuity is maintained during hardware, software or other environmental disasters. However, the management of a cloud service by a single provider itself creates a "single point of failure" for the clients of this service. There is a risk that the service provider may include punitive contractual amendments, change pricing structures or even go out of business which could cause unintended consequences to the clients of this service.

The ability of the 3rd party to deliver services may also be impacted by compliance and legal issues which arise when for example, the provider, falls foul to UK/US laws and regulations or gets involved in expensive legal issues. In addition, cloud providers may not be transparent about the geographical countries and mechanisms used to process data which could impact a firm's data protection responsibilities.

According to our informants, it is not only the increased dependency on 3rd parties that can lead to risk of concentration. Factors such as reverse concentration occur when a client represents a very large portion of the 3rd parties business. In this scenario, any dramatic changes in the business volume caused by market conditions or mergers and acquisitions can undermine the viability of the service provider. Such reverse concentration risks could crystallise in multiple layers of the service value chain at the subcontractor level thus increasing the complexity in risk identification.

3.4 Driver 4—Availability and Performance Issues

The major selling point of cloud services is that, they build on resilient infrastructure that offers high availability and reliability. This enables several of the major cloud providers offer very aggressive service levels—for example, Rackspace offer 100% uptime and AWS offers 99.95%. Nonetheless, like any IT organization, cloud providers are also susceptible operational and environmental issues, such as application bugs, spikes in consumer demand, technology failures and human error.

According to the incidents reported by firms: From Jan 2016 to July 2017, 7 incidents were caused by cloud providers which impacted multiple regulated firms with one particular SaaS provider being responsible for three of them. In addition, 4 additional incidents were caused by managed service providers with just one provider being responsible for 3 of them. In total, seventeen incidents were caused by IS service providers with most of them attributed to technical issues. Indeed, the number of incidents may seem minor when compared to the total of 257 technology related incidents reported during this period. Nevertheless, the unequivocal response among our respondents is that numbers don't count. To quote one of technology risk expert, *"Due to consolidation among tech providers, it just takes one significant incident to get the banking sector to its knees. If firms are not prepared for this eventuality, it could trigger a contagion affect that could ultimately lead to significant harm to customers."*

To put this in perspective, it takes just one significant outage at AWS to cause disruption to multiple firms at the same time. Moreover, since 2011, AWS suffered 4 outages, Microsoft's Azure platform had 3 disruptions and Rackspace had an 11 h disruption in 2014. The AWS outage that occurred in Australia in 2016 impacted multiple online financial services, with many customers unable to use their cards or ATM's. More recently in 2017, AWS endured another outage which impacted many firms such as GitHub, Adobe, Citrix, Salesforce and Autodesk among others. The impact of the outage, which was reportedly caused by "human error", caused companies to lose an estimated $150 million and even Amazon itself was unable to access its dashboard to provide updates. It should be noted that, despite this incident, AWS still reported meeting its target of 99.99% service and data availability.

Similarly, in 2016, SSP Worldwide, which is major insurance SaaS provider, suffered an extended two week outage that left 40% of insurance brokers unable to access its Pure Broking Platform (ComputerWeekly 2017). Even after a week, SSP were unable to provide recovery timelines thus adding to the uncertainty. This lack of communication and support is echoed by our informants with one interviewee quoting, *"Unless you are a very big player with an expensive support contract, smaller firms are usually in the back of the queue for support during an incident. Without dedicated contracts, such firms need to rely on standard Service Level Agreements (SLA's) wherein even responding to support requests can take 24–48 h."*

This research acknowledges that systems do fail, however under the absence of appropriate due-diligence, recovery and business continuity procedures, these failures can easily lead to widespread disruption.

3.5 Driver 5—Cyber Attacks

Access to cloud services depends on the availability and performance of the wider internet infrastructure such as webservers and domain name (DNS) servers. Although these components are highly resilient with multiple gateways, they are vulnerable to cyber-attacks which can potentially cause widespread disruption. Cyber-attacks are not limited by geography and can target facilities anywhere in the world. For example, cyber criminals may target a service provider production and backup facilities simultaneously rendering both inoperable.

According to Bisong and Rahman (2011), cyber criminals view cloud providers as an important frontier to attack. This is because it serves as a repository for valuable information from different companies and the attack only needs to target the weakest link in the supply chain to get access to this information. This importance of "value chain" due-diligence was further emphasised by our respondents with one technology risk specialist stating, *"Most firms are worried only about the top 10% of their outsourcing relationship usually based on the amount of money spent. However, this may not be reflective of the end to end risk profile; as the firm may be spending millions with a major service provider, however it may be a small 3rd party with immature controls that's holding key customer data that gets targeted by cybercriminals creating an significant risk into the firms service value chain"*.

This need to perform a supplier-based value chain analysis was also highlighted as part of the DCE recommendations, *"We recommend that the vendor management framework considers smaller suppliers who are nonetheless critical to the bank's operations and its clients."*

The impact that smaller firms can have on security is evident in the case of Epsilion, which is a customer engagement service provider used by a number of financial firms such as JP Morgan, Citibank and Barclays among other 8 other non-financial firms. Epsilion reported a data breach which enabled cyber criminals to steal sensitive customer information across all these firms. With all these firms using the same service provider, they were all connected by the same security infrastructure with only one break-in required to gain access to customer lists of 12 companies. In retrospect, if each of these firms had ensured that Epsilion was subject to stricter security controls, the damage could have been avoided.

4 Guidance for Managers: Mitigating Concentration Risks

It is important to understand that a "cloud service" is technically not that different from an internally hosted system. It is still based on hardware and application software that firms have dealt with for decades. When firms outsource the hosting and management of their platforms, they are still accountable for the availability, performance, security and resilience of their services (Gozman and Willcocks 2019). A key observation is that, usage of cloud providers should be in line with firm's strategic objectives and should be adopted after appropriate due diligence and oversight. Regarding mitigation of concentration risk, our respondents strongly felt that firms should have stronger supplier management capabilities along with robust business continuity plans to recover from disruptions. In addition, there was also consensus among the respondents that, firms and regulators should do more to coordinate standards and guidelines around interoperability and reporting. These findings are distilled into 'Mitigation Principles to reduce risks and are described in the following sections.

4.1 Mitigation Principle 1—Supplier Due Diligence

A well written legally binding contract can go a long way in reducing risks of non-performance or disagreements regarding service levels, scope and nature of services provided. It is important to ensure that the firm's legal team understands the vocabulary related to cloud technologies and can appreciate how service level definitions can vary by service provider. This feedback is also echoed in the DCE findings, wherein all the firms had either 3rd party risk management or DR/BCP capabilities identified as areas of improvement. The DCE report makes the following general recommendations regarding these observations, *"The security and resilience aspects of captive and third-party outsourcing arrangements that support retail economic functions should be assessed more comprehensively through robust vendor risk assessments and independent internal audits."*

Mainstream cloud providers such as Microsoft Azure and AWS design their service-levels around the needs to the majority, where the loss due to interruption to services is offset by the considerable savings in infrastructure costs. Typically, such providers advertise availability service-levels ranging from 99.5% to 100% and financial firms need to conduct appropriate due diligence to understand the meaning and assumptions in these availability figures. For instance, the availability figures could refer to the number of time intervals within a billing cycle during which the services are not "up" for the entire interval. For example, if a provider specifies an availability interval of 10 min and the service was not functional for 9 min, the provider still reports 100% availability. Although, firms may seek to build systems that are 100% resilient, it may not be practical or cost effective to do so as resonated by one of our informants, *"Financial firms such as retail banks design their internal* *systems to an availability target of five nines which relates to just a few seconds of unplanned outage per year. Although, cloud providers may be able to meet this target, it may be may not be cost effective which needs to be considered during the decision-making process".*

The firm should be aware of the provider's obligation during extended outages, support priorities and the extent of compensation that the providers can be held legally liable for losses endured. For example, standard contracts from 3rd party providers offer limited financial guarantee and the client bears the liability for failure. During the SSP outage for instance, brokers had difficulty claiming compensation for losses since there were no contractual provisions for business continuity planning. In addition, clients of financial firms have privacy related rights that govern confidentially and integrity of personal data and the ability to erase this data when required. It is important that firms understand the geographical distribution of client data and ensure that cloud providers are contractually bound to protect data and client confidentiality.

4.2 Mitigation Principle 2—Recovery and Business Continuity

The distributed nature of cloud computing could instil a sense of false confidence among financial firms, leading to complacency when thinking about business continuity. For example, during the FCA's engagement with the one of the firms, the CTO mentioned, *"We use a cloud based SaaS solution and do not find it logical to plan for DR or BCP since the solution has built in resiliency and will never fail. This is because regular snapshots of our instance are saved and these can be restored at a short notice when necessary".*

Arguments like these fail to consider situations wherein services may be interrupted due a wide variety of reasons, ranging from natural disasters, operational disruption, cyber-attacks, financial constraints or even failure of business continuity plans during operational disruptions. These shortcomings were also highlighted within the DCE report, which made the following observation regarding a large retail bank, *"Lack of proven end-to-end DR capability presents a significant risk. Ahead of the [supplier's] strategic solution, there is no short-term viable plan to mitigate effectively the conduct and financial system integrity risks. We recommend that the satisfactory completion of the actions from the IT remediation programme combined with the findings from the [supplier's] discovery phase is ensured to provide assurance to the Board in relation to the bank's overall recovery capabilities".*

During the SSP incident for example, brokers without a viable BCP/DR plan had to not only endure two weeks of lost business/revenue but also suffer loss of reputation among their customers. Firms should ensure that they are aware of the service provider's capability to respond to disasters and the contracted recovery time/point objectives. These competences need to be considered during the initial due diligence process, contract negotiations and also take into account business continuity processes when contracts are terminated. These contracted capabilities should be tested regularly and should form part of the firms overarching disaster recovery and business continuity plans. Beyond physical and cyber threats, financial pressures can lead service providers to make decisions not to invest fully in appropriate security controls or resilience measures that would facilitate continuity of operations. To mitigate this risk, firms should explore alternative solutions such as using redundant cloud providers or using hybrid solutions that firms could fall back on during operational disruptions. If the plan involves using an alternative provider, the financial firm should ensure that the provider has sufficient capacity in space, systems, and personnel to deliver the service effectively.

4.3 Mitigation Principle 3—Regulatory Influence

The role of information systems in financial services has transitioned significantly over the last decade and now forms an integral part of a firm's core competency. The extent of change is so dramatic that many of the financial firms are now sometimes referred to technology companies with a banking license. With excessive reliance on 3rd parties to deliver their technology strategy, it is of utmost importance, that there is appropriate standards and governance around technology providers that can enable both firms and customers to confidently adopt these services.

At present the regulatory oversight is around accountability, with the obligation resting with the regulated firms to exercise due skill and care when entering into any arrangement of outsourcing. According to the FCA handbook, "If a firm outsources critical or important operational functions or any relevant services and activities, it remains fully responsible for discharging all of its obligations under the regulatory system". However, as discussed in this research, there is an appetite for standards and guidelines that allow financial firms to work with third parties that can make it easier to compare, contrast, deploy and terminate services across providers.

One of the options discussed during this research was the scope for the FCA or another authority to regulate technology service providers. This is based on the premise that technology is as important as "prudential" or "conduct" considerations, and should form part of the overall regulatory perimeter. However, there were contrasting viewpoints regarding the scope and timelines of these activities, not the mention the implied resistance from service providers. According to one of our respondents, *"The problem with regulating technology providers is the scope. How do you decide which firms are in scope and what happens to those firms out of scope? Will financial firms stop using these firms because they fall outside our regulatory perimeter? Wouldn't that affect competition and drive smaller firms out of the market."*

5 Guidance for Regulators: Prioritizing Responses to Concentration Risk

Based on analysis and interviewee feedback, this research outlines areas and a conceptual framework (Ref. Fig. 2), that regulators could use to influence firms and service providers.

5.1 *Increased Oversight and Collaboration with Firms*

Regulated firms are already subject to a number of rules and guidelines to ensure that firms conduct appropriate due-diligence and manage risk effectively before using 3rd parties to deliver services. Regulators are also in the unique position to understand

Fig. 2 Conceptual framework to mitigate concentration risk

and influence the shortcomings of firms during both supervision and authorization processes. For example, regulators could further emphasize the role of firms in ensuring that appropriate due diligence is conducted on the "services value chain" and this information should be regularly documented and maintained. Firms could be instructed to send this information to regulators as part of their notification obligations, which would allow supervisors to view supplier concentration risks within each sector. This information could be anonymized and made available to firms to consider during their due diligence processes.

5.2 Interoperability Between Service Providers

Underlying cloud Application Programming Interfaces (API's) are essentially proprietary, because of which customers cannot easily move data from one provider to another. This can potentially cause clients to be "locked-in" and hence susceptible to unfair contract or pricing changes. Issues with interoperability may also reduce competition in the cloud computing sector, with customer concentrated around a few major players thus contributing to concentration risk. Such data lock-in tactics could obstruct data portability and interoperability which could frustrate firms looking to take advantage of the benefits of cloud computing.

Mergers and acquisitions represent a threat in the area of concentration risk. Merged institutions may find that they have inadvertently rapidly increased their exposure to a service provider (BITS 2010). In the pressure to consolidate service provider relationships, renegotiate or cancel contracts, and integrate vendor management programs, concerns around "concentration" may not receive immediate and appropriate attention. Similarly, mergers and acquisitions among other financial institutions can also affect an institution's exposure to specific suppliers. It would be useful for guidance around service provider and firm responsibilities during such situations that could help alleviate this risk.

An obvious solution is to standardize the Application Programming Interfaces (API's) so that clients could deploy SaaS applications across multiple cloud providers or can seamlessly migrate data from one provider to the next. Such functionality could potentially reduce concentration risk to a considerable extent. Regulators could further the development of such standards by coordinating focus groups between the technology providers and consumers to arrive at secure interoperability standards between service providers. During their assessment of firms, regulators could also instil controls to ensure that applications use technologies such as web services with appropriate abstraction layers to ensure that these "application blocks" can be migrated from one provider to the next without major disruption.

5.3 *Obligations During Incidents*

When major incidents occur in SaaS platforms, it is usually the responsibility of the supplier to manage the incident and conduct appropriate root cause analysis. As mentioned in the above sections, during times of crisis, clients usually find it very difficult to extract useful information such as estimated resolution time, workarounds etc. In addition, regulators may also find themselves in a similar position when dealing with incidents caused by a SaaS provider. In such cases, it helps to develop guidelines regarding appropriate communication and cooperation during incidents.

It will also be useful for monitoring supplier responsibilities around data preservation. For example, many providers state that they have no obligation to preserve any of the data when the subscriber's access is terminated in situations where the subscribers have violated the providers terms and conditions such as "breaching acceptable use policy". Providers may also assert that they are not responsible for security issues such as unauthorized disclosure of consumer data or service interruptions caused by cyber-attacks. It will be useful for regulators to develop a consistent view for such scenarios and help streamline contractual expectations.

5.4 *Consistent Terminology*

Due to the lack of a common vocabulary, service providers may have different connotations attached to important deliverables such as availability, reliability and security. Regulators could play an important role in helping to coordinate a standardized vocabulary and reporting mechanism that helps compare and contrast between different service options.

Although there was a concerted view that regulators could positively influence the controls around adoption of cloud services, there were concerns about the progress and the impetus required for the regulator to take action. According to a risk specialist, *"The issue is that the regulators or policy makers will not take action until it is very late. It usually requires a major crisis to occur before policy makers wake up and start taking action"*

Unfortunately, time is of the essence in this case. With the increasing trend of firms looking to adopt cloud services in the near future, it would be useful for authorities to take concerted at the earliest, so that firms and providers can benefit from these recommendations during transition.

6 Concluding Comments

The scope of the study is limited to firms in the financial sector that fall in the regulatory perimeter within the FCA. However, considering that the FCA regulates over 55,000 firms, this research focuses on the larger or systemically important firms who are obliged to report significant events to the FCA. Due to limitations in data sources, the examples in this study does not completely distinguish between public, private and hybrid cloud services and the inherent risk may vary depending on the architecture of cloud services. For example, in a private cloud environment, service providers may offer dedicated infrastructure and bespoke contracts which could reduce some of the risks related to availability and performance.

In conclusion this paper's approach of using a sector wide lens to view the impact of concentration risk, driven by outsourcing, has several practical implications for firms, service providers and regulators. Firms clearly have an obligation to conduct appropriate due-diligence and consider the impact of concentration risk on thier ability to operate effectively. Firms are recommended to improve their supplier management capabilities and institute appropriate backup and recovery mechanisms to ensure business continuity. Service providers need to be cognizant of security and resilience of their applications and infrastructure. Cloud providers should collaborate with each other to develop standards in service levels and interoperability that will enable customers to compare services and make an informed decision. Regulators have a significant role in terms of increased oversight of firm specific risks as well as looking into sector wide implications. Regulators are encouraged to work with both firms and service providers and provide appropriate guidelines that could further increase the resilience of the financial sector.

Appendix: Methodology

See Table 1.

This paper analyses data from the following four sources. For a sector specific view, we will adopt a qualitative approach by using interviews to gather primary data. According to Myers (2009), there are three types of interviews that can be used—structured, semi-structured, and unstructured. This paper will use semi-structured interviews since this format allows the researcher to be flexible about the questions and at the same time, respondents can provide additional details or opinions about their thoughts on the research questions. We believe that this will supplement the information gathered in the quantitative phase, thus enabling us to benefit from the advantages of the embedded mixed mode approach.

To ensure uniformity, "technology risk specialists" from each of the FCA sector's were selected as interviewees. The interview questions are specifically focused on outsourcing, concentration risk and sector impact. Since the interviews are semi-structured, the respondents will be given sufficient time to elaborate on issues that are

Table 1 Data sources

Description	Source	Primary/Secondary	Data type
Incident data reported by financial firms	As regulatory entities, firms are expected to adhere to FCA principles of which principle 11 states that "A firm must disclose to the appropriate regulator appropriately anything relating to the firm of which that regulator would reasonably expect notice" (FCA 2016). To comply with this principle, firms need to report incidents or disruptions caused by information systems outages	Secondary	Quantitative
Retail Banking Supervisory Review (Dear Chairman Exercise)	The FCA initiated a supervisory review (the "Dear Chairman Exercise II—2014") which sought to understand levels of resilience and the degree to which the largest UK retail banks are exposed to technology risk. The output of the review will be analysed to provide additional insight into the research topic	Secondary	Qualitative
Sector specific views of concentration risk	To promote efficient regulation, the regulated firms are divided into a system of sectors based on the nature of the firm's regulated activity. This paper will use the sectors published by the FCA in their 2017 business plan which are as follows: Retail Banking and Payments, Retail Lending, General Insurance and Protection, Pensions and Retirement income, Retail Investments, Investment Management and Wholesale financial markets. (FCA 2017)	Primary	Qualitative
Publicly available information	Depending on the impact of disruptions, incidents could also be subject to media coverage or be the subject of thematic studies. Publicly available information will be analyzed to provide useful input into this research	Secondary	Qualitative

important to their sector. Although, there was no a hard stop, we estimate each interview to lasted for about an hour. The interviews were anonymized and transcribed to preserve the confidentially and integrity of the responses.

For the Retail Banks and Payments sector, we will also use anonymized inputs from the Retail Banking Supervisory Review, also known as the Dear Chairman Exercise (DCE-2014 follow-up), which sought to understand levels of technology resilience among the UK's seven largest banks. This research will support the sector specific findings by using information from significant incidents reported by regulated firms. These incidents are logged into an FCA database that captures fields such as firm name, incident impact and the root cause. Due to the confidential nature of this data, the firm name and associated vendor information were anonymized. However, when such incidents are subject to media coverage, we used the available information from the media to corroborate our findings.

To examine the interview data, we adopted the process suggested by Gioia et al. (2013), which allows the researcher to systematically introduce methodological rigor into qualitative analysis. The approach involves three phases, which begins by grouping similar respondent quotes resulting in first order concepts. In the second phase, these first order codes are organized to identify relationships and correlations (second order concepts) that can be further distilled into overarching aggregate dimensions in the third phase. The advantage of this process is that it can provide a constructive data structure that presents the raw data in terms of manageable discussion themes—drivers of concentration risk and mitigation factors, which could be related back to the original descriptive codes from the interviews.

References

Ang, S., & Straub, D. (1998). Production and transaction economies and IS outsourcing: A study of the U.S. bank industry. *MIS Quarterly*.

Armbrust, M. et al. (2009). Above the clouds: A Berkeley view of cloud computing. UC Berkeley Technical Report.

Basel. (2005). The joint forum at basel committee on banking supervision, outsourcing in financial services. in *Bank for International Settlement.* Basel.

BBC News. (2016). Fast cash: The high-speed world of cloud-based finance–BBC News. [Online] Available at: http://www.bbc.co.uk/news/business-36129408.

BITS. (2010). BITS Guide to Concentration Risk in Outsourcing Relationships. Financial Services Roundtable.

Bisong, A., & Rahman, S. (2011). An overview of the security concerns in enterprise cloud computing. *International Journal of Network Security & Its Applications (IJNSA) 3*(1), 30–45.

Bolton, J. (2008). The strategic case for outsourcing back-office chores. *American Banker Magazine.*

Braun. Ch,. & winter, R. (2005). Classification of Outsourcing Phenomena in Financial Services. ECIS Proceedings. http://aisel.aisnet.org/ecis2005/2.

Brusnahan. (2015). Saxo Payments selects Oracle FLEXCUBE platform - Electronic Payments International. [Online] Electronic Payments International. Available at: http://www.electronicpaymentsinternational.com/news/saxo-payments-selects-oracle-flexcube-platform-011015-4683699/.

Capgemini. (2015). Shifting Sands: Banking in the Digital Era. https://www.temenos.com/globalassets/mi/rep/shiftingsands-temenos_8th_annual_surveyfinal-w.pdf.

Cloud Pro. (2017). Tesco Bank moves to AWS cloud in just eight months. [Online] Available at: http://www.cloudpro.co.uk/leadership/5575/tesco-bank-moves-to-aws-cloud-in-just-eight-months

ComputerWeekly. (2017). Insurance brokers count cost of lost business as SSP SaaS platform outage enters second week. [Online] Available at: http://www.computerweekly.com/news/450303913/Insurance-brokers-count-cost-of-lost-business-as-SSP-SaaS-platform-outage-enters-second-week.

Gozman, D., & Willcocks, L. (2019). The emerging cloud dilemma: Balancing innovation with cross-border privacy and outsourcing regulations. *Journal of Business Research, 97,* 235–256.

Gozman, D., Liebenau, J., & Mangan, J. (2018). The innovation mechanisms of fintech start-ups: insights from SWIFT's innotribe competition. *Journal of Management Information Systems, 35*(1), 145–179.

Dobinson, C. (2017). HSBC CIO Darryl West uses bank adopting cloud-first strategy. [Online] CIO UK. Available at: http://www.cio.co.uk/cloud-computing/hsbc-cio-darryl-west-says-bank-adopting-cloud-first-strategy-3655976/.

Enterprise Cloud News. (2017). Royal Bank of Scotland Banks On Cloud - Enterprise Cloud News. [Online] Available at: http://www.enterprisecloudnews.com/author.asp?section_id=560&doc_id=732701.

FCA. (2016), Guidance for firms outsourcing to the 'cloud' and other third-party IT services. Available at: http://www.fca.org.uk/static/fca/article-type/news/fg16-5.pdf

FCA (2017), FCA Sector Views. Available at https://www.fca.org.uk/publications/corporate-documents/sector-views

Finextra Research. (2016). Tandem selects Fiserv Agiliti as IT backbone. [Online] Available at: https://www.finextra.com/newsarticle/28441/tandem-selects-fiserv-agiliti-as-it-backbone.

Finnegan. (2017). HSBC turns to Google Cloud for analytics and machine learning capabilities. [Online] ComputerworldUK. Available at: http://www.computerworlduk.com/cloud-computing/hsbc-turns-google-cloud-for-analytics-machine-learning-3655688/.

Finnegan. (2015). How Tesco Bank has adopted AWS cloud as 'business as usual' in eight months. [Online] ComputerworldUK. Available at: http://www.computerworlduk.com/cloud-computing/how-tesco-bank-has-adopted-aws-cloud-as-business-as-usual-in-eight-months-3629767/

FIS Global. (2017). [Online] Available at: https://www.fisglobal.com/EmpoweredClients/Atom.

Fiserv. (2017). Agiliti from Fiserv Fuelling Change in UK Fintech. [Online] Available at: http://investors.fiserv.com/releasedetail.cfm?releaseid=938273].

Fiserve. (2010). Tesco Bank Live on Signature Bank Platform from Fiserv. [Online] Available at: http://investors.fiserv.com/releasedetail.cfm?releaseid=532522

Foley. (2016). Global DNS outage hits Microsoft Azure customers| ZDNet. [Online] ZDNet. Available at: http://www.zdnet.com/article/global-dns-outage-hits-microsoft-azure-customers/

Gioia, D. A., Corley, K. G., & Hamilton, A. L. (2013). Seeking qualitative rigor in inductive research notes on the Gioia methodology. *Organizational Research Methods*.

Hirschheim, R., & Lacity, M. C. (1993). The information systems outsourcing bandwagon. *Sloan Management Review.*

Hon, W. K., & Millard, C. (2016). Use by Banks of Cloud Computing: An Empirical Study. Queen Mary School of Law Legal Studies Research Paper No. 245/2016. Available at SSRN: https://ssrn.com/abstract=2856431.

HP. (2017). HP News—Deutsche Bank and Hewlett-Packard sign agreement. [Online] Available at: http://www8.hp.com/uk/en/hp-news/press-release.html?id=1919059.

IBM. (2017a). Lloyds Banking Group Signs IBM Cloud Deal. [Online] Available at: http://www-03.ibm.com/press/uk/en/pressrelease/52569.wss.

IBM. (2017b). Scalable hybrid cloud infrastructure [online] Ibm.com. Available at: https://www.ibm.com/it-infrastructure/sa-en/hybrid-cloud/scalability/.

IBS Intelligence. (2015). Nutmeg moves back office ops in-house, goes live with new platform from Babel Systems—IBS Intelligence. [Online] Available at: https://ibsintelligence.com/ibs-journal/ibs-news/nutmeg-moves-back-office-ops-in-house-goes-live-with-new-platform-from-babel-systems/.

Intelligent Environments. (2016). The rise and rise of the challenger banks—Intelligent Environments. [Online] Available at: https://www.intelligentenvironments.com/the-rise-and-rise-of-the-challenger-banks

IOSCO, (2005), Technical Committee of the international organisation of securities commissions.

Judge. (2014). Microsoft confirms Azure outage was human error. [Online] DatacenterDynamics. Available at: http://www.datacenterdynamics.com/content-tracks/colo-cloud/microsoft-confirms-azure-outage-was-human-error/92880.fullarticle.

Kakabadse, N., & Kakabadse, A. (2000). Critical review—outsourcing: A paradigm shift. *Journal of Management Development.*

Keahey, K., Armstrong, P., Bresnahan, J., LaBissoniere, D., & Riteau, P. (2012). Infrastructure outsourcing in multi-cloud environment, in *Proceedings of the 2012 Workshop on Cloud Services, Federation, and the 8th Open Cirrus Summit*.

Badger, L., Patt-Corner, R., & Voas, J., Draft cloud computing synopsis and recommendations of the national institute of standards and technology. NIST Special Publication (vol. 146). [Online]. Available: http://csrc.nist.gov/publications/drafts/800-146/Draft-NIST-SP800-146.pdf.

Lacity, M., & Hirschheim, R. (1993). Information Systems Outsourcing; Myths, Metaphors, and Realities. New York: Wiley.

López, J A. (2004). Outsourcing by financial services firms: the supervisory response. *FRBSF Economic Letter*. http://www.frbsf.org/publications/economics/letter/2004/el2004-34.pdf

Manning, S., Massini, S., & Lewin, A. (2008). A dynamic perspective of next generation offshoring: The global sourcing and science and engineering talent. *Academy of Management Perspectives, 22*(3), 35–54.

Myers, M. D. (2009). *Qualitative research in business & management*. Sage Publications Ltd.

Myllykoski, J., & Ahokangas, P. (2013). Transformation towards a cloud business model. *Communications of the Cloud Software*.

Ohpen. (2012). Ohpen provides Robeco ICT platform—Ohpen. [Online] Available at: http://www.ohpen.com/ohpen-provides-robeco-ict-platform/.

Oracle.com. (2015). Hampden and Co. Selects Oracle FLEXCUBE on Managed Cloud Services to Deliver Superior Customer Service. [Online] Available at: http://www.oracle.com/us/corporate/press/2404861.

Prahalad, C. K., & Gary, H. (1990). The core competence of the corporation. *Harvard Business Review*.

PwC. (2016). Traditional Financial Services Firms fear almost a quarter of their business is at risk from Fintechs. [Online] Available at: http://press.pwc.com/News-releases/traditional-financial-services-firms-fear-almost-a-quarter-of-their-business-is-at-risk-from-fintech/s/f3f3dea5-cbf9-4b0d-aff2-164f418e6451.

Quinn, J. B., & Hilmer, F. G. (1994). Strategic outsourcing. *Sloan Management Review*.

Savvas. (2014). Monitise and IBM join forces to offer cloud-based mobile payment platform. [Online] ComputerworldUK. Available at: http://www.computerworlduk.com/it-vendors/monitise-ibm-join-forces-offer-cloud-based-mobile-payment-platform-3542146/.

Saxo Payments. (2015). Saxo Payments Teams Up With Oracle|Saxo Payments. [Online] Available at: https://www.saxopayments.com/saxo-payments-teams-up-with-oracle-10885.

Techflier. (2016). Top 20 High Profile Cloud Failures of All Time—Techflier. [Online] Available at: https://www.techflier.com/2016/01/25/top-20-high-profile-cloud-failures-all-time/

Temenos, (2013). [Online] Available at: https://www.temenos.com/en/news-and-events/news/2013/September/temenos-expands-the-availability-of-t24-on-the-azure-platform/.

Temenos. (2017). [Online] Available at: https://www.temenos.com/globalassets/mi/cs/cs-bco-capital-rapid-growth-with-t24.pdf.

Verner, J. M., & Abdullah, L. M. (2012). Exploratory Case Study Research: Outsourced Project Failure.

Hon, W., Millard, C., & Walden, I. (2013). Who Is Responsible for Personal Data in Clouds?

Willcocks, L. P., & Lacity, M. C. (1998). *Strategic Sourcing of Information Systems*. Chichester: Wiley.

Benefitting from Service Workforce Platforms

Scoping Review of Crowdsourcing Literature: Insights for IS Research

Dorit Nevo and Julia Kotlarsky

Abstract Fueled by the growth of Internet-based platforms that provided its technological foundation, and the need for an agile and uniquely skilled workforce, crowdsourcing has grown from the grassroots. Initially linked to more mature concepts such as open innovation and outsourcing, it is slowly developing into its own phenomenon, with a growing body of research investigating its many aspects. To gain insight into the crowdsourcing phenomenon, this chapter thoroughly reviews the literature to identify both areas of saturation and gaps, with a focus on the implications for IS research. Pulling together knowledge on specific aspects of crowdsourcing, we offer a scoping review that provides high-level picture of the current literature. Through this review, we identify key themes that emerge out of the many applications of crowdsourcing, and synthesize the literature to chart a more focused research path moving forward.

1 Introduction

Crowdsourcing was introduced in a short paper by Howe in 2006 as "the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call" (Howe 2006). It enables organizations to reach beyond immediate resources and tap into new knowledge and skills. Fueled by the growth of Internet-based platforms that provided its technological foundation, and the need for an agile and uniquely skilled workforce, crowdsourcing grew from the grassroots. It was initially linked to more mature concepts, such as open innovation and outsourcing, before slowly developing into its own phenomenon. Accordingly, research on crowdsourcing is somewhat dispersed between a descriptive focus on applications and cases (spanning multiple domains from product innovation, to public policy, to scientific discovery) and

D. Nevo
Lally School of Management, Rensselaer Polytechnic Institute Troy, New York, USA

J. Kotlarsky (✉)
The University of Auckland, Auckland, New Zealand
e-mail: j.kotlarsky@auckland.ac.nz

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_15

focused investigation of specific aspects of crowdsourcing, such as crowd motivation, metrics and performance measures, or platform design.

To gain insight into crowdsourcing research in particular, and the crowdsourcing phenomenon in general, this paper thoroughly reviews the literature to identify areas of saturation and gaps, with a focus on the implications for IS research. Pulling together knowledge on specific aspects of crowdsourcing, we offer a higher-level scoping review of the literature, identify key themes that emerge out of the many applications of crowdsourcing, and synthesize the literature to chart a more focused research path moving forward. To this end, we gather insights under a single literature review umbrella that captures crowdsourcing as a stand-alone phenomenon, while also acknowledging its ancestry and nomological network.

2 A Brief Overview and Definition

Crowdsourcing is most commonly defined using the abovementioned definition coined by Howe in 2006. While many of the papers reviewed use the above definition, or some close variation of it, examples of other definitions found in the literature are shown in Table 1. Taken together, these definitions highlight unique characteristics of crowdsourcing, which include: (i) solicitation of work that cannot be completed (or cannot be completed efficiently) within the organization or within a specific group; (ii) a temporary workforce (internal or external) with diverse knowledge and skills; (iii) a technology platform; (iv) a task decomposition approach (e.g. micro-tasks versus competitions); and (iv) a compensation mechanism.

Common crowdsourcing models include: peer production (approach based on voluntary contributions from multiple actors as in the cases of Open Source Software and Wikipedia); tournaments (where only the top submissions contributed by the crowd are awarded payment; this approach is often associated with open innovation); and microsourcing (tender-based approach often used for relatively small and labor intensive tasks such as bug testing)[1] (Gefen and Carmel 2008; LaToza and van der Hoek 2016). The platform itself can serve as a facilitator, an aggregator, an arbitrator, or a governor (Kaganer et al. 2013). Work is characterized by the granularity of the task, the selection of workers, and coordination and collaboration approaches (Deng and Joshi 2016). Further, crowdsourcing can happen directly, i.e., when a focal firm reaches out to the crowd of individuals through an open call, or it can happen indirectly through an intermediary that connects the crowd and the firm.

The popularity of crowdsourcing is fueled by two important organizational trends: an increasing focus on process flexibility and agility (Ågerfalk et al. 2009) that has made organizations more comfortable with using non-traditional approaches in accomplishing work activities; and, an increasing familiarity and comfort with

[1] This review does not include papers on crowdfunding or pure open innovation papers (papers that are not linked to crowdsourcing). We include work on the different crowdsourcing models, as well as related concepts such as liquid talent, gig economy, and human cloud.

Table 1 Crowdsourcing definitions

Authors	Definition (Emphasis added by authors)
Armstrong (2010)	Crowdsourcing can be viewed as a force multiplier: companies and other entities can sometimes **get far more work done** by opening their projects to collaborative input than they could have accomplished solely through the efforts of their own employees. (p. 608)
Dalal et al. (2011)	Crowdsourcing is a novel method of **online, distributed** idea generation, problem-solving, and decision-making that involves an open call to a large, often undefined network or community of people (a "crowd") to provide either independent or collaborative contributions to solving a problem or performing a task (p. 1434)
Deng and Joshi (2016)	A CSWE *[Crowdsourcing Work Environment]* should have all of the six essential characteristics: **on-demand virtual labor,** open access to work, Internet access to join the crowd, three stakeholders, human tasks, and **modular technical architecture**. (p. 650)
Estellés-Arolas and González-Ladrón-de-Guevara (2012)	Crowdsourcing is a type of **participative online activity** in which an individual, organization, or company with enough means proposes to a group of individuals of varying knowledge, **heterogeneity**, and number, via a **flexible open call**, the voluntary undertaking of a task. The undertaking of **the task, of variable complexity and modularity**, and in which the crowd should participate bringing their work, money, knowledge and/or experience, **always entails mutual benefit.** The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantage that what the user has brought to the venture, whose form will depend on the type of activity undertaken. (p. 11)
Franke et al. (2013)	Crowdsourcing communities consist of particularly large numbers of **heterogeneous, self-selected, and voluntary individuals** who engage in **temporary, decentralized** problem-solving activities for the firm. (p. 1495)

(continued)

Table 1 (continued)

Authors	Definition (Emphasis added by authors)
Guesmi (2014)	We can define crowdsourcing as the fact of an organisation **externalising a problem** or activity to a large number of anonymous individuals, with the idea of finding among them people who are capable of solving the problem or successfully carrying out the task in question either individually or collectively. (p. 58)
Karger et al. (2014)	Crowdsourcing systems have emerged as an effective paradigm for **human-powered problem solving** and are now in widespread use for large-scale data-processing tasks such as image classification, video annotation, form data entry, optical character recognition, translation, recommendation, and proofreading. (p. 2)
Leal Bando et al. (2015)	Crowdsourcing is a **mechanism to distribute micro tasks,** which are work items broken into small units. These tasks are made available to a large population of participants who carry them out for a small payment. The tasks require human intelligence to be completed, because they cannot be straightforwardly automated. (p. 973)
Martinez (2015)	Crowdsourcing is defined as a type of participative activity, **powered by advanced internet technologies,** in which an individual or organisation proposes to a large group of individuals, with diverse knowledge, the voluntary undertaking of a task to a pre-determined goal. (p. 1419)
Saxton et al. (2013)	Crowdsourcing is **a sourcing model** in which organizations use predominantly advanced Internet technologies to harness the efforts of a virtual crowd to perform specific organizational tasks. (p. 5)
Thuan et al. (2016)	Crowdsourcing is defined **as an online strategy**, in which an organisation proposes defined task (s) to the members of the crowd via **a flexible open call** in order to harness their work, knowledge, skills and/or experience. (p. 49)

(continued)

Table 1 (continued)

Authors	Definition (Emphasis added by authors)
Volk et al. (2015)	Crowdsourcing can be considered as the evolution of the outsourcing principle, where tasks are submitted to a **huge crowd of usually anonymous workers** by a requester in the form of an open call, instead of a designated employee or subcontractor that is assigned a specific job by the employer. (p. 100)
Yoo et al. (2013)	Crowdsourcing has been distinguished as an effective means of open innovation for collecting creative ideas from **people who have various degrees of expertise and diversity** in knowledge. (p. 73)
Zuchowski et al. (2016)—Internal Crowdsourcing	**Internal crowdsourcing** is an (a) IT-enabled (b) group activity based on an (c) open call for participation (d) in an enterprise. (p. 168)

social technologies and open innovation (e.g., von Krogh et al. 2012). Crowdsourcing, with its inherent social and market characteristics, offers value in both these aspects, enabling reduced time to market, reduced costs, improved quality through broad participation, generation of alternate solutions with increased creativity, and the ability to employ specialists, often on an ad hoc basis (LaToza and van der Hoek 2016; Stol and Fitzgerald 2014).

As a phenomenon, crowdsourcing has been studied through early case studies as well as conceptual and empirical work. We examine these studies by providing an overview of the broad crowdsourcing literature to date in a form of a scoping review (Paré et al. 2015) that aims to provide a bird's-eye perspective on this emerging phenomenon and includes a broad range of published papers.

3 Review Method

Our approach to reviewing the crowdsourcing literature followed recommendations for a comprehensive and systematic literature review by Webster and Watson (2002), Rowe (2014), and Fink (2013). We designed a broad scoping review that involved a comprehensive search for conceptual and empirical articles that we analyzed thematically. We elaborate below on the detailed criteria used for the searches, and inclusion and exclusion of papers.

A scoping review "attempts to provide an initial indication of the potential size and nature of the available literature on a particular topic" (Paré et al. 2015; p. 186). Our aim was to understand the extent and range of research on crowdsourcing, to capture the scope of the crowdsourcing phenomenon with its boundary conditions.

We conducted a search using ABI/INFORM Complete, searching initially on the keywords (i) "crowdsource" OR "crowdsourcing" within a paper's abstract, and limiting the search to peer reviewed journals in English only. This resulted in 529 papers. We then conducted a second search adding the keywords (ii) "innovation contest", (iii) "crowd" AND "tournament", and (iv) "crowd" AND "contest". This brought the total number of results to 572 papers. Finally, we conducted an additional search for conference proceedings and, after consulting with existing literature review papers regarding inclusion/exclusion of conference proceedings (e.g., Dibbern et al. 2004; Karimi Alaghehband et al. 2011), we decided to focus on full research papers included in the proceedings from the top IS conference—International Conference of Information Systems (ICIS). We found 28 ICIS papers, which we added to our search results.

The 600 journal and conference articles resulting from the above search were reviewed in our initial scoping review. After removing duplicates, we were left with 591 articles. To develop the coding scheme, one of the authors first scanned the articles to elicit broad topic categories. Once there was sufficient convergence and no new categories emerging, the coding scheme was reviewed and finalized. The final list of coding categories along with their definitions is provided in Table 2. A deeper

Table 2 Categories of papers reviewed

Category	Definition
Applications	Papers that describes an application, or application domain, for crowdsourcing
Crowdsourcing as a Research Method	Papers that describe the specific use of crowdsourcing to conduct research
Design	Papers that offer insights on designing crowdsourcing competitions/platforms
Literature Review	Either broad or specialized literature reviews of some aspects concerning crowdsourcing
Managing Crowdsourcing	Papers that focus on managerial issues related to crowdsourcing (such as challenges)
Outcome Metrics	Papers that offer specific metrics to measure competition outcomes
Overview	Broad overview papers that introduce crowdsourcing either in general or within a specific domain
Participation	Papers that focus specifically on aspects of participation in crowdsourcing competition, such as motivation and incentives
Theorizing	Papers that offer theoretical insights for crowdsourcing. When a paper is classified as a theory paper, a secondary category is chosen to specify the focus of the theory (e.g., participation, design, etc.)
Value	Papers that discuss the value of crowdsourcing

discussion of each group, along with insights from our scoping review, follows in the next section.

Next, both authors reviewed articles for exclusion. A total 107 of the initial 591 articles were excluded from further analysis. The excluded articles were either deemed irrelevant to our core crowdsourcing focus, or as not offering sufficient content (for example, editorials, commentaries, and other notes). Appendix 1 provides examples of such excluded articles. Finally, one of the authors and one other reviewer (a research assistant) separately reviewed the remaining 484 abstracts and assigned them to the coding categories. Initial agreement between the two reviewers on the assigned categories was 78%. To determine the categories for the remaining 22% of papers, the full text (rather than just the abstract) was carefully reviewed by the authors and a category assigned.

4 Scoping Review of Crowdsourcing Literature

Figure 1 shows the number of papers included in the scoping review by category (as listed in Table 2). Figure 2 provides a longitudinal view of papers within each category (with single years as the unit of analysis). The lines plot the number of

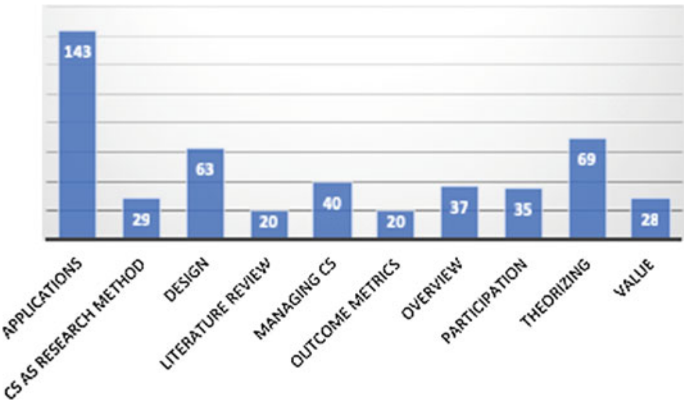

Fig. 1 Count of papers by category

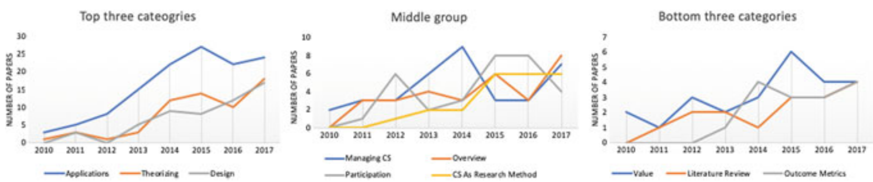

Fig. 2 Count of papers within categories and over time

papers published within each category from 2010 until 2017. Prior to 2010 there were only a few papers published eaFch year and we therefore excluded this early period for presentation simplicity. We also excluded 2018 from these graphs, as our data represent only part of the year. For ease of viewing, we have split the figure into three groups based on the total number of papers published within each category. As can be seen from the line charts, the overall number of papers shows a generally upward trend, reflecting a growing interest in research and practice in the area of crowdsourcing. Below, we discuss each category of papers in depth.

4.1 Applications

A large number of the papers we reviewed describe some application of crowdsourcing. One common application example is the use of crowdsourcing in disaster management and recovery (e.g., Han et al. 2018; Munro 2013). Another example includes government and public policy applications (e.g., Glaeser et al. 2016 at the city level, Mergel and Desouza 2013 at the country level), where crowdsourcing is used to increase citizen participation in government (Lastovka 2015). Mobile applications of crowdsourcing is another widely studied area; these are generally location based applications (e.g., Komninos et al. 2017; Wang et al. 2015). Other common applications prevalent in the literature include innovation contests (Merchant et al. 2014), crowdsourcing for software development (Badihi and Heydarnoori 2017), and scientific challenges (Franzoni and Sauermann 2014; Savage 2012). Finally, some papers also deal with applications that more strongly integrate human and machine components, for example annotations, classifications, and information retrieval (Granell et al. 2018; Lease and Yilmaz 2013; Ntalianis et al. 2014).

A slightly less common but quite interesting application employs crowdsourcing within organizations (e.g., Smith et al. 2017). Internal crowdsourcing refers to an IT-enabled group activity that is based on an open call for participation in an enterprise (Zuchowski et al. 2016, p. 168). Further, offering an organizational strategic focus, Steiger et al. (2012) suggest using crowdsourcing to include organization members and tap into their knowledge when developing organizational strategy. Similarly, Aten and Thomas (2016) describe a case of open strategizing in an organization, focusing on enabling technological affordances.

4.2 *Theorizing*[2]

A good number of papers in recent years have examined theoretical aspects of crowdsourcing. These studies span a wide range of phenomena including, for example, cost signaling (Huang et al. 2014), award structures (Terwiesch and Xu 2008), participation decisions (Faullant et al. 2016), solver engagement (Martinez 2015), and user types (Füller et al. 2014). Broadly, the crowdsourcing theory papers can be differentiated based on the crowdsourcing (CS) model they study, or their focal entity.

Saxton et al. (2013) analyzed over a hundred crowdsourcing websites to develop a "taxonomic theory" of crowdsourcing. The taxonomy differentiates models based on their level of collaboration, compensation structure, and role of the community, among others, and results in nine different crowdsourcing models. The intermediary model, for example, is used for R&D and innovation, for software development, and for office/knowledge work. For contracts, the digital goods sales model hosts crowdsourcing websites that simply offer digital stock photos, and audio and video clips. Such differentiation of CS models can play an important role in defining the boundaries of theoretical work. Jian et al. (2017), for example, studied the performance of simultaneous versus sequential contests. Their insights are limited to contest-based CS models. Deng and Joshi (2016), on the other hand, focus on a different CS model in their study of workers' participation in microtask crowdsourcing environments. Our review reveals that the majority of theory papers focus on contest-based models.

A second differentiating factor in theoretical CS papers is the focal entity. Wen and Lin (2016), for example, focused on the platform itself to describe a game-theoretic model that identifies properties of the platform's optimal fee schedule. Afuah and Tucci (2012) adopted the firm's point of view to study the circumstances under which the firm may choose crowdsourcing over internal solutions. Nickerson et al. (2017) offer yet another perspective by focusing on the problem itself in order to derive strategic value from the CS experience. Similarly, Zhao and Zhu (2016) offer task affordances as a new lens under which to study crowdsourcing projects. Finally, papers that adopt a crowd's point of view study crowd members' behaviors (Hutter et al. 2015), effort (Sun et al. 2015), and successful participation (Mack and Landau 2015), to mention a few.

From a strategic perspective, several theoretical papers focus on drivers of organizational value from crowdsourcing. For example, Piezunka and Dahlander (2015) studied the impact of organizational attention to near versus distant solutions, Xu et al. (2015) studied the link between crowdsourcing and firm performance, and Piyathasanan et al. (2018) examined a longitudinal model of creative engagement and value creation.

[2]Note that many of the theory papers will be discussed again in our critical review section, which focuses on a select subset of papers. Here we only provide an overview of papers that fall under the category.

4.3 Design

The different CS models are also quite apparent in the design focused articles identified in the scoping review. Papers here range from those focused on tweaking very specific managerial features of crowdsourcing, such as preventing cheating (Eickhoff and de Vries 2013), offering algorithms for task allocation (Karger et al. 2014), or maintaining workers' privacy (Kajino et al. 2014), to broader designs of different models. For example, Yuan and Ching-Fang (2018) propose a new design for an intermediary crowdsourcing solution, whereas Fang et al's (2017) design a crowdsourcing workflow of image labeling on a more automated microtask platform. A final group of papers within this category offers designs for a particular application; for example, a specific design for missing value imputation with crowdsourcing (Wang et al. 2017), and a design for a crowdsourcing-based road surface monitoring system (Chen et al. 2016). Quite a few papers in the design category are very specific in nature and detail oriented, thus offering little focus on higher level strategic impacts.

4.4 Managing Crowdsourcing and Overview

These two categories are fairly similar in their focus on a managerial audience, and we thus combine them here. Compared with other categories, we found a larger portion of early overview and managerial papers, with many papers in both categories published by 2015. This is not surprising, as early literature tends to broadly introduce a new phenomenon before studying it in depth. The overview papers, as the name suggests, provide a general overview of crowdsourcing and its potential applications. More so than papers in other categories, overview papers tend to compare and contrast the crowdsourcing phenomenon with other phenomena such as mass collaboration (Doan et al. 2011), outsourcing (Przybylska 2013), open source software (Olson and Rosacker 2013), and open innovation (Battistella and Nonino 2012). The papers on managing crowdsourcing cover a range of management topics, with the common thread being organizational strategy. These papers focus on topics such as risk management (Kannangara and Uguccioni 2013), barriers for success (Lüttgens et al. 2014), organizational capabilities (Nevo and Kotlarsky 2014), business models (Kohler and Nickel 2017), and scalability (Kohler 2018).

4.5 Participation

A unique subcomponent of managing crowdsourcing, participation receives a category of its own based on the number of papers that focus on either incentives, motivation, or contributions of crowd members.[3] Incentives are the focus of most recent papers within this category, with researchers either designing incentive mechanisms (Liu et al. 2017; Wang et al. 2018) or examining the impact of incentives on participation (Acar 2018; Li and Hu 2017). A second subcategory focuses on motivation to participate. Studies here explore specific designs such as gamification (Kavaliova et al. 2016), or specific drivers such as commitment (Schulten and Schaefer 2015), and enjoyment and empowerment (Goncalves et al. 2015). Finally, a small number of papers examine the contributions of crowd members over time (Bayus 2013; Heo and Toomey 2015; Soliman and Tuunainen 2015).

4.6 Crowdsourcing as a Research Method

As crowdsourcing gains popularity in applications, its use is mirrored in the literature as a promising research method. The papers reviewed either focus on the applicability of crowdsourcing as a research method, or simply utilize crowdsourcing in research projects. The first group of papers investigate the validity and usefulness of crowdsourcing as a research method. Some of these papers focus on the validity of crowd work. For example, Foody et al. (2013) assessed the accuracy of crowd collected data in the case of land surveys, and Lovett et al. (2018) studied data quality on Amazon Mechanical Turk. Other papers explore the generalizability of insights obtained through crowdsourcing. For example, Volk et al. (2015) examined the suitability of crowdsourcing for field experiments and its value compared to laboratory experiments. Yank et al. (2017) studied how participants recruited through crowdsourcing are similar to- (or different from-) the general US population in terms of their health profiles. Based on their findings, they caution against making generalized assumptions in crowdsourcing studies. Finally, Steelman et al. (2014) compared student samples, consumer samples, and crowdsourced samples in information systems (IS) research. While they found the psychometric properties of the tested models were similar across samples, they advise caution with regard to possible cultural differences between crowdsourced samples.

A second group of papers reporting the general use of crowdsourcing as a research method explores potential applications for crowdsourcing in supporting research. For example, Love and Hirschheim (2017) offer a framework for using crowdsourcing as a new research genre for IS research. Weiss (2016) discusses the usefulness of crowdsourcing as a method for conducting literature reviews in new domains, and Conley and Tosti-Kharas (2014) propose crowdsourcing as a method to support content analysis.

The final group of papers that fall under this category report studies that have simply utilized a crowdsourcing platform as their approach for data collection. Among

[3] Several papers explore theoretical aspects of participation such as the effect of intrinsic and extrinsic motivation (Faullant et al. 2016) and are included under the "Theorizing" category..

these papers, the dominant platform was Amazon Mechanical Turk (e.g., Kaufmann and Tummers 2017; Kim et al. 2016).

4.7 Value

Value is an important factor when considering any new technology. In the case of crowdsourcing, papers categorized as exploring the value of crowdsourcing offer empirical evidence for the crowd's value (for example Garrigos-Simon et al. (2017) measured the value created by crowdvoting in the case of booking.com). Other papers explore the ability of the crowd to contribute value, studying, for example, whether crowd users can compete with professionals when it comes to new product development and creating value for the company (Poetz and Schreier 2012). Yet other papers focus on how to enhance the value created by the crowd, for example proposing an approach to enhancing the value of crowdsourced data (Barbier et al. 2012). Finally, interesting and innovative applications for drawing value from crowdsourcing are also proposed in the literature, for example by using crowdsourcing to enhance the competitive ability of lean start-ups (Erkinheimo et al. 2015), or enabling organizations to compete with ordinary resources (Fréry et al. 2015). At a higher level, Kohler (2015) studied crowdsourcing based business models in depth to explore how value is created by each model. Analyzing crowdsourcing business models at varying levels of success, he addresses specific challenges to value creation and prescribes actions for organizations.

4.8 Literature Reviews

Our search yielded 20 papers that we classified under the literature review category. These papers range from those placing crowdsourcing within the broader context of open innovation, to reviews of direct crowdsourcing literature, to targeted reviews of specific aspects of crowdsourcing (such as motivation to participate) or applications (such as crowdsourcing in education).

Starting with the broader reviews, these papers place crowdsourcing models within the context of open innovation (Hossain 2013; Schweisfurth et al. 2011). For example, Schweisfurth et al. (2011) examined five models of open innovation, crowdsourcing being one of them, and compared them in terms of the actors involved, motivation for individuals and organizations, contractual framework, decision rights, and the innovative process. Among the open innovation models reviewed, crowdsourcing is characterized as open to all actors, strongly linked to financial motivation (for both individuals and organizations), relatively transparent and with centralized decision making, and spanning different stages of the innovation process from ideation to development to marketing.

Narrowing in on the phenomenon itself, Estellés-Arolas and González-Ladrón-de-Guevara (2012) reviewed crowdsourcing definitions and examined specific components of the definition in depth. Their review offers insights on who forms the crowd, what the crowd is doing, what they get in return, who is initiating the work and what value they receive, as well as the medium used, the type of call, and the overall process employed. The outcome is a detailed definition of crowdsourcing, spanning the above dimensions (see Table 1 above for their definition), and an application of their definitional criteria to various tools, thereby setting an inclusion boundary for what constitutes crowdsourcing.

Three early reviews (Hossain 2015; Hossain and Kauranen 2015; Zhao and Zhu 2014) provide an overview of crowdsourcing research up to 2013. As an early exploration of the topic these reviews are relatively descriptive, exploring intended audience, theoretical foundations and research methods employed, the focus of the papers reviewed (conceptualization, system, or application), and common applications. Two later reviews, covering the literature up to 2015, provide an overview of emerging research areas and applications (Palacios et al. 2016) and avenues for future research structured around the Input–Process–Output framework (Ghezzi et al. 2018). Some promising trends include problem solving, new product development, and organizational innovation (Palacios et al. 2016). Interesting research questions are posed concerning: When is crowdsourcing most beneficial?; How to govern the crowdsourcing process?; and How to better integrate crowdsourcing outcomes into organizations? (Ghezzi et al. 2018).

Finally, targeted reviews have examined motivations for crowdsourcing participation (Smith et al. 2013), design (Haller et al. 2017), task allocation (Guo et al. 2018), factors influencing the decision to crowdsource (Thuan et al. 2016), and provided an overview of decisions along the crowdsourcing workflow (Neto and Santos 2018). Equally targeted, but focusing on application areas, several reviews have focused on the use of crowdsourcing in public policy and government (Liu 2017; Prpic et al. 2015; Taeihagh 2017), education (Skarzauskaite 2012), information visualization (Borgo et al. 2018), and data management (Crescenzi et al. 2017). An interesting review from 2016 examined internal crowdsourcing in depth (Zuchowski et al. 2016). The authors draw a picture of the state of practice in internal crowdsourcing, including common problems to which crowdsourcing is applied (collective intelligence, design, decisions), governance tasks, key actors, the role of IT, process, and outcomes. They also offer a useful summary of differences between internal and external crowdsourcing, as well as traditional work.

4.9 Outcome Metrics

The final group of papers we describe provides metrics for measuring crowdsourcing outcomes. Common in this group are quality metrics such as accuracy (Foody et al. 2013; Oosterman et al. 2015), solution diversity (Zheng et al. 2014), idea distinctiveness—which is a compounded measure of content, contributor, and crowd feedback

(Hoornaert et al. 2017), and the trustworthiness of user contributions (Ceolin et al. 2014). A second group of metrics focuses on impact; for example, linking crowdsourcing project risk dimensions with project performance (Liu et al. 2016), linking attention to productivity in crowdsourced settings (Huberman et al. 2009), or assessing engagement with different platforms (Mustafa and Mohd Adnan 2017). Somewhat related to attention, several studies have focused on measuring and predicting crowd votes and majority agreement (Ertekin et al. 2014; Hofstetter et al. 2018; Salk et al. 2017; Wan 2015), and their insights are important for organizations using the crowd to identify promising solutions.

Delving deeper into quality, and comparing crowdsourcing with traditional approaches, an interesting study on visual object segmentation (Carlier et al. 2016) measured "crowdsourcing loss", which is defined as the loss incurred when the task is completed by a crowd, as opposed to domain experts. The study provides rich insights by comparing these two groups and potential explanations for why even the expert crowd worker performs worse than the average domain expert. Along similar lines, Lovett et al. (2018) employed a mixed methods approach to study in depth the expected data quality from AMT surveys.

5 Conclusion and Implications

The nine categories presented in this paper provide an overview of high-level themes that are evident in the crowdsourcing literature to date. Crowdsourcing, which was initially associated with more mature literature streams such as open innovation and IS outsourcing, is evolving towards becoming a phenomenon, and therefore a literature stream, of its own. While it is important to understand and capture its ancestry, it is no less important to identify unique characteristics of this phenomenon, its boundary conditions, as well as to establish and articulate links to its nomological network.

Pulling together knowledge on specific aspects of crowdsourcing, we conducted a high-level scoping review of the literature. Based on these analyses, we identified key themes that emerge out of the many applications of crowdsourcing, and synthesized the literature to chart a more focused research path moving forward.

The paper offers an important contribution to IS researchers as well as practitioners. It provides a snapshot of the current crowdsourcing literature spanning applications, theories, design, metrics and value, and general management studies that shed light on what has been done within this research domain to date. Large number of papers on applications of crowdsourcing, in comparison to any other categories, shows relevance of crowdsourcing to many different context. We believe that crowdsourcing has reached the stage when more theorizing is needed to better understand this phenomenon.

One of the important observations that is evident from this scoping review is that the term "crowdsourcing" has become so popular that it is often used to refer to any crowd-related activity, which makes it really difficult to compare results reported by different studies, and to build upon earlier work towards theory-building about the crowdsourcing phenomenon. A bottom-up taxonomy of crowdsourcing platforms by Saxton et al. (2013) that classified 103 popular platforms into nine groups provides a useful illustration to the breadth of what could be considered as "crowdsourcing". The taxonomy distinguishes between the different roles the platform can play in supporting a wide range of crowd-involving activities, including peer-to-peer social financing (e.g., Lendingclub, Kiva), citizen media production (e.g., Artistshare), digital goods sales (e.g., iStock, ShutterStock), as an intermediary between crowdsourcer and crowd (e.g., Amazon Mechanical Turk, Innocentive, eLance, TopCoder), knowledge base building (e.g., Answers.Yahoo, Wikipedia), the Collaborative Science Project Model (e.g., Planetary, reCaptcha), the Consumer Report Model (e.g., AngiesList), the Product Design Model (e.g., Cafepress, Threadless), and the Collaborative Software Development Model (e.g., FossFactory.org).

One way to deal with the large number of papers that talk about various crowdsourcing models and platforms is to consider specific boundary conditions that identify specific crowd-involving activities that have sufficient common ground to allow integration and theorizing. For example, in our recent work (critical literature review by Nevo and Kotlarsky 2020) we focus on crowdsourcing studies that have an explicit element of "sourcing", that is, undertaking of a task, having a clearly identified crowdsourcer and a specified call. When applying this boundary condition to the taxonomy of crowdsourcing platforms by Saxton et al. (2013), we see that peer-to-peer social financing and digital goods sales do not involve undertaking a task, while citizen media production and knowledge-base building do not typically have a clearly identified crowdsourcer, or a specific call. Therefore, as we argue in Nevo and Kotlarsky (2020), despite the various ways these online platforms enable and facilitate involvement of the crowd, some models do not facilitate engagement in *sourcing*. The intermediary role, however, is a typical example of sourcing that occurs between crowdsourcer and crowd, and meets all the criteria for inclusion in crowd*sourcing*.

We find a similar issue with another taxonomy developed by Boudreau and Lakhani (2013), which addresses three of the definitional criteria—the participative process, the task, and value for the crowdsourcer. This taxonomy distinguishes between different crowdsourcer-crowd interactions in relation to the value the crowdsourcer is expecting to receive, and the nature of the task. Specifically, it defines four ways to use the crowd as a partner through crowdsourcing contests designed to generate high value solutions to complex and novel problems, and in particular, highly challenging technical or creative problems. Collaborative communities allow organizations to aggregate large numbers of diverse contributions and are suited to addressing customer support, creating wikis, or for open collaboration. Complementors offer user generated solutions to product challenges, for example though mashups and apps. Finally, labor markets enable efficient and flexible work, matching talent to task in human computation and repeated tasks. According to the criteria adopted in Nevo and Kotlarsky (2020), crowdsourcing contests and labor markets are clear examples of crowd*sourcing*. However, collaborative communities and complementors only qualify as a sourcing activity if studied in a context that involves a specific project initiated by a crowdsourcer (e.g., Dell or LEGO) in their dedicated collaborative community, or in response to a specific open call that attracts the attention of a number of potential competitors or complementors (e.g., the space poop challenge[4] launched by Nasa in October 2016, or Fiat's "Fiat Mio" project[5] launched in August 2009).

Therefore, while good taxonomies exist to classify crowdsourcing models from the bottom up (i.e., by classifying observed models into emerging groups), there is a degree of misalignment between these taxonomies and how different studies define (or what they view as) crowdsourcing. One way to overcome this issue is to examine typologies, which are similar to taxonomies but based on conceptual classification criteria rather than empirically derived (Bailey 1994). For example, Malone et al. (2010) classify collective intelligence projects according to four building blocks: (1) what is being crowdsourced?; (2) who is performing the task?; (3) why do people do this?; and (4) how is the task being done? They define various combinations of attributes for the above four building blocks, which they refer to as genes ("a particular answer to one of the key questions (What, Who, Why or How) associated with a single task in a collective intelligence system" (p. 22)). Hence, Wikipedia will have a different genome to Threadless or Innocentive. The genome concept is appealing in that it provides a simple approach to crowdsourcing classification. However, as the authors themselves note, additional work is needed to identify all the different genes.

Future research would benefit from establishing more narrow perspectives on crowdsourcing. Examples of such perspectives are the genome approach proposed by Malone et al. (2010) and the focus on sourcing aspect of crowdsourcing adopted in our recent work (Nevo and Kotlarsky 2020). In addition to advancing these perspectives towards developing theoretical understanding of the crowdsourcing phenomenon, one area where we see a potential for IS scholars to distinguish between crowdsourcing platforms and any other digital platforms.

[4] https://www.nasa.gov/feature/space-poop-challenge.

[5] https://www.ennomotive.com/making-the-first-crowdsourced-car-fiats-journey/.

Appendix A

Examples of Excluded Papers

Title	Abstract	Exclusion reason
The Influence of Sports on International Tourism (Howe 1969)	Games attract people to travel abroad, in order to play and to watch. International travel owes a great deal to the spectator sports, particularly to international contests and "world championships;" they attract partisans of the various performers, and the crowd that always gathers to watch a fight	Search result: Crowd AND Contest Exclusion reason: no relevance to crowdsourcing
Mergers in Patent Contest Models with Synergies and Spillovers (Jost and van der Velden 2006)	We consider mergers in an innovation contest between n firms in the presence of synergetic effects. We assume that a merger may affect the R&D efficiency of the merging firm due to increasing returns to scale in R&D. We show that mergers are beneficial for the merging firms even if the efficiency gains of the merging firms are not substantial, and that merging reduces R&D costs by only 6%. We also consider the influence of unintended knowledge flows in R&D. In the presence of knowledge spillovers, we show that higher efficiency gains are needed to make the merger profitable	Search result: Innovation Contest Exclusion reason: minimal relevance to crowdsourcing

(continued)

(continued)

Title	Abstract	Exclusion reason
President's Message (Pace 2008)	The LITA board's main objective is to oversee the affairs of the division during the period between meetings. As a mea culpa for the board, but without placing the blame on any one individual, the author is willing to concede that the board has not done an adequate job of engaging the membership between American Library Association (ALA) meetings. While ALA itself is addressing this problem with recommendations for virtual participation and online collaboration, LITA should be at the forefront of setting the benchmark for virtual communication, participation, education, planning, and membership development. In the past, LITA focused on the necessary technologies for crowdsourcing. Now that the technology is commoditized, perhaps it is time to embrace the philosophy of crowdsourcing	Search result: Crowdsourcing Exclusion reason: not an academic paper (commentary)
Let's Talk (Guzmán 2016)	Here, Guzman says by making the news a conversation, journalists can better serve the communities they cover. Journalists are feeling all this pressure to go big, go viral and get scale. She thinks most of them would do better going small. They must find the communities whose conversations they can do the most to strengthen and get to know them. Not so they can serve them with clicks and shares and crowdsourcing, but so they can serve them better than anyone else	Search result: Crowdsourcing Exclusion reason: not an academic paper (commentary); no relevance to crowdsourcing

References

Acar, O. A. (2018). Harnessing the creative potential of consumers: money, participation, and creativity in idea crowdsourcing. *Marketing Letters, 29,* 177–188.

Afuah, A., & Tucci, C. L. (2012). Crowdsourcing as a solution to a distant search. *The Academy of Management Review, 37,* 355–375.

Ågerfalk, P. J., Fitzgerald, B., & Slaughter, S. A. (2009). Introduction to the special issue—flexible and distributed information systems development: state of the art and research challenges. *Information Systems Research 20*, 317–328.

Armstrong, T. K. (2010). Crowdsourcing and open access: collaborative techniques for disseminating legal materials and scholarship. *The Santa Clara Computer and High—Technology Law Journal, 26,* 591–630.

Aten, K., & Thomas, G. F. (2016). Crowdsourcing strategizing. *International Journal of Business Communication, 53,* 148–180.

Badihi, S., & Heydarnoori, A. (2017). CrowdSummarizer: Automated generation of code summaries for java programs through crowdsourcing. *IEEE Software, 34,* 71–80.

Bailey, K. D. (1994). *Methods of social research*. The Free Press, New York.

Barbier, G., Zafarani, R., Gao, H., Fung, G., & Liu, H. (2012). Maximizing benefits from crowdsourced data. *Computational and Mathematical Organization Theory, 18,* 257–279.

Battistella, C., & Nonino, F. (2012). Open innovation web-based platforms: The impact of different forms of motivation on collaboration. *Innovation: Management, Policy and Practice, 14,* 557–575.

Bayus, B. L. (2013). Crowdsourcing new product ideas over time: An analysis of the dell ideastorm community. *Management Science, 59,* 226–244.

Borgo, R., Micallef, L., Bach, B., McGee, F., & Lee, B. (2018). Information visualization evaluation using crowdsourcing. *Computer Graphics Forum, 37,* 573–595.

Boudreau, K., & Lakhani, K. R. (2013). Using the crowd as an innovation partner. *Harvard Business Review,* 61–69.

Carlier, A., Salvador, A., Cabezas, F., Giro-i-nieto, X., Charvillat, V., & Marques, O. (2016). Assessment of crowdsourcing and gamification loss in user-assisted object segmentation. *Multimedia Tools Application, 75,* 15901–15928.

Ceolin, D., Nottamkandath, A., & Fokkink, W. (2014). Efficient semi-automated assessment of annotations trustworthiness. *Journal of Trust Management, 1,* 1–31.

Chen, K., Tan, G., Lu, M., & Wu, J. (2016). CRSM: A practical crowdsourcing-based road surface monitoring system. *Wireless Networks, 22,* 765–779.

Conley, C., & Tosti-Kharas, J. (2014). Crowdsourcing content analysis for managerial research. *Management Decision, 52,* 675–688.

Crescenzi, V., Fernandes, A. A. A., Merialdo, P., & Paton, N. W. (2017). Crowdsourcing for data management. *Knowledge and Information Systems, 53,* 1–41.

Dalal, S., Khodyakov, D., Srinivasan, R., Straus, S., & Adams, J. (2011). ExpertLens: A system for eliciting opinions from a large pool of non-collocated experts with diverse knowledge. *Technological Forecasting and Social Change, 78,* 1426.

Deng, X (Nancy)., & Joshi, K. D. (2016). Why individuals participate in micro-task crowdsourcing work environment: Revealing crowdworkers' perceptions. *Journal of the Association for Information Systems 17*, 648–673.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2004). Information systems outsourcing: a survey and analysis of the literature. ACM SIGMIS Database DATABASE Adv. *Information Systems, 35,* 6–102.

Doan, A., Ramakrishnan, R., & Halevy, A. Y. (2011). Crowdsourcing systems on the world-wide web. *Association for Computing Machinery Communications of the ACM, 54,* 86.

Eickhoff, C., & de Vries, A. P. (2013). Increasing cheat robustness of crowdsourcing tasks. *Information retrieval Boston, 16,* 121–137.

Erkinheimo, P., Jussila, J., & Kärkkäinen, H. (2015). Lean start-up companies by means of crowdsourcing. *International Journal of Innovation Management, 3,* 17–20.

Ertekin, S., Rudin, C., & Hirsh, H. (2014). Approximating the crowd. *Data Mining and Knowledge Discovery, 28,* 1189–1221.
Estellés-Arolas, E., & González-Ladrón-de-Guevara, F. (2012). Towards an integrated crowdsourcing definition. *Journal of Information Science, 38,* 189.
Fang, Y., Sun, H., Chen, P., & Deng, T. (2017). Improving the Quality of Crowdsourced Image Labeling via Label Similarity. *Journal of Computer Science and Technology, 32,* 877–889.
Faullant, R., Holzmann, P., & Schwarz, E. J. (2016). Everybody is Invited but not Everybody Will Come—the influence of personality dispositions on users' entry decisions for crowdsourcing competitions. *International Journal of Innovation Management, 20,* 1.
Fink, A. (2013). *Conducting Research Literature Reviews: From the Internet to paper*. LA: Sage Publichsing.
Foody, G. M., See, L., Fritz, S., Van der Velde, M., Perger, C., Schill, C., et al. (2013). Assessing the accuracy of volunteered geographic information arising from multiple contributors to an internet based collaborative project. *Transactions in GIS, 17,* 847–860.
Franke, N., Keinz, P., & Klausberger, K. (2013). "Does this sound like a fair deal?": Antecedents and consequences of fairness expectations in the individual's decision to participate in firm innovation. *Organization Science, 24,* 1495–1516.
Franzoni, C., & Sauermann, H. (2014). Crowd science: The organization of scientific research in open collaborative projects. *Research Policy, 43,* 1.
Fréry, F., Lecocq, X., & Warnier, V. (2015). Competing with ordinary resources. *MIT Sloan Management Review, 56,* 69–77.
Füller, J., Hutter, K., Hautz, J., & Matzler, K. (2014). User Roles and Contributions in Innovation-Contest Communities. *Journal of Management Information Systems, 31,* 273.
Garrigos-Simon, F. J., Galdon, J. L., & Sanz-Blas, S. (2017). Effects of crowdvoting on hotels: the Booking.com case. *International Journal of Contemporary Hospitality Management, 29,* 419–437.
Gefen, D., & Carmel, E. (2008). Is the world really flat? a look at offshoring at an online programming marketplace. *MIS Quarterly, 32,* 367.
Ghezzi, A., Gabelloni, D., Martini, A., & Natalicchio, A. (2018). Crowdsourcing: A review and suggestions for future research. *International Journal of Management Reviews, 20,* 343–363.
Glaeser, E. L., Hillis, A., Kominers, S. D., & Luca, M. (2016). Crowdsourcing city government: Using tournaments to improve inspection accuracy. *American Economic Review, 106,* 114–118.
Goncalves, J., Hosio, S., Rogstadius, J., Karapanos, E., & Kostakos, V. (2015). Motivating participation and improving quality of contribution in ubiquitous crowdsourcing. *Computer Networks, 90,* 34.
Granell, E., Romero, V., & Martínez-Hinarejos, C. D. (2018). Multimodality, interactivity, and crowdsourcing for document transcription. *Computational Intelligence, 34,* 398–419.
Guesmi, S. (2014). Open data, big data and crowdsourcing: emergent mobile apps business models. *Communications and Strategies, 53–72,* 166.
Guo, B., Liu, Y., Wang, L., Li, V. O. K., Lam, J. C. K., & Yu, Z. (2018). Task allocation in spatial crowdsourcing: current state and future directions. *IEEE Internet Things J. 5*, 1749–1764.
Guzmán, M. (2016). *Let's Talk. Nieman Reports, 70,* 62.
Haller, J. B. A., Velamuri, V. K., Schneckenberg, D., & Möslein, K. M. (2017). Exploring the design elements of open evaluation. *Journal of Strategy and Management, 10,* 40–65.
Han, S., Huang, H., Luo, Z., & Foropon, C. (20180. Harnessing the power of crowdsourcing and internet of things in disaster response. *Annals of Operations Research*, 1–16.
Heo, M., & Toomey, N. (2015). Motivating continued knowledge sharing in crowdsourcing. *Online Information Review, 39,* 795–811.
Hofstetter, R., Aryobsei, S., & Herrmann, A. (2018). Should you really produce what consumers like online? empirical evidence for reciprocal voting in open innovation contests. *The Journal of Product Innovation Management 35*, 209–229.

Hoornaert, S., Ballings, M., Malthouse, E. C., & Van den Poel, D. (2017). Identifying new product ideas: waiting for the wisdom of the crowd or screening ideas in real time. *Journal of Product Innovation Management, 34,* 580–597.

Hossain, M. (2013). Open innovation: So far and a way forward. *World Journal of Science, Technology and Sustainable Development, 10,* 30–41.

Hossain, M. (2015). Crowdsourcing in business and management disciplines: an integrative literature review. *The Journal of Global Entrepreneurship Research, 5,* 1–19.

Hossain, M., & Kauranen, I. (2015). Crowdsourcing: A comprehensive literature review. *Strategic Outsourcing: An International Journal, 8,* 2–22.

Howe, M. (1969). The influence of sports on international tourism. *Business and Society, 9,* 38.

Howe, J. (2006). The rise of crowdsourcing. *Wired Magazine 14*.

Huang, Y., Singh, P. V., & Srinivasan, K. (2014). Crowdsourcing new product ideas under consumer learning. *Management Science, 60,* 2138–2159.

Huberman, B. A., Romero, D. M., & Wu, F. (2009). Crowdsourcing, attention and productivity. *Journal of Information Science, 35,* 758–765.

Hutter, K., Füller, J., Hautz, J., Bilgram, V., & Matzler, K. (2015). Machiavellianism or morality: Which behavior pays off in online innovation contests? *Journal of Management Information Systems, 32,* 197.

Jian, L., Li, Z., & Liu, T. X. (2017). Simultaneous versus sequential all-pay auctions: an experimental study. *Exp. Econ., 20,* 648–669.

Jost, P.-J., & van der Velden, C. (2006). Mergers in patent contest models with synergies and sillovers. *Schmalenbach Business Review ZFBF, 58,* 157–179.

Kaganer, E., Carmel, E., Hirschheim, R., & Olsen, T. (2013). Managing the human cloud. *The MIT Sloan School of Management, 54,* 23.

Kajino, H., Arai, H., & Kashima, H. (2014). Preserving worker privacy in crowdsourcing. *Data Mining and Knowledge Discovery, 28,* 1314–1335.

Kannangara, S. N., & Uguccioni, P. (2013). Risk management in crowdsourcing-based business ecosystems. *Technology Innovation Management Review, 3,* 32–38.

Karger, D. R., Oh, S., & Shah, D. (2014). Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research, 62,* 1–24.

Karimi Alaghehband, F., Rivard, S., Wu, S., & Goyette, S. (2011). An assessment of the use of transaction cost theory in information technology outsourcing. *The Journal of Strategic Information Systems, 20,* 125–138.

Kaufmann, W., & Tummers, L. (2017). The negative effect of red tape on procedural satisfaction. *Public Management Review, 19,* 1311–1327.

Kavaliova, M., Virjee, F., Maehle, N., & Kleppe, I. A. (2016). Crowdsourcing innovation and product development: Gamification as a motivational driver. Cogent Business and Management *3*.

Kim, H.-R., Kang, H., & Lee, I.-K. (2016). Image recoloring with valence-arousal emotion model. *Computer Graphics Forum, 35,* 209–216.

Kohler, T. (2015). Crowdsourcing-based business models: How to create and capture value. *California Management Review, 57,* 63–84.

Kohler, T. (2018). How to scale crowdsourcing platforms. *California Management Review, 60,* 98–121.

Kohler, T., & Nickel, M. (2017). Crowdsourcing business models that last. *Journal of Business Strategy, 38,* 25–32.

Koivisto, N. (20120. User driven radical innovations in open innovation, in *ISPIM Conference Proceedings. The International Society for Professional Innovation Management (ISPIM)* (pp. 1–9). Manchester.

Komninos, A., Besharat, J., Ferreira, D., Garofalakis, J., & Kostakos, V. (2017). Where's everybody? Comparing the use of heatmaps to uncover cities' tacit social context in smartphones and pervasive displays. *Information Technology and Tourism, 17,* 399–427.

Kotlarsky, J., Oshri, I., Dibberns, J., Mani, D. (2018). Information systems sourcing. *MIS Quarterly Research Curation article*. Available online on MIS Quarterly Research Curations https://www.misqresearchcurations.org.

Lastovka, M. (2015). Crowdsourcing as new instrument in policy-making: making the democratic process more engaging. *European View, 14,* 93–99.

LaToza, T. D., & van der Hoek, A. (2016). Crowdsourcing in software engineering: models, motivations, and challenges. *IEEE Software, 33,* 74.

Leal Bando, L., Scholer, F., & Turpin, A. (2015). Query-biased summary generation assisted by query expansion. *The Journal of the Association for Information Science and Technology, 66,* 961–979.

Lease, M., & Yilmaz, E. (2013). Crowdsourcing for information retrieval: introduction to the special issue. *Information Retrieval Boston, 16,* 91–100.

Li, D., & Hu, L. (2017). Exploring the effects of reward and competition intensity on participation in crowdsourcing contests. *Electronic Markets, 27,* 199–210.

Linton, J. D., & Thongpapanl, N. T. (2004). Ranking the technology innovation management journals. *Journal of Product Innovation Management, 21,* 123–139.

Liu, H. K. (2017). Crowdsourcing government: Lessons from multiple disciplines. *Public Administration Review, 77,* 656–667.

Liu, T. X., Yang, J., Adamic, L. A., & Chen, Y. (2014). Crowdsourcing with all-pay auctions: A field experiment on taskcn. *Management Science, 60,* 2020–2037.

Liu, S., Xia, F., Zhang, J., Pan, W., & Zhang, Y. (2016). Exploring the trends, characteristic antecedents, and performance consequences of crowdsourcing project risks. *The International Journal of Project Management, 34,* 1625.

Liu, J., Ji, Y., Lv, W., & Xu, K. (2017). Budget-aware dynamic incentive mechanism in spatial crowdsourcing. *The Journal of Computer Science and Technology, 32,* 890–904.

Love, J., & Hirschheim, R. (2017). Crowdsourcing of information systems research. *European Journal of Information Systems, 26,* 315–332.

Lovett, M., Bajaba, S., Lovett, M., & Simmering, M. J. (2018). Data quality from crowdsourced surveys: A mixed method inquiry into perceptions of amazon's mechanical turk masters. *Applied Psychology, 67,* 339–366.

Lüttgens, D., Pollok, P., Antons, D., & Piller, F. (2014). Wisdom of the crowd and capabilities of a few: internal success factors of crowdsourcing for innovation. *Zeitschrift für Betriebswirtschaft, 84,* 339–374.

Mack, T., & Landau, C. (2015). Winners, losers, and deniers: Self-selection in crowd innovation contests and the roles of motivation, creativity, and skills. *The Journal of Engineering and Technology Management, 37,* 52.

Malhotra, A., & Majchrzak, A. (2014). Managing crowds in innovation challenges. *California Management Review, 56,* 103–123.

Malhotra, A., Majchrzak, A., Kesebi, L., & Looram, S. (2017). Developing innovative solutions through internal crowdsourcing. *The MIT Sloan School of Management, 58,* 73–79.

Malone, T., Laubacher, R., Dellarocas, C. (2010). The collective intelligence genome. *MIT Sloan Management Review, 51*, (3) 21–31.

Martinez, M. G. (2015). Solver engagement in knowledge sharing in crowdsourcing communities: Exploring the link to creativity. *Research Policy, 44,* 1419.

Martinez, M. G. (2017). Inspiring crowdsourcing communities to create novel solutions: Competition design and the mediating role of trust. *Technological Forecasting and Social Change, 117,* 296–304.

Martinez, M. G., & Walton, B. (2014). The wisdom of crowds: The potential of online communities as a tool for data analysis. *Technovation, 34,* 203.

Merchant, R. M., Griffis, H. M., Ha, Y. P., Kilaru, A. S., Sellers, A. M., Hershey, J. C., et al. (2014). Hidden in plain sight: A crowdsourced public art contest to make automated external defibrillators more visible. *American Journal of Public Health, 104,* 2306–2312.

Mergel, I., & Desouza, K. C. (2013). Implementing open innovation in the public sector: The case of challenge.gov. *Public Administration Review, 73,* 882–890.

Merz, A. B., Maier, R., Schimpf, R., Seeber, I., Richter, A., Füller, J., & Schwabe, G. (2016). Exploring the effects of contest mechanisms on idea shortlisting in an open idea competition, in Thirty *Seventh International Conference on Information Systems* (pp. 1–18).

Morgan, J., & Wang, R. (2010). Tournaments for ideas. *California Management Review, 52,* 77–97.

Moussawi, S., & Koufaris, M. (2015). Working on low-paid micro-task crowdsourcing platforms: An existence, relatedness and growth view, in *Thirty Sixth International Conference on Information Systems*.

Munro, R. (2013). Crowdsourcing and the crisis-affected community. *Information Retrieval, 16,* 210–266.

Mustafa, S. E., & Mohd Adnan, H. (2017). Crowdsourcing: A platform for crowd engagement in the publishing industry. *Publishing Research Quarterly, 33,* 283–296.

Nagar, Y., De Boer, P., Garcia, A. C. B. (2016). Accelerating the review of complex intellectual artifacts in crowdsourced innovation challenges, in *Thirty Seventh International Conference on Information Systems* (pp. 1–17).

Natalicchio, A., Petruzzelli, A. M., & Garavelli, A. C. (2017). Innovation problems and search for solutions in crowdsourcing platforms—a simulation approach. *Technovation, 64*(65), 28.

Neto, F. R. A., & Santos, C. A. S. (2018). Understanding crowdsourcing projects: A systematic review of tendencies, workflow, and quality management. *Information Processing and Management, 54,* 490.

Nevo, D., & Kotlarsky, J. (2014). Primary vendor capabilities in a mediated outsourcing model: Can IT service providers leverage crowdsourcing? *Decision Support Systems, 65,* 17.

Nevo, D., Kotlarsky, J. (2020). Crowdsourcing as a strategic is sourcing phenomenon: Critical review and insights for future research. *Journal of Strategic Information Systems*.

Nickerson, J. A., Wuebker, R., & Zenger, T. (2017). Problems, theories, and governing the crowd. *Strategic Organization, 15,* 275–288.

Nishikawa, H., Schreier, M., Fuchs, C., & Ogawa, S. (2017). The value of marketing crowdsourced new products as such: Evidence from two randomized field experiments. *Journal of Marketing Research, 54,* 525.

Ntalianis, K., Tsapatsoulis, N., Doulamis, A., & Matsatsinis, N. (2014). Automatic annotation of image databases based on implicit crowdsourcing, visual concept modeling and evolution. *Multimedia Tools and Applications, 69,* 397–421.

Olson, D. L., & Rosacker, K. (2013). Crowdsourcing and open source software participation. *Service Business, 7,* 499–511.

Oosterman, J., Yang, J., Bozzon, A., Aroyo, L., & Houben, G.-J. (2015). On the impact of knowledge extraction and aggregation on crowdsourced annotation of visual artworks. *Computer Networks, 90,* 133.

Pace, A. K. (2008). President's Message. *Information Technology and Libraries, 27,* 2.

Palacios, M., Martinez-Corral, A., Nisar, A., & Grijalvo, M. (2016). Crowdsourcing and organizational forms: Emerging trends and research implications. *Journal of Business Research, 69,* 1834.

Paré, G., Trudel, M.-C., Jaana, M., & Kitsiou, S. (2015). Synthesizing information systems knowledge: A typology of literature reviews. *Information and Management, 52,* 183–199.

Parker, G., Van Alstyne, M., Choudary, S. (2016). Platform Revolution: How Networked Markets are Transforming the Economy, and How to Make Them Work for You. WW Norton.

Piezunka, H., & Dahlander, L. (2015). Distant search, narrow attention: how crowding alters organizations' filtering of suggestions in crowdsourcing. *Academy of Management Journal, 58,* 856.

Piyathasanan, B., Mathies, C., Patterson, P. G., & de Ruyter, K. (2018). Continued value creation in crowdsourcing from creative process engagement. *Journal of Services Marketing, 32,* 19–33.

Poetz, M. K., & Schreier, M. (2012). The value of crowdsourcing: can users really compete with professionals in generating new product ideas? *Journal of Product Innovation Management, 29,* 245.

Pongratz, H. J. (2018). Of crowds and talents: discursive constructions of global online labour. *New Technol. Work. Employ., 33,* 58–73.

Prpic, J., Taeihagh, A., & Melton, J. (2015). The fundamentals of policy crowdsourcing. *Policy and Internet, 7,* 340–361.

Przybylska, N. (2013). Crowdsourcing As A Method Of Obtaining Information. *Foundations of Management, 5,* 43–51.

Radu, R., Zingales, N., & Calandro, E. (2015). Crowdsourcing ideas as an emerging form of multistakeholder participation in internet governance. *Policy and Internet, 7,* 362–382.

Raford, N. (2015). Online foresight platforms: Evidence for their impact on scenario planning and strategic foresight. *Technological Forecasting and Social Change, 97,* 65.

Rowe, F. (2014). What literature review is not: diversity, boundaries and recommendations. *European Journal of Information Systems, 23,* 241–255.

Salk, C. F., Sturn, T., See, L., & Fritz, S. (2017). Limitations of majority agreement in crowdsourced image interpretation. *Transactions in GIS, 21,* 207–223.

Savage, N. (2012). Gaining wisdom from crowds. *Association for Computing Machinery Communications of the ACM, 55,* 13.

Saxton, G. D., Oh, O., & Kishore, R. (2013). Rules of crowdsourcing: Models, issues, and systems of control. *Information Systems Management, 30,* 2–20.

Schemmann, B., Herrmann, A. M., Chappin, M. M. H., & Heimeriks, G. J. (2016). Crowdsourcing ideas: Involving ordinary users in the ideation phase of new product development. *Research Policy, 45,* 1145.

Schlagwein, D., & Bjørn-Andersen, N. (2014). Organizational learning with crowdsourcing: The revelatory case of LEGO. *Journal of the Association for Information Systems, 15,* 754–778.

Schulten, M. B., & Schaefer, F. (2015). Affective commitment and customer loyalty in crowdsourcing: Antecedents, interdependencies, and practical implications. *The International Review of Retail, Distribution and Consumer Research, 25,* 516–528.

Schweisfurth, T., Raasch, C., & Herstatt, C. (2011). Free revealing in open innovation: a comparison of different models and their benefits for companies. *International Journal of Product Development, 13,* 95.

Scoppa, V. (2008). Are subjective evaluations biased by social factors or connections? An econometric analysis of soccer referee decisions. *Empirical Economics, 35,* 123–140.

See, L., Fritz, S., Perger, C., Schill, C., McCallum, I., Schepaschenko, D., et al. (2015). Harnessing the power of volunteers, the internet and Google Earth to collect and validate global spatial information using Geo-Wiki. *Technological Forecasting and Social Change, 98,* 324.

Skarzauskaite, M. (2012). The application of crowd sourcing in educational activities. *Social Technologies 2*.

Smith, D., Manesh, M. M. G., & Alshaikh, A. (2013). How can entrepreneurs motivate crowdsourcing participants? *Technology Innovation Management Review, 3,* 23–30.

Smith, C., Fixson, S. K., Paniagua-Ferrari, C., & Parise, S. (2017). The evolution of an innovation capability: Making internal idea competitions work in a large enterprise. *Research Technology Management, 60,* 26–34.

Soliman, W., & Tuunainen, V. K. (2015). Understanding continued use of crowdsourcing systems: An interpretive study. *The Journal of Theoretical and Applied Electronic Commerce Research, 10,* 1–18.

Steelman, Z. R., Hammer, B. I., & Limayem, M. (2014). Data collection in the digital age: innovative alternatives to student samples. *MIS Quarterly, 38,* 355.

Steiger, D., Matzler, K., Chatterjee, S., & Ladstaetter-Fussenegger, F. (2012). Democratizing strategy: How crowdsourcing can be used for strategy dialogues. *California Management Review, 54,* 44–68.

Stol, K.-J., & Fitzgerald, B. (2014). Two's Company, Three's a Crowd: A Case Study of Crowdsourcing Software Development.

Sun, Y., Wang, N., Yin, C., & Zhang, J. X. (2015). Understanding the relationships between motivators and effort in crowdsourcing marketplaces: A nonlinear analysis. *The International Journal of Information Management, 35,* 267.

Taeihagh, A. (2017). Crowdsourcing: A new tool for policy-making? *Policy Science, 50,* 629–647.

Terwiesch, C., & Xu, Y. (2008). Innovation contests, open innovation, and multiagent problem solving. *Management Science, 54,* 1529–1543.

Thuan, N. H., Antunes, P., & Johnstone, D. (2016). Factors influencing the decision to crowdsource: A systematic literature review. *Information Systems Frontiers, 18,* 47–68.

Volk, T., Keimel, C., Moosmeier, M., & Diepold, K. (2015). Crowdsourcing versus laboratory experiments—QoE evaluation of binaural playback in a teleconference scenario. *Computer Networks 90*, 99.

von Krogh, G., von Haefliger, S., Spaeth, S., & Wallin, M.W. (2012). Carrots and rainbows: motivation and social practice in open source software development. *MIS Q. 36*, 649.

Wan, Y. (2015). The Matthew effect in social commerce. *Electronic Markets, 25,* 313–324.

Wang, R.D. (2010). Essays on competitive strategy and innovation management. ProQuest Diss. Theses. University of California, Berkeley, Ann Arbor.

Wang, L., Liu, W., Jing, N., & Mao, X. (2015). Simultaneous navigation and pathway mapping with participating sensing. *Wireless Networks, 21,* 2727–2745.

Wang, H., Qi, Z., Shi, R., Li, J., & Gao, H. (2017). COSSET + : Crowdsourced Missing Value Imputation Optimized by Knowledge Base. *Journal of Computer Science and Technology, 32,* 845–857.

Wang, Y., Cai, Z., Tong, X., Gao, Y., & Yin, G. (2018). Truthful incentive mechanism with location privacy-preserving for mobile crowdsourcing systems. *Computer Networks, 135,* 32.

Webster, J., Watson, R.T. (2002). Analyzing the past to prepare for the future: writing litearure review. *MIS Q. 26*.

Weiss, M. (2016). Crowdsourcing literature reviews in new domains. *Technology Innovation Management Review, 6,* 5–14.

Wen, Z., & Lin, L. (2016). Optimal fee structures of crowdsourcing platforms. *Decision Sciences, 47,* 820.

West, J., & Bogers, M. (2014). Leveraging external sources of innovation: A review of research on open innovation. *Journal of Product Innovation Management, 31,* 814–831.

Wilson, K. B., Vikram, B., & Samson, D. (2018). Crowdsourcing. *International Journal of Operations and Production Management, 38,* 1467–1494.

Xu, Y., Ribeiro-Soriano, D. E., & Gonzalez-Garcia, J. (2015). Crowdsourcing, innovation and firm performance. *Management Decision, 53,* 1158–1169.

Yank, V., Agarwal, S., Loftus, P., Asch, S. & Rehkopf, D. (2017). Crowdsourced health data: Comparability to a US National Survey, 2013–2015. *American Journal of Public Health 107*, 1283–1289.

Yoo, J., Lee, K., & Choi, M. (2013). Crowdsourcing for device manufacturers in the convergent media industry. *Communications and Strategies* 73–93.

Yuan, S.-T. D., & Ching-Fang, H. (2018). An impactful crowdsourcing intermediary design - a case of a service imagery crowdsourcing system. *Information Systems Frontiers, 20,* 841–862.

Zadeh, A. H., & Sharda, R. (2014). Modeling brand post popularity dynamics in online social networks. *Decision Support Systems, 65,* 59.

Zhao, Y. (Chris), & Zhu, Q. (2016). Conceptualizing task affordance in online crowdsourcing context. *Online Information Review 40*, 938–958.

Zhao, Y., & Zhu, Q. (2014). Evaluation on crowdsourcing research: Current status and future direction. *Information Systems Frontiers, 16,* 417–434.
Zheng, H., Xie, Z., Hou, W., & Li, D. (2014). Antecedents of solution quality in crowdsourcing: The sponsor's perspective. *Journal of Electronic Commerce Research, 15,* 212–224.
Zuchowski, O., Posegga, O., Schlagwein, D., & Fischbach, K. (2016). Internal crowdsourcing: Conceptual framework, structured review, and research agenda. *Journal of Information Technology, 31,* 166–184.

Technology-Mediated Control Legitimacy in the Gig Economy: Conceptualization and Nomological Network

Martin Wiener, W. Alec Cram, and Alexander Benlian

Abstract The rise of the gig economy has become a global phenomenon that encompasses various industries. Instead of hiring full-time employees, gig economy companies 'outsource' work via online platforms to freelance workers who are paid for completing a given task ('gig'). While gig workers are often portrayed as independent contractors, gig firms leverage advanced digital technologies and smart algorithms to exercise control over their freelance workforce, referred to as technology-mediated control (TMC). This independence-control paradox raises interesting questions in terms of how gig workers perceive the legitimacy of such controls. Against this backdrop, this chapter builds on extant research to propose a three-dimensional conceptualization of TMC legitimacy attuned to the unique features of the gig economy: autonomy, fairness, and privacy. On this conceptual basis, the chapter sets forth to start exploring the nomological network of gig workers' perceptions of TMC legitimacy and outlines a set of key antecedents, consequences, and contextual boundary conditions, thereby offering directions for future research in the area.

1 Introduction

In the current digital age, a growing number of workers are no longer employed in traditional, full-time 'jobs'; rather, they work as independent contractors who are

M. Wiener (✉)
Technische Universität Dresden, Chair of Business Informatics, esp. Business Engineering, Helmholtzstr. 10, 01069 Dresden, Germany
e-mail: martin.wiener@tu-dresden.de

W. A. Cram
University of Waterloo, School of Accounting and Finance, 200 University Avenue West, Waterloo N2L 3G1, ON, Canada
e-mail: wacram@uwaterloo.ca

A. Benlian
Technische Universität Darmstadt, Chair of Information Systems & E-Services, Hochschulstr. 1, 64289 Darmstadt, Germany
e-mail: benlian@ise.tu-darmstadt.de

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_16

paid for completing a particular task, or 'gig' (Friedman 2014). Here, it should be noted that the dramatic decline in full-time employment—which has been considered the norm for decades—and the associated rise of the so-called gig economy are not unique to individual countries, but describe a global phenomenon. For example, according to a recent survey, up to 162 million individuals in Europe and the United States engage in some kind of independent work, reflecting 20–30% of the working-age population in those countries (McKinsey Global Institute 2016).

Generally, gig work can be characterized by high work autonomy, payment by task, and a short-term nature of work arrangements (McKinsey Global Institute 2016). While gig work largely shares these characteristics with other forms of temporary work (e.g., De Stefano 2016), what is new is the extent to which gig work is disconnected from the notion of traditional organizations and is managed through online platforms and algorithms (Cherry and Aloisi 2017; Friedman 2014). One prominent example of a gig economy company is the ride-hailing firm Uber, which offers platform-based transportation services by connecting drivers of privately-owned cars with people seeking a ride.[1] Uber drivers—referred to as "partners" by the company—work as independent contractors who are free to set their own schedule and make money on their terms (Rosenblat 2018). However, despite drivers' freelance status, Uber uses advanced digital technologies and smart algorithms to exercise (tight) control over its workforce, referred to as technology-mediated control (TMC) (Cram and Wiener 2020; cf. Constantiou et al. 2017). Specifically, to ensure high service reliability and quality, Uber uses a mobile app that collects detailed data on driver behaviors and leverages algorithmic management capabilities to steer their actions, for instance by enforcing "blind passenger acceptance" (Rosenblat and Stark 2016, p. 3762).

The paradox between independent work and tight control raises interesting questions regarding gig workers' perceptions of TMC legitimacy. For example, in traditional, human-to-human control relationships, the controller's ability to influence controllee behavior is typically derived from legitimate sources of social power (French and Raven 1959), such as the controller's formal position in the organizational hierarchy, formal job descriptions, or legal contracts (Wiener et al. 2016). However, neither of these legitimate power sources arguably applies to the control relationship between Uber and its drivers, nor to controller-controllee relationships observed in other gig-economy contexts (e.g., Airbnb vs. hosts). Against this backdrop, we take into account the unique characteristics of gig work and draw on the organizational control and information systems (IS) security and privacy literatures to propose a multidimensional conceptualization of TMC legitimacy. On this basis, we then set forth to explore the concept's nomological network by outlining and discussing a set of antecedents, consequences, and boundary conditions, which can serve as a basis for future research.

[1]Throughout the chapter, we use Uber as a running example for illustration purposes.

2 Conceptual Foundations

In this section, we introduce key concepts that form a foundation for our work. In particular, we first describe the characteristics that distinguish different types of gig economy platforms. Next, we introduce the concept of technology-mediated control (TMC) and discuss two basic types of how organizations utilize TMC in day-to-day operations.

2.1 Gig Economy Platforms

The term ‘gig economy’ stems from the music industry and originally referred to musicians playing a ‘gig’ in the sense of a one-time performance at a specific location and time. In the current digital age, gig work describes a form of non-standard employment and spans a wide spectrum of work tasks and industries (Friedman 2014). A central characteristic of gig work is that it is supported by, and managed through, online platforms (e.g., Cherry and Aloisi 2017). According to De Groen et al. (2016), gig economy platforms can be differentiated along two dimensions. First, corresponding platforms differ in terms of worker skills required, ranging from low/medium-skilled work (e.g., Uber) to high-skilled work (e.g., Upwork). Second, they differ in terms of services offered, ranging from virtual services that can be provided from everywhere in the world (e.g., Amazon Mechanical Turk) to physical services to be performed at a specific location (e.g., Uber). Further, Constantiou et al. (2017) highlight that gig/sharing economy platforms differ in terms of the level of control exerted by the platform owner (loose vs. tight) and the degree of rivalry among platform participants (low vs. high). Based on these two dimensions, they distinguish among four platform types: *chaperones*, *franchisers*, *gardeners*, and *principals* (see Fig. 1).

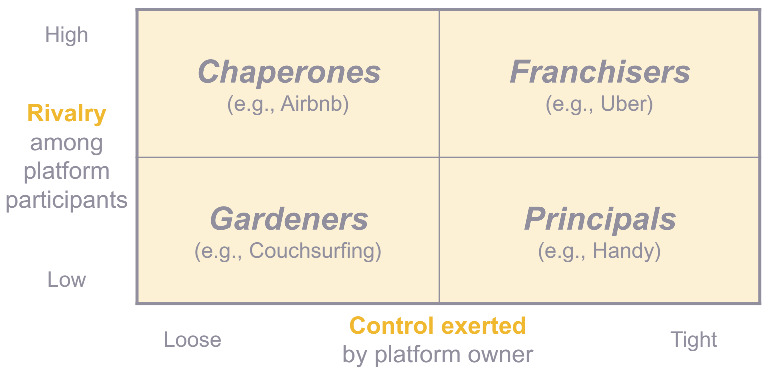

Fig. 1 Typology of gig economy platforms (based on Constantiou et al. 2017, p. 232)

In Constantiou et al.'s (2017) typology, *gardeners* and *franchisers* represent the two extreme cases. Featuring low rivalry and loose control, *gardeners*—such as the peer-to-peer accommodation sharing platform Couchsurfing—focus on supporting the development and maintenance of an online community. In particular, *gardeners* offer an infrastructure for a community to use, while setting only minimum standards for platform participation and fostering low or no rivalry among participants on the supply side (Constantiou et al. 2017). On the other hand, *franchiser* platforms—such as the one operated by the ride-hailing firm Uber—feature high rivalry and tight control. Specifically, in contrast to *gardeners* and *principals* (e.g., Handy), Uber's online platform promotes intense rivalry among drivers who are in direct competition for passengers, while fares are set by Uber based on current market demand and supply. Also, in contrast to *chaperones* (e.g., Airbnb) and *gardeners*, Uber exerts tight control over platform participants in that drivers are required to adhere to strict rules and standard procedures when using the platform to deliver ride-hailing services (e.g., Lee et al. 2015; Möhlmann and Zalmanson 2017; Rosenblat and Stark 2016). These rules and procedures are enforced via Uber's driver app, which is also used to influence when, where, and how long drivers work (Scheiber 2017). The control approach used by Uber thus represents a prime example of technology-mediated control, as discussed in the following section.

2.2 Technology-Mediated Control (TMC)

TMC is broadly defined as the managerial use of advanced digital technologies (e.g., Internet of Things [IoT] sensors, mobile apps, wearable devices) and smart algorithms as a means to influence workers to behave in a way that is consistent with organizational expectations (Cram and Wiener 2020; Wiener and Cram 2017). This view of TMC is largely consistent with past conceptualizations of algorithmic management, which has been characterized by constant tracking of worker behaviors, continuous evaluation of worker performance, and automatic implementation of decisions (Lee et al. 2015; Möhlmann and Zalmanson 2017).[2]

According to Cram and Wiener (2020), two basic types of TMC can be distinguished (see Fig. 2). In the first type, digital technology is used to *support* managerial control processes by acting as a monitoring tool that provides (human) managers with useful insights into subordinate behaviors. For example, the global logistics firm United Parcel Service (UPS) equips its trucks with sensors that collect detailed data about driver behaviors. UPS managers (controllers) then use this data to ensure that drivers (controllees) behave in a manner that is consistent with pre-specified guidelines and rules. Where workers are not adequately complying with the policies,

[2] Whereas algorithmic management specifically considers how the behavior of remote workers is influenced by software algorithms, exclusive of any human intervention (Lee et al. 2015; Möhlmann and Zalmanson 2017), we follow Cram and Wiener's (2020) conceptualization of TMC, which recognizes the potential for technology to *support* the control activities of human managers, as well as the potential to *automatically* act in place of human managers.

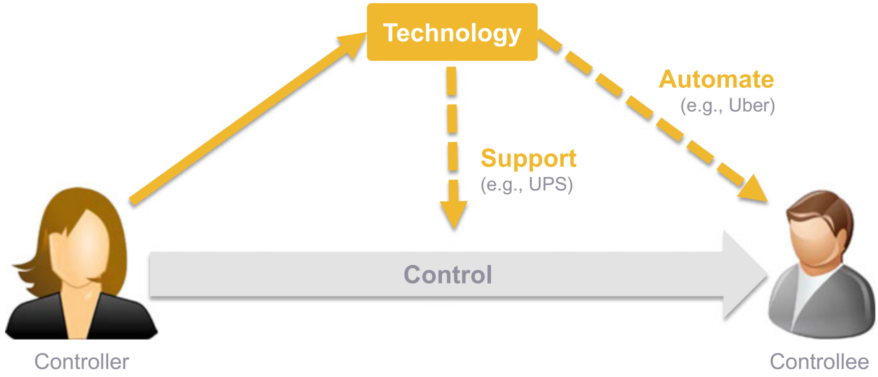

Fig. 2 Basic TMC types: support versus automate (based on Cram and Wiener 2020, p. 74)

the manager can act to correct the behavior. In the second type of TMC, technology is used to *automate* managerial control processes by acting as a proxy for human controllers. This TMC type is commonly used in the gig economy. For example, as noted above, the ride-hailing firm Uber uses intelligent algorithms in combination with a mobile app to guide, monitor, evaluate, and reward or sanction driver behaviors without any substantive human intervention, thereby minimizing the need for costly and time-consuming interactions between Uber managers and drivers (Rosenblat and Stark 2016).

The use of automated TMC approaches shows several unique characteristics that set them apart from automated controls embedded in traditional software systems (e.g., SAP ERP). For example, while enterprise systems are usually limited to the collection of cross-sectional, behavioral data (e.g., who entered what data), the use of TMC relies on ubiquitous technologies, including smartphones and mobile sensors, which are able to capture the minutiae of worker behaviors (Marabelli et al. 2017). Further, system-embedded controls tend to enforce static rules, such as forcing software users to provide certain information before they can move on to the next process step. In contrast, TMC employs dynamic controls and rules that take into account relevant context factors, while still being applied without human involvement. For instance, Uber employs complex algorithms to determine and inform drivers about so-called "surge pricing zones" that require more drivers at a particular point in time due to a temporary spike in customer demand (Rosenblat and Stark 2016). Or, when trying to log off, Uber drivers may receive alerts informing them about being close to achieving the next earning target (Scheiber 2017). Given these characteristics, the organizational use of TMC has the potential to put workers at a considerable disadvantage by creating a number of information and power asymmetries in favor of the platform operators (cf. Rosenblat and Stark 2016). Arguably, this is in sharp contrast to gig economy companies, such as Uber, promising workers a partnership-like relationship and flexible employment (ibid), which in turn raises questions about gig workers' perceptions of TMC legitimacy.

3 Conceptualizing Control Legitimacy in the Gig Economy

Drawing on pioneering work from the sociology and organizational literatures as well as more contemporary perspectives in the related research from IS, this section first reviews different perspectives on legitimacy. Focusing on the specific context of the gig economy, it then introduces the concept of TMC legitimacy and conceptualizes this concept in terms of three focal dimensions.

3.1 Perspectives on Legitimacy

Legitimacy is widely acknowledged to be a socially constructed phenomenon (Brenner and Ambos 2013) that "represents a reaction of observers to the organization as they see it; thus, legitimacy is possessed objectively, yet created subjectively" (Suchman 1995, p. 574). For example, Suchman (1995) defines legitimacy as "a generalized perception or assumption that the actions of an entity are desirable, proper, or appropriate within some socially constructed systems of norms, values, beliefs, and definitions" (p. 574).

Past legitimacy research has adopted a diversity of approaches and perspectives when studying the phenomenon (Deephouse and Suchman 2008). In one approach, legitimacy is considered from the perspective of managers, focusing on the actions they can deploy to enhance legitimacy as means to achieve organizational goals. This compares to a higher-level institutional perspective of legitimacy that is oriented around the societal view of structures and belief systems that lead to cultural pressures within organizations (Suchman 1995). Furthermore, extant research indicates that an organization (as a whole) can be collectively viewed as legitimate by workers (Brenner and Ambos 2013; Dowling and Pfeffer 1975), but that this outcome is, in part, a consequence of individual-level perceptions of lower-level structures, such as top management teams or technology innovations (Deephouse and Suchman 2008). These microfoundations[3] point to the multi-level nature of legitimacy, which past commentators have recognized as forming at an individual level (e.g., an employee perceives a technology innovation as legitimate) and then over time aggregating together to form higher-level macro-judgements on legitimacy (e.g., employees view the overall organization as legitimate) (Deephouse and Suchman 2008; Suddaby et al. 2017). Under this approach, legitimacy can be considered as either a *property* (i.e., a stable, measurable degree of fit between an organization and its environment), a *process* (i.e., an actively changing product of ongoing social interactions), or a *perception* (i.e., an individual-level judgement on whether an organizational action is appropriate) (Suddaby et al. 2017).

[3] In the context of legitimacy, microfoundations represent the perceptions, attitudes, and judgements of individuals. By clarifying the microfoundational legitimacy perceptions of individuals, we can better understand a key antecedent to the collective, macro-level view of organizational legitimacy (Barney and Felin 2013; Suddaby et al. 2017).

In considering how IS scholars have approached legitimacy in the past, it becomes apparent that corresponding studies—including Avgerou (2000), Kohli and Kettinger (2004), and Mehrizi et al. (2019)—primarily incorporate legitimacy fundamentals as a perspective for evaluating employee perceptions related to new technologies. However, one emerging area of study within the organizational literature that has not yet been widely applied to IS research is the study of *control* legitimacy (Cram and Wiener 2018; cf. Bijlsma-Frankema and Costa 2010), which adopts a narrower view of legitimacy by focusing specifically on the collection of controls enacted by managers as the driver of employee perceptions. The value in studying control legitimacy stems from the resulting implications, which include the potential for increased employee trust (Sitkin and George 2005), compliance (Bijlsma-Frankema and Costa 2010), job satisfaction (Niehoff and Moorman 1993), and organizational success (Meyer and Rowan 1977). The study at hand seeks to build on this emerging perspective by focusing on the concept of TMC legitimacy in the specific context of the gig economy.

3.2 TMC Legitimacy Concept and Dimensions

Within the gig economy context of this research, we adopt a managerial perspective on legitimacy and define TMC legitimacy, in a broad sense, as an individual gig worker's general *perception* that the TMC approach used by a given gig company is appropriate to guide day-to-day worker behavior. The concept of TMC legitimacy thus suggests that gig workers do not blindly follow the enacted controls, but instead make judgements on the controls used by the platform operator (cf. Brenner and Ambos 2013; Schnedler and Vadovic 2011; Yang 2015).

Prior studies have identified several dimensions, or sources, that contribute to the formation of an individual's perception of (control) legitimacy. For example, focusing on traditional control relationships between managers and employees, Bijlsma-Frankema and Costa (2010) identify four sources of control legitimacy perceptions (autonomy, competence development, group identification, justice/fairness). They argue that an employee's perception of the legitimacy of a given control, or set of controls, is a direct consequence of a combined interpretation to what extent that control positively contributes to these four sources. However, Bijlsma-Frankema and Costa (2010), as well as other studies (e.g., Suchman 1995), also recognize that the conceptualization of legitimacy is dependent on the type of legitimacy being examined (e.g., control vs. institutional) and the work context.

Against this backdrop, and considering the individual-level focus that commonly defines gig economy work (in comparison to the increasingly team-oriented focus that exists in organizational processes, such as systems development projects), we determined that group identification is of only limited importance in this work context, where arguably all of the work is completed in isolation from co-workers (Anthes 2017). In addition, we reason that because the TMC approaches used on gig economy platforms are often very prescriptive and explicit, there is relatively little opportunity for extensive competence development in gig work. For example, an Uber driver is instructed what kind of car is required, where to pick up passengers, and what route to take. On the other hand, keeping in mind our study's focus on gig workers' individual perceptions of TMC legitimacy, we were able to derive a set of three legitimacy dimensions that clearly apply within a gig economy context. Although we acknowledge that other dimensions might contribute to TMC legitimacy perceptions as well, we believe that the three dimensions noted below—autonomy, fairness, and privacy—are the most prominent and relevant in this context (see also Table 1 for an overview and brief description of each dimension). This viewpoint is consistent with Bijlsma-Frankema and Costa (2010), who argue that only a few sources of legitimacy tend to be the most prominent within a particular context.

First, autonomy contributes to perceptions of control legitimacy when controls are viewed as empowering workers and allowing them to act freely and independently (Bijlsma-Frankema and Costa 2010). Grounded in self-determination theory and empowerment theory, autonomy provides employees with a sense of freedom and competence (Niehoff and Moorman 1993). Relatedly, Suddaby and Greenwood (2005) recognize the importance of autonomy for employees by providing them with a sense of identity. In doing so, where (gig) workers view controls as allowing them to act with autonomy, it contributes to their overall perception of control legitimacy and can lead to improved work quality and continuance intentions (cf. Goldbach et al. 2018). Although flexible work hours are often cited by gig economy workers as an enticing characteristic of the overall job (Kessler 2016; Rosenblat and Stark 2016), TMC is generally oriented towards reducing the opportunity for workers to act independently of the platform-based app (Möhlmann and Zalmanson 2017). For example, the day-to-day activities of workers are typically monitored closely by

Table 1 TMC legitimacy dimensions

Dimension	Brief description	Illustrative example	Relevant literature base
Autonomy	Controls are perceived to empower workers, allowing them to act freely and independently	An Uber driver values the flexibility of working when and where she or he chooses	Organizational control
Fairness	Controls are perceived as fair, just, and reasonable	An Uber driver perceives the passenger ratings to be an effective, fair, and reasonable feedback mechanism	Organizational control IS security and privacy
Privacy	Controls are perceived as appropriately respecting personal worker information during its collection, use, and storage	An Uber driver feels that the location and app-usage information collected by the company is relevant to her/his daily work and will contribute to better customer service	IS security and privacy

the app and automated guidance is provided where non-compliance with accepted practices is noted (Addady 2016, Griffin 2016).

Second, fairness is commonly identified as a key contributor to perceptions of control legitimacy (e.g., Bijlsma-Frankema and Costa 2010; Brockner et al. 2001; Long et al. 2011; Ouchi 1980; Sitkin and George 2005). In this context, managers implementing controls that employees judge to be fair, rational, and reasonable, will contribute to establishing a routine of organizational interactions and allow employees to make sense of the organizational structures around them. Where employees believe they are being treated equitably on a day-to-day basis, they attribute legitimacy to the control (Ouchi 1980). This view is grounded in the workplace justice and employee citizenship behavior literature (Bijlsma-Frankema and Costa 2010; Long et al. 2011). Within gig economy work, a foundational assumption surrounding fairness perceptions is that platforms often claim to be in a partnership-like relationship with its workers (Rosenblat and Stark 2016). As such, workers often have expectations that the controls enacted by the platform app will treat them in an equitable way. Where platforms do not provide sufficient transparency into their business practices or how the underlying algorithms operate, workers may perceive controls as unfair. At Uber, for example, drivers commonly view the ratings system and fare calculations as being obscure and unreasonable (Chan and Humphreys 2018; Möhlmann and Zalmanson 2017).

Third, privacy refers to an individual's ability to decide who gets access to personal information and what is done with that information (Smith et al. 1996; Stone et al. 1983). With very few exceptions, such as Alge et al. (2006) and Posey et al. (2011), privacy has not been considered as a key contributor to control legitimacy perceptions. However, due to the unique context of the gig economy, the heavy reliance on monitoring worker behavior (Addady 2016; Goldstein 2014; Woyke 2018) has introduced a growing collection of granular and real-time worker data, including location, response time (e.g., how long it takes a worker to respond to a request), and mobile device usage (e.g., texting while driving). Past research finds that workers perceive their privacy expectations to be met when they conclude that controls are in place to ensure that: (a) the personal information being collected relates to an existing relationship; (b) workers have the ability to decide how the information is used; (c) the information is relevant to a transaction; and (d) the information will translate into reliable judgements (Culnan and Armstrong 1999). Where organizations have fulfilled these information management expectations, workers will increasingly view the controls as legitimate (Alge et al. 2006; Posey et al. 2011). However, where an organization provides insufficient notice (e.g., lack of clarity on why personal information is being collected or what it will be used for), consent (e.g., worker is unable to opt out), or is seen to use the data in an inappropriate way (e.g., intrusive data collection, not used for a relevant business decision), individuals are increasingly likely to perceive the company's activities as invasive and illegitimate (Bies 1993; Culnan and Armstrong 1999; Malhotra et al. 2004; Marx and Sherizen 1987). We consider the privacy dimension to be distinct from the fairness dimension on the basis that it is oriented specifically around the controls associated with the appropriate use and safeguarding of personal data. This compares to the fairness dimension, which is concerned with more broad-based judgements on the overall equitability and reasonability of controls.

4 Exploring the Nomological Network of TMC Legitimacy

Based on the three-dimensional conceptualization of TMC legitimacy introduced above, this section presents an initial exploration of the concept's nomological network.[4] In particular, again using the gig economy firm Uber as an illustrative example, we first explore the role of different (formal and informal) control modes as key antecedents of TMC legitimacy. Next, we shed light on key behavioral consequences related to the dichotomies between continuance and turnover intentions, as well as between control compliance and violation. Lastly, we look into contextual boundary conditions that are likely to influence workers' perceptions of TMC modes and legitimacy, as well as their downstream effects.

4.1 Antecedents of TMC Legitimacy

Formal TMC modes: To exercise control over their freelance workforce, gig companies use TMC approaches that typically rely on a combination of all three basic modes of formal control: input control, behavior control, and output control (e.g., Jaworski 1988; Wiener et al. 2016). For example, before accepting a new driver, Uber requests formal documentation from each applicant (e.g., proof of residency and vehicle insurance), checks the candidate and her/his car against a list of formal requirements, and conducts background checks (***input control***) (Lee et al. 2015; see also Uber 2019a). Here, it should be noted that, more recently, Uber started running background checks each year in an effort to "ensure drivers continue to meet [its] standards on an ongoing basis, long after they take their first trip" (Khosrowshahi 2018). Relatedly, Uber invested in new technology that "can identify new criminal offenses via public records or pending DUI [driving under the influence] charges as they happen" (O'Brien 2018). In relation to ***output control***, Uber drivers are rated by passengers on a five-point scale after each and every trip. These ratings arguably constitute the "most significant performance metric" (Rosenblat and Stark 2016, p. 3772), not least because drivers are required to maintain a minimum rating of around 4.6 in order to remain active (ibid). This suggests that Uber's driver rating feeds back into input control. In terms of ***behavior control***, Uber sends regular feedback messages to each driver, including reports on driver-specific issues (e.g., concerns about driving safety and professionalism) along with suggestions on how a

[4]In line with the definition provided above (see Sect. 3.2), we acknowledge that when referring to TMC legitimacy, it implies the perception of TMC legitimacy by an individual gig worker.

driver can improve their customer ratings. For example, drivers may receive a message such as the following: "*Riders give the best ratings to drivers who [go] above and beyond to make the experience special, such as opening doors for riders when possible*" (Rosenblat and Stark 2016, p. 3776).

Taking into account the unique characteristics of algorithmic management, we argue that gig workers' perceptions of the three formal TMC modes (refer to Table 2 for brief definitions and illustrative examples) can be expected to vary considerably for at least two reasons. First, due to individual preferences, personality traits and other context factors, gig workers are likely to perceive the very same control mechanism (e.g., the Uber driver rating) very differently (e.g., as loose vs. tight control). Second and relatedly, since automated TMC approaches mimic the behavior of a human controller, the perceived control degree will also vary for objective reasons. For example, high-performing Uber drivers will arguably receive, and thus most likely perceive, less control than low-performing drivers.

Further, we argue that the control logic that underlies input control is fundamentally different from the logic that underlies behavior and output control. In particular, while the former is used to determine *who* is allowed to participate, the latter explicates *what* participants should do and *how* they should behave. This key difference in underlying control logic can be expected to have a notable effect on how drivers perceive the control modes, as well as their legitimacy. For example, given its coercive design (cf. Adler and Borys 1996), input control is likely to be perceived as a 'constant threat' by Uber drivers. In contrast, providing drivers with clear and direct

Table 2 Perceptions of formal TMC modes

Antecedent	Definition	llustrative example
Perceived input control	The degree to which a gig worker perceives that a given online platform operator uses gatekeeping and screening procedures to allow her/him to work, or to continue working, on the platform (adapted from Croitor and Benlian 2019; Tiwana 2015)	Through the driver app, Uber requests formal documentation from each potential driver (e.g., proof of residency and vehicle insurance); also, the company uses advanced technology to continuously monitor each driver's criminal record
Perceived behavior control	The degree to which a gig worker perceives that the platform operator oversees and guides her/his work behaviors (cf. Goldbach et al. 2018)	Through the driver app, Uber offers suggestions to drivers on behaviors that have been found to increase customer satisfaction (e.g., opening doors, playing jazz music, etc.)
Perceived output control	The degree to which a gig worker perceives that the platform operator monitors and provides feedback on her/his work performance (cf. Goldbach et al. 2018)	Through the driver app, Uber provides drivers with performance metrics, including average customer ratings as well as ride acceptance and completion rates

feedback on their performance along with suggestions on how to improve it, behavior and output controls may be perceived as empowering, or enabling (ibid), thereby contributing to driver perceptions of autonomy. In addition, while earlier studies highlight that algorithmic management in general is characterized by low transparency (Möhlmann and Zalmanson 2017), we argue that the level of transparency can differ noticeably across formal TMC modes, leading to distinct effects on gig workers' perceptions of TMC legitimacy. For example, driver ratings and other output controls (e.g., acceptance rates) are communicated clearly and are thus well known among Uber drivers. In this regard, it has been argued that automated controls that adhere to a generally accepted set of rules increase transparency, leading to perceptions of fairness among controllees (Hansen and Flyverbom 2015; Möhlmann and Zalmanson 2017). For example, when Uber drivers receive a ride request through the app, they are well aware that they have around 15 seconds to accept or reject it (Rosenblat and Stark 2016; see also Uber 2019b). Also, while the Uber driver app collects detailed data on driver behaviors for control purposes (Cram and Wiener 2020), Uber openly shares information on what and how data are collected on its website (e.g., Beinstein and Sumers 2016). In contrast, Uber drivers tend to perceive some of the enacted input controls, and in particular the required background checks, to be an invasion of their privacy. Among other things, this is because it is not entirely clear to drivers how Uber conducts these background checks, and especially what data are collected and analyzed (O'Brien and Yurieff 2017).

However, at this point, we would like to acknowledge that, especially in the specific case of Uber, one could also argue that the behavior and output controls enacted through the driver app follow a coercive logic as well, along with high control frequency and intensity (cf. Cram and Wiener 2018; Gregory et al. 2013). For example, feedback reports delivered via the driver app may be perceived as overly tight, as they are sent out at least weekly and include very detailed directions on how an Uber driver should behave on the job (behavior control), thereby restricting a driver's perceived autonomy. The latter is in keeping with earlier research, which finds that tight behavior control is associated with controllee perceptions of low autonomy (Cram and Wiener 2018). In addition, while the basic (behavioral) rules associated with enacted behavior and output controls are clearly communicated to drivers, Uber does not disclose the 'rules' that underlie those controls (e.g., who receives what suggestions). Also, the underlying rules are adaptive in nature and thus frequently change (Rosenblat and Stark 2016). In such situations, as noted by Möhlmann and Zalmanson (2017), control transparency in algorithmic management is generally low, prompting Uber drivers to question the fairness of the enacted controls. Moreover, some behavior controls (e.g., 15-second time limit to accept or reject a ride request) put considerable pressure on drivers, while leveraging information asymmetries in favor of the platform owner (Rosenblat and Stark 2016). For example, when Uber drivers receive a ride request through the app, they are not shown relevant information (e.g., trip destination, fare estimate) until they accept the request, even though this information is already known by the company (ibid.). Finally, not only input control (see above) but also behavior and output controls may be perceived as an invasion to driver privacy. In this regard, related research finds that the use of behavior controls—which is closely linked to, and often based on, worker monitoring/surveillance—is particularly prone to triggering privacy concerns (e.g., Moussa 2015). Arguably, this general observation translates to the specific control context of Uber, where the driver app is used to collect detailed data on driver behaviors for control purposes (Cram and Wiener 2020). For example, the company uses "harsh braking and acceleration as indicators of unsafe driving behavior" (Beinstein and Sumers 2016; see also Scheiber 2017).

Informal TMC modes: In addition to gig workers' perceptions of the three formal TMC modes discussed above, prior studies on gig economy platforms indicate that TMC legitimacy dimensions, such as autonomy, are influenced by a worker's perceived degree of ***self-control*** (e.g., Goldbach et al. 2018; Lehdonvirta 2018). Generally speaking, self-control is an informal control mode that relies on a worker's intrinsic motivation and individual standards (e.g., Jaworski 1988). More specifically, in self-control, gig workers (controllees) set their own goals, define the actions required to achieve these goals, and self-monitor their behavior (e.g., Kirsch 1996; Wiener et al. 2016). While Uber's business model builds on the promise of flexible employment, its tight TMC approach arguably leaves little room for self-control (cf. Constantiou et al. 2017; Rosenblat and Stark 2016). Still, prior research indicates that gig economy companies use their TMC approach to promote and facilitate the exercise of self-control by providing workers with information and tools (e.g., earning tracker) for self-organization and self-regulation (Goldbach et al. 2018; Lehdonvirta 2018). For example, analyzing the TMC approach used on Google's mobile software platform, Goldbach et al. (2018) find that Android app developers' perceptions of self-control are positively and significantly related to their perceived autonomy.

A second informal control mode is ***clan control***, which operates when goal-direct behavior in a peer group is guided by shared norms and values, along with a common vision (Kirsch et al. 2010; Wiener et al. 2016). When compared to the three formal TMC modes and self-control, clan control appears to be less pertinent on many gig economy platforms because workers, such as Uber drivers, tend to work in isolation from one another (i.e., not as part of a team or peer group). However, we recognize that several gig economy platforms do employ a model that draws on ad hoc group work to perform tasks, referred to as crowdsourcing, which in some scenarios makes clan control increasingly relevant. Specifically, while crowdsourcing platforms such as Amazon Mechanical Turk or 99Designs focus on micro-tasks (i.e., small-scale, simple activities) that are completed independently by individual workers and then consolidated together, other platforms—such as 10EQS and OnFrontiers—pursue the crowd-based completion of macro-tasks, which represent more complex activities and often, though not always, include worker collaboration (e.g., Robert 2019; Schmitz and Lykourentzou 2018). For those platforms that engage in collaborative macro-tasks, clan control has the potential to play an important role for workers, as there may be a need to interact and establish social norms and values.[5] Here, platform features such as FAQ lists, work-progress trackers, and peer ratings (including 'voting a member out' functionality) can promote the exercise of (social) clan control by enabling peers to guide, monitor, and sanction others' behavior (Gallivan 2001). However, the question of how the enactment of clan controls relates to gig workers' perception of TMC legitimacy appears to be anything but clear-cut. One possibly important clue to solving this 'puzzle' relates to Gallivan's (2001) distinction between core and peripheral workers (observed in the context of open source software projects), which may also apply to group work on crowdsourcing platforms. In particular, a small core group of gig workers may play a decisive role in shaping and enacting team-based clan controls (and thus perceive these controls to be legitimate), whereas workers not being part of this core group may only 'react' to those controls and thus perceive them to be illegitimate. The latter can be explained by what Lowry and Moody (2015) term reactance theory, which defines reactance as "a negative emotional response caused by threats to or losses of behavioral freedom (often resulting from a persuasion [or control] attempt by another party)" (p. 439). Here, Lowry and Moody emphasize that a person's expectation of behavioral freedom (e.g., an expectation of not being controlled by her or his peers) is a key condition for the occurrence of reactance.

4.2 Consequences of TMC Legitimacy

Continuance versus turnover intentions: Gig economy companies rely on workers' willingness to be part of, and keep contributing to, a specific ecosystem (Boudreau 2012; Ceccagnoli et al. 2012). In this context, Goldbach et al. (2018) define gig workers' continuance intentions as their behavioral intention to remain part of a platform ecosystem and offer their services on that platform. Here, the importance that gig economy workers place on securing flexible work arrangements is consistent with the historical preferences of independent contractors (Hall and Krueger 2015). As a result, the flexibility, or autonomy, that gig workers have in setting their own schedules represents an important recruitment and retention mechanism for companies such as Uber (Lee et al. 2015; Möhlmann and Zalmanson 2017; Rosenblat and Stark 2016). For example, in a 2015 survey commissioned by Uber, 85% of respondents agreed that flexibility was a major motivator for driving for the company, while 42% of women and 29% of men stated that a flexible schedule was mandatory for them, due to family, education, or health factors (Hall and Krueger 2015).

As well, existing literature establishes a range of benefits that result from worker perceptions of fairness. From a continuance perspective, workers who believe they will be treated fairly are able to more clearly anticipate the long-term benefits of

[5]Robert (2019) finds that the vast majority of past crowdsourcing research focuses on the performance of micro-tasks and includes no discussion of informal controls.

remaining with an organization (Cropanzano et al. 2007). This is supported by organizational research, which finds a positive relationship between fairness/justice and organizational commitment (Folger and Konovsky 1989) and a negative relationship between fairness and turnover intentions (Alexander and Ruderman 1987; Chalykoff and Kochan 1989). In a similar vein, prior research finds that organizational processes—which includes managerial control processes—that are perceived by workers as fair contribute to the building of trust and commitment, and eventually lead to voluntary cooperation (Cropanzano et al. 2007).

Further, with regard to privacy, gig workers are often well aware of firms' surveillance capabilities but still seem to perceive them differently. For example, some drivers value Uber monitoring their behavior as it can act as a means to adjudicate disputes with riders, as noted in an interview conducted by Rosenblat and Stark (2016), where a driver highlighted that Uber can "actually log the exact route that you took" (p. 3765). However, generally, in cases where workers perceive their privacy has been infringed upon by an organization, existing research suggests that lower levels of organizational commitment and increased turnover will result (Smith and Tabak 2009; Tabak and Smith 2005). For example, prior studies find that constant tracking and close monitoring can lead to feelings of anxiety and tension (e.g., Lee et al. 2015; Möhlmann and Zalmanson 2017), or may be perceived by workers as an indication of a lack of trust (Smith and Tabak 2009). In this regard, Pfaffenberger (1992) notes that "the surveillance systems used to track the performance of airline telephone reservation clerks […] were constructed with the designers' conscious assumption that the clerks have little loyalty to the firm, are poorly educated, will try to avoid giving good service, and will quit in a few months anyway; the surveillance tries to transform them into tractable, cooperative cogs in a smooth-running machine" (p. 283).

Control compliance versus violation: In general, controls that are viewed as legitimate by gig workers are expected to be increasingly accepted and followed by workers, while controls that are viewed as illegitimate will be at increased risk for non-compliance (Bijlsma-Frankema and Costa 2010). For example, studying IS security policies and conceptualizing legitimacy in terms of perceived fairness (i.e., the extent to which such policies are perceived to be appropriate and just), Son (2011) finds that perceived legitimacy has a significant and positive relationship with policy compliance. In other words, workers that are treated more justly are more likely to comply with corporate policies and less likely to engage in counter-productive work behavior (Cohen-Charash and Spector 2001). Therefore, "if the process is perceived as just, employees show greater loyalty and more willingness to behave in an organization's best interests. They are also less likely to betray the institution and its leaders" (Cropanzano et al. 2007, p. 38). Relatedly, past research finds that where controllees perceive their need for autonomy to be met, they are more willing to pursue managerial goals (Bijlsma-Frankema and Costa 2010; Williams and Deci 1996). For example, in the case of Uber, these goals revolve around providing fast and professional services to riders.

On the other hand, the use of workarounds is of particular concern for organizations, as it represents a form of control violation and leads to managers being increasingly unaware of how work is actually being undertaken by workers (Woltjer 2017). Laumer et al. (2017) define workaround use as a worker's "conscious adaptations of work activities that are not expected or specified to be changed in this manner" (p. 335). Here, previous research shows that, where organizational and personal interests are aligned, inappropriate workarounds are less likely to be undertaken (Alter 2014). However, even if interests are aligned, gig workers can still be expected to engage in workarounds if their perceptions of the autonomy they are granted differ from what they had been promised. To that end, Ferneley et al. (2004) argue that "often employees will resist and, if possible, ignore a system which does not allow them the level of discretion and autonomy they see as part of their profession" (p. 1005). Similarly, Pollock (2005) argues that workarounds are employed as "resistance on behalf of users and the means by which they attempt to wrest control back from a technology or an institution" (p. 497). For instance, Rosenblat and Stark (2016) note that some drivers "perceive that Uber favors the passenger in adjudications, and even report having to gather their own data to prevent wages from being retracted" and that, in response, drivers may "resist Uber's power of interpretation by tracking their trips with manual or electronic logs and dash-cams" (p. 3765). Other evidence suggests that some drivers attempt to manipulate their interactions with the app (e.g., turning it off after a long ride, so as not to get too far away from home) when they perceive its guidance to be unfair or inconvenient (Lee et al. 2015).

Other consequences: Going beyond the TMC legitimacy consequences discussed above, past research supports the general assertion that workers who perceive organizational activities as being legitimate will not only behave differently but also perform differently than those who perceive a low level of legitimacy. In particular, perceptions of high legitimacy have been found to be associated with improved job satisfaction (Long et al. 2011) and commitment to team goals (Niehoff and Moorman 1993), while low legitimacy perceptions have been linked to reduced effort (ibid) and decreased work performance. For instance, existing research suggests that, where workers perceive their privacy has been infringed upon by an organization, not only increased turnover (see above) but also decreased performance will result (Smith and Tabak 2009; Tabak and Smith 2005).

4.3 Contextual Boundary Conditions

When exploring a concept's nomological network, it also becomes important to highlight relevant boundary conditions, and in particular to discuss the contextual conditions under which the proposed relationships are most likely to hold (cf. Rivard 2014; Whetten 2002). In this regard, a variety of contextual factors can be expected to influence gig workers' TMC legitimacy perceptions, or to moderate, the relationships between TMC modes and legitimacy, on one hand, and TMC legitimacy and the consequences outlined above, on the other hand. In the following, we focus our discussion on three broad sets of context factors that seem to be particularly relevant in the specific context of the gig economy, namely: worker attitudes and types, as well as platform features.

Gig worker attitudes: A prime example of an attitudinal context factor that is likely to moderate the link between (formal) TMC modes and legitimacy is a gig worker's ***attitude toward gamification***. In the literature, gamification has often been defined as "a process of enhancing services with (motivational) affordances in order to invoke gameful experiences and further behavioral outcomes" (Hamari et al. 2014, p. 2; see also Hamari 2013). For example, some of the gamification elements embedded in the Uber driver app are quite literal: Like video game players, Uber drivers can earn badges for achievements, such as "neat and tidy", "excellent service", and "great amenities" (Uber 2019c). Although the integration of gamification elements into online platforms, and TMC approaches in particular, is a fairly common strategy among gig economy companies (e.g., Scheiber 2017), individual workers' attitude toward such a gamification strategy can be expected to vary considerably (cf. Broer 2017; Tomaselli et al. 2015); that is, while some gig workers may feel negatively about game-like platform features and game-based competition, others seem to enjoy these features. For example, in an interview with the *New York Times*, an Uber driver expressed his pride in the badges he had earned and highlighted the important feedback role that they fulfill: "*It tells me where I'm at*" (Scheiber 2017).

Gig worker types: The extent to which a gig worker perceives a given TMC approach to be legitimate might differ between ***full-time and part-time workers***. In particular, while the former make a living from their gig work, and are thus highly dependent on this work, the latter often 'just' use the income from their gig work to supplement the income from a regular job. A similar pattern may apply to ***high- and low-income workers***. For example, Rosenblat and Stark (2016, p. 3763) point out that the "rhetoric of risk and reward has been retooled to suit a contingent of lower-income workers who are recruited to perform service labor under working conditions controlled by the design and affordances of Uber's platform." In essence, this suggests that low-income drivers are more likely to perceive Uber's TMC approach to be legitimate. On a related note, TMC legitimacy perceptions may vary between workers being active on a single gig platform (e.g., Uber) and workers being active on multiple platforms simultaneously (e.g., Uber and Lyft), also referred to as ***single- versus. multi-homing***. In this regard, Möhlmann and Zalmanson (2017) find that many ride-hailing drivers work for more than one company and switch between platforms to regain control: "This switching behavior provides drivers with leverage against the platform by lowering the risks associated with a ban from existing platforms and allowing them to threaten to or actually abandon the Uber platform" (p. 12).

Another gig worker-related context factor that can be expected to moderate the TMC legitimacy relationships discussed in Sects. 4.1 and 4.2 is the level of ***worker experience***. For example, an Uber driver with several years of experience and thousands of completed rides is likely to interact with the app, and view the controls embedded in this app, differently than a driver with only a few weeks of work experience. Experience-induced differences in TMC legitimacy perceptions seem to be of particular relevance in a gig economy setting due to the relatively high turnover rates (Rosenblat 2018). Relatedly, extensive experience as a gig economy worker allows for an increased opportunity to learn the 'tricks' of how best to utilize the platform, as well as when and how to use workarounds to circumvent or trick the control algorithms. Also, experienced Uber drivers will be savvier in understanding the specific implications of control violations, such as failure to meet the minimum acceptance rates.

Gig platform features: Extant research highlights that gig economy platforms differ sharply in terms of the 'actual' ***degree of control*** exerted by the platform owner (Constantiou et al. 2017). This, in turn, implies that TMC legitimacy perceptions are likely to differ across platforms as well. For example, being characterized by tight control and high rivalry among participants, *franchiser* platforms such as Uber appear to be particularly prone to TMC legitimacy concerns, especially when compared to *gardener* platforms (loose control and low rivalry) (ibid). Similar differences in legitimacy perceptions might be found between gig economy platforms that focus on the mediation of ***high-skilled work*** (e.g., Upwork) and platforms focusing on the mediation of ***low-skilled work*** (e.g., Uber) (De Groen et al. 2016), with low-skilled workers arguably being more likely than high-skilled workers to perceive the use of TMC to be legitimate.

Finally, as noted above, for gig economy firms, the use of gamification elements is a quite common strategy to invoke desired worker behaviors and positively influence their work experience. Still, the actual ***degree of gamification*** can vary considerably across gig economy platforms. For example, when asked about the driver app, a veteran Uber driver stated, "*The whole thing is like a video game*" (Scheiber 2017). By gamifying the driving experience, Uber gives drivers the impression that their 'destiny' is in their own hands, fueling perceptions of high autonomy. For instance, the app shows drivers the number of trips they have completed in the current week, the money they have earned, and the time they have spent logged on. Conveying a feeling of work autonomy, all of these video-game like metrics "can stimulate the competitive juices that drive compulsive game-playing" (Scheiber 2017). Adding to this, the gamification elements included in Uber's driver app have the potential to alleviate drivers' privacy concerns by putting them into a mental state known from video gaming and referred to as "ludic loop" (Scheiber 2017). On this basis, it can be argued that the deployment of a high gamification degree is an effective strategy in masking the exercise of (tight) control, as well as in trivializing the associated collection of sensitive data, and thus in increasing gig workers' perceptions of TMC legitimacy. This is largely in keeping with Rosenblat and Stark's (2016) argument about the "prominence of control [being] not as perceptible" for Uber drivers and other gig workers whose work is primarily mediated electronically.

Taken together, the key antecedents, consequences, and boundary conditions of TMC legitimacy can be integrated into a nomological network (see Fig. 3) that can guide future research in the area.

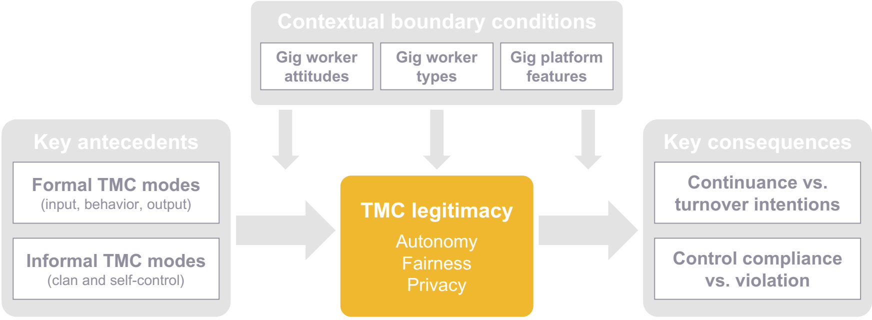

Fig. 3 Nomological network of TMC legitimacy

5 Conclusion

In this chapter, we build on extant research to derive a three-dimensional conceptualization of TMC legitimacy (i.e., in terms of autonomy, fairness, privacy) attuned to the specific context and unique challenges of the gig economy. Future research can use this conceptualization to extend prior studies on the gig economy, which have typically focused on the first legitimacy dimension, namely, perceived autonomy (e.g., Goldbach et al. 2018; Möhlmann and Zalmanson 2017). Further, by exploring the nomological network of TMC legitimacy (perceptions), and in particular by identifying a set of key antecedents, consequences, and contextual boundary conditions, our study provides direction and inspiration for future research on the control strategies used by gig companies. For example, our study contributes to extant research on platform/ecosystem 'health' by pointing to the importance of TMC legitimacy in explaining gig workers' continuance or turnover intentions, on one hand, and their control compliance or violation (including the use of workarounds), on the other hand (cf. Benlian et al. 2015; Iansiti and Levien 2004).

On this basis, our study also offers important implications for practice. Most notably, low retention rates represent a key challenge for Uber and other gig economy platforms. For example, an Uber-internal study shows that only about 55% of the drivers "who started in the first half of 2013 remained active a year after starting" (Hall and Krueger 2015, p. 16). For gig companies, it is thus imperative to understand what factors influence workers' continuance intentions. As such, the concept of TMC legitimacy along with the theoretical relationships proposed in this chapter offer important 'food for thought' on how gig economy companies can improve the design of their platforms whose success ultimately depends on an effective TMC approach and a 'happy' workforce that perceives this approach to be legitimate.

Acknowledgements Financial support from Bentley University's Data Innovation Network and the Deutsche Forschungsgemeinschaft (DFG; grant award numbers: BE 4308/3-1 and BE 4308/3-2) is gratefully acknowledged. Also, we are grateful for constructive feedback from participants at the 2019 International Conference on the Outsourcing of Information Services (ICOIS) in Mannheim, Germany, as well as from research seminar participants at TU Darmstadt.

References

Addady, M. (2016). Uber is starting to monitor drivers for bad behavior. Fortune. Retrieved from http://fortune.com/2016/06/29/uber-monitor-driving-behavior/ (Last Accessed on 20 Sept 2019).

Adler, P. S., & Borys, B. (1996). Two types of bureaucracy: Enabling and coercive. *Administrative Science Quarterly, 41*(1), 61–89.

Alexander, S., & Ruderman, M. (1987). The role of procedural and distributive justice in organizational behavior. *Social Justice Research, 1,* 177–198.

Alge, B. J., Ballinger, G. A., Tangirala, S., & Oakley, J. L. (2006). Information privacy in organizations: Empowering creative and extrarole performance. *Journal of Applied Psychology, 91*(1), 221–232.

Alter, S. (2014). Theory of workarounds. *Communications of the Association for Information Systems, 34*(55), 1041–1066.

Anthes, E. (2017). The shape of work to come. *Nature, 550*(7676), 316–319.

Avgerou, C. (2000). IT and organizational change: An institutionalist perspective. *Information Technology and People, 14*(4), 234–262.

Barney, J., & Felin, T. (2013). What are microfoundations? *Academy of Management Perspectives, 27*(2), 138–155.

Beinstein, A., & Sumers, T. (2016). How Uber Engineering increases safe driving with telematics. Retrieved from https://eng.uber.com/telematics/ (Last Accessed on 31 May 2019).

Benlian, A., Hilkert, D., & Hess, T. (2015). How open is this platform? The meaning and measurement of platform openness from the complementors' perspective. *Journal of Information Technology, 30*(3), 209–228.

Bies, R. J. (1993). Privacy and procedural justice in organizations. *Social Justice Research, 6*(1), 69–86.

Bijlsma-Frankema, K. M., & Costa, A. C. (2010). Consequences and antecedents of managerial and employee legitimacy interpretations of control: A natural open system approach. In S. B. Sitkin, L. B. Cardinal, & K. M. Bijlsma-Frankema (Eds.), *Organizational Control* (pp. 396–433). Cambridge: Cambridge University Press.

Boudreau, K. J. (2012). Let a thousand flowers bloom? An early look at large numbers of software app developers and patterns of innovation. *Organization Science, 23*(5), 1409–1427.

Brenner, B., & Ambos, B. (2013). A question of legitimacy? A dynamic perspective on multinational firm control. *Organization Science, 24*(3), 645–964.

Brockner, J., Ackerman, G., & Fairchild, G. (2001). When do elements of procedural fairness make a difference? A classification of moderating differences. In J. Greenberg & R. Cropanzano (Eds.), *Advances in Organizational Justice* (pp. 179–212). Stanford, CA: Stanford University Press.

Broer, J. (2017). *The Gamification Inventory: An Instrument for the Qualitative Evaluation of Gamification and its Application to Learning Management Systems.* Doctoral Thesis, University of Bremen.

Ceccagnoli, M., Forman, C., Huang, P., & Wu, D. J. (2012). Cocreation of value in a platform ecosystem: The case of enterprise software. *MIS Quarterly, 36*(1), 263–290.

Chalykoff, J., & Kochan, T. A. (1989). Computer-aided monitoring: Its influence on employee job satisfaction and turnover. *Personnel Psychology, 42*(4), 807–834.

Chan, N. K., & Humphreys, L. (2018). Mediatization of social space and the case of Uber drivers. *Media and Communication, 6*(2), 29–38.

Cherry, M. A., & Aloisi, A. (2017). 'Dependent contractors' in the gig economy: A comparative approach. *American University Law Review, 66*(3), 635–689.

Cohen-Charash, Y., & Spector, P. E. (2001). The role of justice in organizations: A meta-analysis. *Organizational Behavior and Human Decision Processes, 86,* 278–321.

Constantiou, I., Marton, A., & Tuunainen, V. K. (2017). Four models of sharing economy platforms. *MIS Quarterly Executive, 16*(4), 231–251.

Cram, W. A., & Wiener, M. (2018). Perceptions of control legitimacy in information systems development. *Information Technology and People, 31*(3), 712–740.

Cram, W. A., & Wiener, M. (2020). Technology-mediated control: Case examples and research directions for the future of organizational control. *Communications of the Association for Information Systems 46* (Article 4), 70–91.
Croitor, E., & Benlian, A. (2019). Perceived input control on online platforms from the application developer perspective: Conceptualisation and scale development. *Journal of Decision Systems, 28*(1), 19–40.
Cropanzano, R., Bowen, D. E., & Gilliland, S. W. (2007). The management of organizational justice. *Academy of Management Perspectives, 21,* 34–48.
Culnan, M. J., & Armstrong, P. K. (1999). Information privacy concerns, procedural fairness, and impersonal trust: An empirical investigation. *Organization Science, 10*(1), 104–115.
Deephouse, D. L., & Suchman, M. (2008). Legitimacy in organizational institutionalism. In R. Greenwood, C. Oliver, R. Suddaby, & K. Sahlin (Eds.), *Handbook of Organizational Institutionalism* (pp. 49–77). London: Sage Publications.
De Groen, W. P., Maselli, I., & Fabo, B. (2016). The digital market for local services: A one-night stand for workers? An example from the on-demand economy. *CEPS Special Report*, No. 133, Centre for European Policy Studies (CEPS), Brussels.
De Stefano, V. (2016). The rise of the 'just-in-time workforce': On-demand work, crowd work and labour protection in the 'gig-economy'. *Comparative Labor Law and Policy Journal, 37*(3), 471–504.
Dowling, J., & Pfeffer, J. (1975). Organizational legitimacy: Social values and organizational behavior. *Pacific Sociological Review, 18*(1), 122–136.
Ferneley, E., Sobreperez, P., & Stevens, J. (2004). Management information or trompe l'oeil? Resistance to workplace surveillance. *Proceedings of the Pacific Asia Conference on Information Systems, Paper, 78,* 1000–1009.
Folger, R., & Konovsky, M. A. (1989). Effects of procedural and distributive justice on reactions to pay raise decisions. *Academy of Management Journal, 32,* 115–130.
French, J. R. P., & Raven, B. (1959). The bases of social power, in D. Cartwright (Ed.), *Studies in Social Power* (pp. 150–167). Ann Arbor: University of Michigan, Institute for Social Research.
Friedman, G. (2014). Workers without employers: Shadow corporations and the rise of the gig economy. *Review of Keynesian Economics, 2*(2), 171–188.
Gallivan, M. J. (2001). Striking a balance between trust and control in a virtual organization: A content analysis of open source software case studies. *Information Systems Journal, 11*(4), 277–304.
Goldbach, T., Benlian, A., & Buxmann, P. (2018). Differential effects of formal and self-control in mobile platform ecosystems: Multi-method findings on third-party developers' continuance intentions and application quality. *Information and Management, 55*(3), 271–284.
Goldstein, J. (2014). To increase productivity, UPS monitors drivers' every move. *NPR.* Retrieved from https://www.npr.org/sections/money/2014/04/17/303770907/to-increase-productivity-ups-monitors-drivers-every-move (Last Accessed on 20 Sept 2019).
Gregory, R. W., Beck, R., & Keil, M. (2013). Control balancing in information systems development offshoring projects. *MIS Quarterly, 37*(4), 1211–1232.
Griffin, A. (2016). Uber to start monitoring drivers through their phones to see whether they drive safely. *The Independent.* Retrieved from https://www.independent.co.uk/life-style/gadgets-and-tech/news/uber-to-start-monitoring-drivers-through-their-phones-to-see-whether-they-drive-safely-a6834551.html (Last Accessed on 20 Sept 2019).
Hall, J., & Krueger, A. B. (2015). An analysis of the labor market for Uber's driver-partners in the United States. Retrieved from https://www.s3.amazonaws.com/uber-static/comms/PDF/Uber_Driver-Partners_Hall_Kreuger_2015.pdf (Last Accessed on 31 May 2019).
Hamari, J. (2013). Transforming homo economicus into homo ludens: A field experiment on gamification in a utilitarian peer-to-peer trading service. *Electronic Commerce Research and Applications, 12*(4), 236–245.

Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does gamification work?—A literature review of empirical studies on gamification, in *Proceedings of the 47th Hawaii International Conference on System Sciences (HICSS).* Big Island: Waikoloa.

Hansen, H. K., & Flyverbom, M. (2015). The politics of transparency and the calibration of knowledge in the digital age. *Organization, 22*(6), 872–889.

Iansiti, M., & Levien, R. (2004). Strategy as ecology. *Harvard Business Review, 82*(3), 68–81.

Jaworski, B. J. (1988). Toward a theory of marketing control: Environmental context, control types, and consequences. *Journal of Marketing, 52*(3), 23–39.

Kessler, S. (2016). How Uber manages drivers without technically managing drivers. Retrieved from https://www.fastcompany.com/3062622/how-ubers-app-manages-drivers-withouttechnically-managing-drivers (Last Accessed on 31 May 2019).

Khosrowshahi, D. (2018). Getting serious about safety. Retrieved from https://www.uber.com/newsroom/getting-serious-safety/(Last Accessed on 31 May 2019).

Kirsch, L. J. (1996). The management of complex tasks in organizations: Controlling the systems development process. *Organization Science, 7*(1), 1–21.

Kirsch, L. J., Ko, D. G., & Haney, M. H. (2010). Investigating the antecedents of team-based clan control: Adding social capital as a predictor. *Organization Science, 21*(2), 469–489.

Kohli, R., & Kettinger, W. (2004). Informating the clan: Controlling physicians' costs and outcomes. *MIS Quarterly, 28*(3), 363–394.

Laumer, S., Maier, C., & Weitzel, T. (2017). Information quality, user satisfaction, and the manifestation of workarounds: A qualitative and quantitative study of enterprise content management system users. *European Journal of Information Systems, 26*(4), 333–360.

Lee, M. K., Kusbit, D., Metsky, E., & Dabbish, L. (2015). Working with machines: The impact of algorithmic and data-driven management on human workers, in *Proceedings of the 33rd ACM Conference on Human Factors in Computing Systems.* Seoul.

Lehdonvirta, V. (2018). Flexibility in the gig economy: Managing time on three online piecework platforms. *New Technology, Work and Employment, 33*(1), 13–29.

Long, C. P., Bendersky, C., & Morrill, C. (2011). Fairness monitoring: Linking managerial controls and fairness judgements in organizations. *Academy of Management Journal, 54*(5), 1045–1068.

Lowry, P. B., & Moody, G. D. (2015). Proposing the control-reactance compliance model (CRCM) to explain opposing motivations to comply with organisational information security policies. *Information Systems Journal, 25*(5), 433–463.

Malhotra, N. K., Kim, S. S., & Agarwal, J. (2004). Internet users' information privacy concerns (IUIPC): The construct, the scale, and a causal model. *Information Systems Research, 15*(4), 336–355.

Marabelli, M., Hansen, S., Newell, S., & Frigerio, C. (2017). The light and dark side of the black box: Sensor-based technology in the automotive industry. *Communications of the Association for Information Systems, 40,* 351–374.

Marx, G. T., & Sherizen, S. (1987). Corporations that spy on their employees. *Business and Society Review, 60*(2), 32–37.

McKinsey Global Institute. (2016). Independent work: Choice, necessity, and the gig economy. Retrieved from https://www.mckinsey.com/featured-insights/employment-and-growth/independent-work-choice-necessity-and-the-gig-economy (Last Accessed on 31 May 2019).

Mehrizi, M. H. R., Modol, J. R., & Nezhad, M. Z. (2019). Intensifying to cease: Unpacking the process of information systems discontinuance. *MIS Quarterly, 43*(1), 141–165.

Meyer, J. W., & Rowan, B. (1977). Institutional organizations: Formal structure as a myth and ceremony. *American Journal of Sociology, 83*(2), 340–363.

Möhlmann, M., & Zalmanson, L. (2017). Hands on the wheel: Navigating algorithmic management and Uber drivers' autonomy, in *Proceedings of the 38th International Conference on Information Systems (ICIS).* Seoul.

Moussa, M. (2015). Monitoring employee behavior through the use of technology and issues of employee privacy in America. *SAGE Open, 5*(2), 1–13.

Niehoff, B. P., & Moorman, R. H. (1993). Justice as a mediator of the relationship between methods of monitoring and organizational citizenship behavior. *Academy of Management Journal, 36*(3), 527–556.

O'Brien, S. A. (2018). Uber tightens driver background checks. Retrieved from https://money.cnn.com/2018/04/12/technology/uber-safety-update/index.html (Last Accessed on 31 May 2019).

O'Brien, S. A., & Yurieff, K. (2017). What we know (and don't know) about Uber background checks. Retrieved from: https://money.cnn.com/2017/11/03/technology/uber-lyft-background-checks-new-york-terror-attack-suspect/index.html (Last Accessed on 31 May 2019).

Ouchi, W. G. (1980). Markets, bureaucracies, and clans. *Administrative Science Quarterly, 25*(1), 129–141.

Pfaffenberger, B. (1992). Technological dramas. *Science, Technology and Human Values, 17*(3), 292–312.

Pollock, N. (2005). When is a work-around? Conflict and negotiation in computer systems development. *Science, Technology and Human Values, 30*(4), 496–514.

Posey, C., Bennett, R. J., Roberts, T. L., & Lowry, P. B. (2011). When computer monitoring backfires. *Journal of Information System Security, 7*(1), 24–47.

Rivard, S. (2014). The ions of theory construction. *MIS Quarterly 38*(2), iii–xiii.

Robert, L. P., Jr. (2019). Crowdsourcing controls: A review and research agenda for crowdsourcing controls used for macro-tasks. In V.-J. Khan, K. Papangelis, I. Lykourentzou, & P. Markopoulos (Eds.), *Macrotask Crowdsourcing: Engaging the Crowds to Address Complex Problems* (pp. 45–126). Springer Nature: Cham.

Rosenblat, A. (2018). *Uberland*. Oakland: University of California Press.

Rosenblat, A., & Stark, L. (2016). Algorithmic labor and information asymmetries: A case study of Uber's drivers. *International Journal of Communication, 10,* 3758–3784.

Scheiber, N. (2017). How Uber uses psychological tricks to push its drivers' buttons. Retrieved from https://www.nytimes.com/interactive/2017/04/02/technology/uber-drivers-psychological-tricks.html (Last Accessed on 31 May 2019).

Schmitz, H., & Lykourentzou, I. (2018). Online sequencing of non-decomposable macrotasks in expert crowdsourcing. *ACM Transactions on Social Computing 1*(1), Article 1.

Schnedler, W., & Vadovic, R. (2011). Legitimacy of control. *Journal of Economics and Management Strategy, 20*(4), 985–1009.

Sitkin, S. B., & George, E. (2005). Managerial trust-building through the use of legitimating formal and informal control mechanisms. *International Sociology, 20*(3), 307–338.

Smith, H. J., Milberg, S. J., & Burke, S. J. (1996). Information privacy: Measuring individuals' concerns about organizational practices. *MIS Quarterly, 20*(2), 167–196.

Smith, W. P., & Tabak, F. (2009). Monitoring employee e-mails: Is there any room for privacy? *Academy of Management Perspectives, 23*(4), 33–48.

Son, J.-Y. (2011). Out of fear or desire? Toward a better understanding of employees' motivation to follow IS security policies. *Information and Management, 48*(7), 296–302.

Stone, E. F., Gardner, D. G., Gueutal, H. G., & McClure, S. (1983). A field experiment comparing information-privacy values, beliefs, and attitudes across several types of organizations. *Journal of Applied Psychology, 68*(3), 459–468.

Suchman, M. C. (1995). Managing legitimacy: Strategic and institutional approaches. *Academy of Management Review, 20*(3), 571–610.

Suddaby, R., Bitektine, A., & Haack, P. (2017). Legitimacy. *Academy of Management Annals, 11*(1), 451–478.

Suddaby, R., & Greenwood, R. (2005). Rhetorical strategies of legitimacy. *Administrative Science Quarterly, 50*(1), 35–67.

Tabak, F., & Smith, W. P. (2005). Privacy and electronic monitoring in the workplace: A model of managerial cognition and relational trust development. *Employee Responsibilities and Rights Journal, 17*(3), 173–189.

Tiwana, A. (2015). Evolutionary competition in platform ecosystems. *Information Systems Research, 26*(2), 266–281.

Tomaselli, F. C., Sanchez, O. P., & Brown, S. A. (2015). How to engage users through gamification: The prevalent effects of playing and mastering over competing, in *Proceedings of the 36th International Conference on Information Systems (ICIS)*. TX: Fort Worth.

Uber. (2019a). Driver requirements: How to drive with Uber. Retrieved from https://www.uber.com/drive/requirements/ (Last Accessed on 31 May 2019).

Uber. (2019b). Pre-first trip: Frequently asked questions. https://www.uber.com/drive/new-orleans/resources/prefirst-trip-faqs/ (Last Accessed on 21 Sept 2019).

Uber.(2019c). Riding with Uber: Driver compliments. https://www.uber.com/ride/how-uber-works/driver-compliments/ (Last Accessed on 31 May 2019).

Whetten, D. A. (2002). Modelling-as-theorizing: A systematic methodology for theory development. In D. Partington (Ed.), *Essential Skills for Management Research* (pp. 45–71). Thousand Oaks, CA: Sage Publications.

Wiener, M., & Cram, W. A. (2017). Technology-enabled control: Effectiveness, socio-emotional consequences, and ethical dilemmas, in *Proceedings of the 23rd Americas Conference on Information Systems*. MA: Boston.

Wiener, M., Mähring, M., Remus, U., & Saunders, C. (2016). Control configuration and control enactment in information systems projects: Review and expanded theoretical framework. *MIS Quarterly, 40*(3), 741–774.

Williams, G. C., & Deci, E. L. (1996). Internalization of biopsychological values by medical students: A test of self-determination theory. *Journal of Personality and Social Psychology, 70*(4), 767–779.

Woltjer, R. (2017). Workarounds and trade-offs in information security—an exploratory study. *Information and Computer Security, 25*(4), 402–420.

Woyke, E. (2018). How UPS Delivers Faster Using $8 Headphones and Code that Decides When Dirty Trucks Get Cleaned. MIT Technology Review. Retrieved from https://www.technology.review.com/s/610183/how-ups-delivers-faster-using-8-headphones-and-code-that-decides-when-dirty-trucks-get/. (Last Accessed on 20 Sept 2019).

Yang, I. (2015). Cross-cultural perceptions of clan control in Korean multinational companies: A conceptual investigation of employees' fairness monitoring based on cultural values. *International Journal of Human Resource Management, 26*(8), 1076–1097.

Replacing Humans by Bots

Knowledge Workers' Reactions to a Planned Introduction of Robotic Process Automation—Empirical Evidence from an Accounting Firm

Aleksandre Asatiani, Esko Penttinen, Joona Ruissalo, and Antti Salovaara

Abstract In this paper, we investigate the initial reactions and perceptions of knowledge workers to a planned implementation of robotic process automation (RPA). Using purposive sampling, we conduct a case study in an industry in which workers' jobs are notoriously vulnerable to automation: we study an accounting firm that is planning to introduce RPA into their core accounting processes. While our informants did raise the expected concerns about job security and loss of control over work, the initial reactions to the technology were surprisingly positive. The informants even expressed enthusiasm and genuine curiosity towards the capabilities of RPA. Overall, our results challenge the views outlined in previous academic literature and popular press concerning the fears and anxieties associated with the introduction of automation technologies in information-intensive knowledge work. To conclude, we theorize on the emerging positively dispersed uncertainty concerning the nature of RPA and the relativistic nature of worker reactions that potentially impact workplace atmosphere.

A. Asatiani
Department of Applied Information Technology, University of Gothenburg, Gothenburg, Sweden
e-mail: aleksandre.asatiani@ait.gu.se

E. Penttinen (✉) · J. Ruissalo
Department of Information and Service Management, Aalto University, Espoo, Finland
e-mail: esko.penttinen@aalto.fi

J. Ruissalo
e-mail: joona.ruissalo@aalto.fi

A. Salovaara
Department of Design, School of Arts, Design and Architecture, Department of Computer Science, School of Science, Aalto University, Espoo, Finland
e-mail: antti.salovaara@aalto.fi

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_17

1 Introduction

Automation of knowledge-intensive work through cognitive automation (CA)[1] and robotic process automation (RPA)[2] has entered the mainstream discussion in industry and academia. In recent years, scholars, industry experts and journalists have put forward predictions concerning the implications of the introduction of such automation tools, ranging from cautiously optimistic accounts (Brynjolfsson and McAfee 2014) to warnings of a dystopian future (Carr 2015; Ford 2015). A widely cited paper released by two Oxford University scholars (Frey and Osborne 2017) claimed that nearly half of the jobs in the US are at risk of being automated. Frey and Osbourne estimate that jobs of accountants and auditors, for example, are particularly susceptible to automation (0.94 probability). Moreover, it has been argued that unlike the manufacturing automation that took place in the late 20th century and was centered on low-skill factory jobs, the new wave of automation is impacting jobs that require advanced cognitive capabilities, thus threatening even high-skill occupations (Akst 2013; Frey and Osborne 2017). For example, tools such as IBM's Watson aim to replace lawyers in various tasks related to pattern recognition and decision making (Fung 2014; Sills 2016). Some authors argue that this development has the potential to unleash an unprecedented rate of human labor automation, turning Keynes' prediction of mass technological unemployment (Keynes 1933) into reality (Autor 2015; Frey and Osborne 2017). Perhaps as a result of these gloomy projections, a recent longitudinal census conducted among European citizens reported a sharp deterioration in attitudes towards robots especially in the area of robots assisting humans at work (Gnambs and Appel 2019).

Notwithstanding the active ongoing discussion around the issue of knowledge-intensive work automation (Salovaara et al. 2019), we note a lack of empirical studies examining how the introduction of tools such as RPA impacts knowledge-intensive organizations and their workers, particularly how knowledge workers respond to the automation of some aspects of their work. It is safe to assume that the active discussion of job losses and technological unemployment in scientific publications (Frey and Osborne 2017), popular business literature (Ford 2015), and media (Cain Miller 2016; The Economist 2016) has had an impact on the reactions of knowledge workers to automation in occupations threatened by it. These reactions present an especially interesting research area in settings where automation plans have recently been announced. Based on the above, in this study, we seek to answer the following

---

[1]Cognitive automation leverages different algorithms and technology approaches to analyze unstructured data such as natural language processing, text analytics, data mining, semantic technology and machine learning (Lacity et al. 2018).

[2]While the word *robot* may bring to mind an image of a physical machine, RPA refers to software that automates service tasks previously performed by humans (Asatiani and Penttinen 2016; Lacity and Willcocks 2016). Software *robots* emulate human execution of tasks (Hallikainen et al. 2018). Instead of interacting with other software through application program interfaces, the software is rules based and interacts with the graphical user interface: typing login credentials to specified fields, moving and clicking a mouse, copying and pasting text from one window to another (Penttinen et al. 2018a).

research question: *How do knowledge workers react to a planned implementation of software robots?*

While earlier studies on the implementation of robots in the manufacturing setting (Argote et al. 1983; Chao and Kozlowski 1986; Herold et al. 1995) inform us on the perceptions, reactions and experiences of manual workers, we argue that this topic warrants a fresh investigation in light of the following points. First, the impact of manufacturing automation is assumed to be limited to unskilled, manual labor (Argote et al. 1983; Chao and Kozlowski 1986). What is novel about the emerging technologies drawing on CA and RPA is that skilled knowledge workers are becoming widely exposed to the rapid threat of automation. Second, contemporary knowledge workers possess more information on robots and work automation than their peers did during earlier industrial revolutions. Hence, the public image of robots is not limited to the *mechanical man* from science fiction (Chao and Kozlowski 1986). Third, knowledge workers have been exposed to the rapid decline in manufacturing jobs in industrialized nations, which is largely attributed to automation (Atkinson 2012). Based on these factors, we argue that knowledge workers' reactions to the introduction of automation tools differs from the reactions of workers within mechanical manufacturing.

Pre-implementation reactions are often overlooked because they are based on workers' preconceptions rather than resulting from first-hand experiences with the technology. However, it has been shown that pre-implementation reactions provide useful insights into post-implementation attitudes and future acceptance of automation (Argote et al. 1983; Blaker et al. 2013; Herold et al. 1995; Vaughan and MacVicar 2004). Therefore, in this study, we seek to establish a foundation for the discussion of the interactions between automation and knowledge workers by studying workers' reactions to the planned introduction of automation tools. We approach the problem by studying an accounting firm at the pre-implementation stage of RPA.

We proceed as follows. After this introduction in Sect. 2, we draw on earlier studies to conceptualize and inform workers' pre-implementation reactions associated with the introduction of automation tools. In Sect. 3, we describe our case company, explain the methodological choices we made, and present our data analysis techniques. In Sect. 4, we present the positive and negative worker reactions with illustrative quotes from the interviews. In the remaining sections, we discuss our findings and provide avenues for further research.

2 Conceptualizing and Informing Pre-implementation Reactions to Automation

Understandably, most scientific enquiry into humans' reactions to automation and robots has been focusing on ex-post implementation attitude, acceptance and assimilation. This focus is probably due to researchers' access and availability of data collection opportunities which are typically plenty in the implementation phase but somewhat scarce in the pre-implementation. Pre-implementation reactions and subsequent attitudes are necessarily limited to workers' preconceptions of the technology in question. Earlier research on both manufacturing and office automation suggests that this stage is often characterized by mix of unfounded optimism (Argote et al. 1983; Chao and Kozlowski 1986; Faunce et al. 1962) and pessimism (Herold et al. 1995; Hoos 1960). Moreover, pre-implementation attitudes are prone to change as workers interact with the technology (Chao and Kozlowski 1986; Herold et al. 1995; Jacobson et al. 1959). Given these characteristics, pre-implementation reactions may appear to provide little value. However, earlier research indicates that pre-implementation reactions might serve as early warning signs (Abdinnour-Helm et al. 2003), impact the success of the implementation, and inform management about accommodations required for reducing friction with workers (Herold et al. 1995).

Pre-implementation reactions are not as straightforward as they may appear at first glance. In practice, workers rarely exhibit either radical Luddite or automation-enthusiast tendencies. Instead, a mix of positive and negative reactions reflect workers' positions within an organization and organizational context. Faunce et al. (1962) studying early office automation suggested that the reactions are not uniform across organizations and vary depending on individual's involvement with automation. The existing literature provides a wide variety of possible positive and negative reactions at the pre-implementation stage. Positive reactions include upgrading jobs, allowing workers to focus on more meaningful tasks (Blaker et al. 2013; Chao and Kozlowski 1986; Herold et al. 1995), enhanced productivity enabled by automation of labor-intensive tasks (Blaker et al. 2013; Gohmann et al. 2005; Herold et al. 1995), opportunities to move to managerial and supervisory roles (Chao and Kozlowski 1986), and reduced errors and streamlining of work tasks (Blaker et al. 2013; Herold et al. 1995).

Negative reactions in prior literature have focused on issues related to the loss of control over work as a result of tasks moving to a *black box* of automation or management scrutinizing work based on data generated by a new system (Argote et al. 1983; Gohmann et al. 2005; Majchrzak and Cotton 1988); job security due to the potential elimination of the need for human labor (Blaker et al. 2013; Chao and Kozlowski 1986; Davis 1962; Herold et al. 1995); social isolation due to a decreased need to interact with other human workers (Argote et al. 1983; Chao and Kozlowski 1986); expanded responsibilities in exchange for automated routine tasks (Argote et al. 1983; Blaker et al. 2013; Chao and Kozlowski 1986; Gohmann et al. 2005; Herold et al. 1995; Majchrzak and Cotton 1988); increased productivity expectations (Argote et al. 1983; Herold et al. 1995); the need to acquire new skills to be competitive within the context of a renewed job description (Chao and Kozlowski 1986; Davis 1962; Herold et al. 1995); and technical difficulties caused by malfunctioning automation (Blaker et al. 2013; Gohmann et al. 2005).

A general consensus from early automation literature is that for unskilled workers, automation presents a threat, while for high-skilled workers, it brings job enhancements (Chao and Kozlowski 1986; Delehanty 1966; Herold et al. 1995; Olson and White 1979). Therefore, in prior literature, the prevalence of either positive or negative reactions has depended upon the skill level of the worker. However, in the era of CA and RPA, we now face a situation where workers impacted by the automation are overall relatively high-skilled. As a result, the observations from manufacturing automation research may not be applicable to this group.

Another aspect that has been found to influence the direction of pre-implementation reactions is the organizational context and capabilities of management to address issues related to business process reengineering. If workers feel that the change process is managed effectively and that change seems inevitable and necessary, they will be more accommodating to automation. On the other hand, if workers perceive that management is incapable of ensuring a successful transition and if there are no apparent benefits for workers, they will be critical towards automation (Herold et al. 1995). In contexts where automation is more clearly aimed at automating particular processes rather than substituting workers altogether, the concerns are somewhat different. One example of such a context is salesforce automation, where workers have been reported to be more concerned with usability issues, management control and the impact on their productivity (Gohmann et al. 2005; Jones et al. 2002) rather than with job security or skill development. Here, less experienced workers were found to be more open to the introduction of the technology, whereas more experienced salespeople were resistant. At the same time, experienced top performers had a more positive reaction to automation than underperformers (Keillor et al. 1997).

3 Method

As our aim was to probe how knowledge workers react to a planned implementation of an automation tool, we chose to conduct a qualitative case study (Yin 2013). Our strategy was not to validate the classes of positive and negative reactions that have been identified in prior literature and outlined in the previous section, as this would have yielded a somewhat mechanistic reporting of empirical evidence on ready-made types of reactions. Additionally, this kind of probing would have guided the workers too much, and we wanted to avoid that. Rather, we approached the case company and data collection openly, letting the informants speak freely and honestly about their perceptions of automation. We next proceed to describing our case selection, data collection, and data analysis in greater detail.

3.1 Case Selection

To select the case company, we employed purposive sampling (Polkinghorne 2005) with two main principles. First, we searched for an information-rich case company (Patton 2001) from an industry in which workers' jobs would be vulnerable to automation. Second, we wished to study a company operating within knowledge-intensive work that would be actively contemplating deploying automation tools, thus providing a natural case setting where the decision-makers and workers would be realistically considering the roles of human experts and automation tools in knowledge-intensive work.

Based on these two principles, the case company chosen for this study was AccComp (pseudonym), a Finnish accounting company specializing in creating and maintaining solutions to data management, analytics and outsourcing. In 2016, AccComp generated a turnover of approximately 130 million Euros and employed slightly fewer than 1000 people. AccComp had three offices in Finland and eight offices in other Nordic countries. In this study, we focus on AccComp's financial process services offering, which is a complete financial services outsourcing solution that includes business process-as-a-service (BPaaS), software-as-a-service (SaaS) and IT support. Internally, AccComp organizes financial process services through shared service centers (SSCs). In SSCs, AccComp has migrated outsourced processes to modern information systems such as Microsoft Dynamics AX and Exflow AX. To further improve the operations, AccComp is in the process of incorporating RPA into its financial services and is considering adding machine learning as part of its palette of automation tools in the future.

3.2 Data Collection

We conducted 13 semi-structured, face-to-face interviews. All interviews were done in Finnish, and they were all audio-recorded and transcribed. The transcribed material resulted in 200 pages of text (single spaced). Immediately after the interviews, we recorded all our observations. We noted the date, location and other relevant circumstances of the interviews. The data collection took place between November 2016 and January 2017.

To select and contact informants within the case company, we used the known sponsor approach (Patton 2001). One of the authors had access to the senior management of the case company. Together with one of the senior managers, we selected informants for the study, ensuring that the participants would represent all the main functions within the financial administration SSCs and have different age profiles, positions in the organization, and educational backgrounds. Additionally, we wanted to interview both specialists (operative-level workers) and managers (executive-level workers) within the case company to obtain a rich overview of the reactions to the planned implementation of RPA. The nine specialist interviews were conducted with experts from order-to-cash (O2C), purchase-to-payment (P2P), record-to-report (R2R), and debt collection functions. Table 1 lists all of the informants and their work-related backgrounds. We have changed the names of the informants to preserve their anonymity.

The interview guides were iterated by the authors to ensure that the questions would be understandable for the interviewees and that the order of different topics would be logical. The questions focused on the knowledge-intensive work and specific actions taken in the work tasks. The questionnaire included questions about

Table 1 Informants

Type	Pseudonym	Age	Experience in financial administration	Interview length (min)
Specialist interviews	Specialist Susan	42	10 years: includes billing, P2P, O2C, R2R, financial statements, and tax returns	82
	Specialist Elizabeth	63	About 30 years: R2R and budgeting	75
	Senior specialist Jane	56	24 years: financial manager, financial controller	75
	Specialist Emily	32	Several years: payroll clerk, transactions handling, billing	68
	Specialist Margaret	56	36 years: R2R, accounts payable, and billing	67
	Specialist Jenny	35	Several years: accounts payable and receivable, debt collection	69
	Specialist Helen	45	Several years: accounts payable and receivable	62
	Specialist Sarah	29	4,5 years: R2R, accounts payable and receivable.	72
	Specialist Christine	51	About 30 years: Accounts payable and receivable, billing, and assisting in accounting	87
Manager interviews	Team lead John	48	Several years: shared service center manager, project manager	70
	Director Lisa	52	28 years: Financial manager, and senior vice president	85

(continued)

Table 1 (continued)

Type	Pseudonym	Age	Experience in financial administration	Interview length (min)
	Vice president Robert	60	27 years: Business controller, group controller, and senior vice president	55
	Manager Mary	50	25 years: Financial manager and software development manager	56

automation in general and how the interviewees perceived it. The interview protocol can be found in Appendix 1. Halfway through the interview, the informants were asked to recall two informative, introductory videos on software robots that they had been asked to view in preparation for the interview.[3] After that, the interviewer asked several questions to be able to decipher the informant's initial reactions to and perceptions of RPA.

3.3 Data Analysis

To analyze our data, we adopted elements of analysis from grounded theory (Bryant and Charmaz 2007; Charmaz 2006). With this decision, we sought to retain a holistic view while investigating a contemporary phenomenon in its real-world context (Yin 2013). Following the grounded theory approach, we pursued inductive theory-building that involved moving from detailed descriptions to more abstract concepts (Bryant and Charmaz 2007). We analyzed notes taken during and after each interview to refine our interview guides. All interviews were transcribed after the data collection stage was finished. We began the data analysis immediately after finishing the data collection.

We used the NVivo qualitative research software in our analysis and coded the data corpus in three iterations. We first started with an open coding to better learn our data. At this stage, we created codes based on informants' discourse to tag our data. As the end product of this coding stage, we created 30 codes with a total of 160 quotes. These 30 codes are described in Appendix 3 with illustrative example quotes. In this first stage of coding, we remained as open as possible and avoided theory-guided coding (Charmaz 2006).

[3]Link to one of the videos: https://www.youtube.com/watch?v=fjdLAqgwMKA.

In the second stage, we used the axial coding approach (Charmaz 2006; Strauss and Corbin 1998) to identify and relate categories within the 30 initial open codes. The purpose of this stage was to create axes to gather conceptually connected codes and, as a result, reduce a large number of codes to a smaller number of logical categories. This stage yielded eight axial codes that we then used in our theory-generating interpretive analysis.

In the last stage, we revisited the axial codes and coded data based on theoretical concepts identified in the literature review. This stage allowed us to create a framework with which to analyze the reactions and perceptions of workers. The three coding stages, the codes, and their description can be found in Appendix 3.

4 Findings

During the interviews, we started our discussion by addressing the reactions to the two RPA videos, asking whether the respondents had heard about RPA prior to the interview. We also asked about their general attitude towards technological change overall. We report these findings in our summary table in Appendix 2. Most of our informants were not familiar with RPA before they were exposed to the two videos. Three out of the nine informants had noticed articles in popular press but did not consider themselves as understanding the technology or its capabilities. The others had not heard the term RPA prior to our contact. Concerning the general attitude towards technology, the informants were quite adaptive, meaning that they understood that their profession (accounting) is one that undergoes significant disruptions and that as workers, they need to adapt and cope with that change. As their client firms employ different accounting information systems (AIS), accounting firms typically cater to various systems, both traditional locally installed AIS and cloud-based AIS. As a result, the respondents had become accustomed to using different kinds of systems and were regularly exposed to technological change.

4.1 Positive Perceptions of RPA

Somewhat contrary to our expectations, our informants' initial perceptions of the videos were rather positive. After seeing the two introductory videos, the informants were quite enthusiastic about RPA and excited to see it in their accounting work processes.

> I think that RPA is an innovative solution, although I have not seen one in action yet. I am eager to see that happen. Really interesting! (Senior accountant Jane)

> I found the videos interesting. I would like to see such software robots doing our work. It could definitely do many things, like in that other video, where the software robot takes the file from e-mail automatically, those types of things, it could easily do […] that would be interesting to see. (Accountant Sarah)

The informants were also curious about the software robots' capabilities to learn new skills:

> What is most interesting to me would be to see how I would teach my work tasks to this kind of software robot. From a sort of research perspective, does the robot understand it or not. And then, I would like to know if they learn to do work that way. (Accountant Jenny)

In more detailed discussions, the initial perceptions focused on RPA's possibilities for upgrading jobs, evening out peaks in the workload, enabling more in-depth analysis of accounting, and reducing errors. We next turn to presenting each of those categories with illustrative quotes from the interviews.

4.1.1 Upgrading Jobs

Most interviewees saw the introduction of RPA as an avenue for them to upgrade their work tasks from manual, routine work to more value-added work

> I am positive towards it [RPA]. It might not replace human work, but human experts start to do different work, and then, it might be that human experts focus more on analyzing numbers and not on manual work. It feels now that we do not have time to do that, that we could analyze the book-keeping more. When we get numbers from the client firm, we could serve the firm better by giving them useful ratios such as a solvency ratio or some other key figures that we could calculate. (Accountant Sarah)

Upgrading jobs was sometimes also associated with a reduced risk of offshoring accounting work outside national borders. The rationale for this was that if the amount of manual work can be reduced by introducing RPA, this would then lead to a decrease in the company's motivation to offshore accounting work to low-cost countries.

> [After having implemented RPA], the residual work requires higher education. Then, probably, these robots can work in Finland; no need to go to India. Maybe some of the accounting work will be back-sourced [from India] to robots in Finland. (Accountant Elizabeth)

4.1.2 Evening Out Peaks in the Workload

From a workload perspective, accounting is notoriously seasonal. The end of the month is usually hectic, as accountants need to ensure that all required receipts and bookings are in the systems for the end-of-month closing. The time around the closing of the financial period (typically the end of the year) is also hectic. Several informants felt that technological advancements such as RPA would have the potential to even out peaks in the workload.

> My initial thoughts [about the videos] were very curious and positive. From resourcing and scalability perspectives, an accountant is a difficult resource. In any given month, only 1.5 weeks consist of efficient work time, and that is at the turn of the month. During that period, an accountant can handle only a limited number of client firms. So how could we add more client firms? Exactly like that, that someone else would do the routine job, either a

> knowledgeable student or automation or something else. So that the accountant could simply look at the end result and verify with his/her expertise that the figures are correct. Then, more client firms could be allocated to that accountant. So, this is very important from a scalability perspective and overall efficiency. (Manager Mary)

4.1.3 Enabling More In-Depth Analysis of Accounting

Most accountants are quite meticulous and like to keep things under control. However, the seasonality and hectic nature of accounting work do not always allow accountants to double check and verify the input data and the outcome of accounting. Many accountants perceived RPA as a tool to offload manual work so that this type of verification and checking could be done.

> [By using this kind of RPA], I could, especially at the turn of the year, focus on those numbers because you always have that feeling of 'What did I do?' when you do them in a hurry and the schedule is tight. So, I would take a deeper dive into the numbers to get them right. It would be wonderful to have time for doing that, to get a feeling that they are correct and I have checked them. (Accountant Christine)

4.1.4 Reduced Errors

Like all information-intensive work processes, accounting work is prone to human errors. The informants were willing to have their work descriptions changed to accommodate the introduction of RPA, and this was partly motivated by the foreseeable decrease in the number of human errors.

> [Through the implementation of RPA], the amount of manual work decreases and, supposedly, the amount of errors should decrease. Because when you insert numbers manually, there is always the possibility of human error. (Accountant Emily)

4.2 Fears and Anxieties Vis à Vis RPA

While the majority of the reactions we coded were positive, there were some negative perceptions as well. These were mainly associated with job security and loss of control over work through fragmentation of work tasks.

4.2.1 Job Security

Quite unsurprisingly, the most cited negative perception of RPA implementation was its negative effects on job security. RPA emulates the actions of humans, and the informants understood that some human work would be replaced by these robots.

> I am not sure if I am right, but these robots will replace a lot of human jobs. (Accountant Elizabeth)

> This is an interesting situation when we start to have more and more people that are pushed aside, and they might have very little expertise, so what do we do? [...] For the first time, we are in a situation where we cannot offer them alternatives. Before, there was always something; we could have them making and archiving paper folders and copying stuff, but that has now been changed. (Vice president Robert)

Job security was often mentioned as a side note in conjunction with positive perceptions, such as that RPA would upgrade jobs and reduce errors. The following pair of quotes illustrate this:

> It would be positive that it would handle the routine stuff, but then again, it would replace my work. (Accountant Helen)

> I am quite positive towards RPA. It speeds up the process [by reducing the number of human errors], but then again, there is the other side: where to allocate the workers when we do not need as many of them as before. (Accountant Susan)

4.2.2 Loss of Control Over Work Through Fragmentation of Tasks

Informants were worried about how they would keep track of their work flow if an RPA took responsibility for certain tasks. They felt that through the implementation of RPA, tasks might be fragmented and accountants might not be able to form a good overall understanding of the process, possibly leading to deskilling of the accountants. They thereby perceived a well-known problem known as one the main *ironies* of automation (Bainbridge 1983):

> Yes, [an accountant] would probably no longer have this kind of overall understanding of what book-keeping is if a robot did some tasks in every part of the process. So, an accountant that has not done the book-keeping process from start to finish would not understand what is happening in book-keeping. If the process is fragmented, then the accountant would not know what leads to what, how income statements are formed from invoices and other documents, and which data the balance sheet consists of. (Accountant Elizabeth)

> I think it would be more difficult to track down the errors made by the software robot; it is easier to track down human errors. And then if everything became so automated [...] if there were errors, I might not be able to track them or even know what transactions were posted and how. There should be some sort of mechanism to keep track of these things. (Accountant Sarah)

4.2.3 Perplexity of What a "Robot" Is

The rules-based nature of the software robots was something that the informants talked a lot about in the interviews. In accounting, there are accounting laws and regulations that, to a great extent, dictate the outcomes of accounting processes. However, the informants were uncertain how the rules that the software robot followed would be written into the systems.

> I do not know how capable the software robot is. Could it draft a tax report and balance sheet? I do not know how it would do that. The balance sheet is a sort of collection of existing data, and if someone just finds them and inserts them into the correct cell, those figures, it [the RPA] can probably code that balance sheet as well. Why not tax reports, as well, so that it picks out the numbers? Because the numbers are ready in the book-keeping, you can define them […] so, why couldn't a software robot do it? […] Everything can probably be in electronic form if the software robot is capable enough. It depends on the programmer [who writes the rules onto the RPA]. (Accountant Elizabeth)

Additionally, the informants were confused about the different types of automation. RPA is a good example of a lightweight IT (Bygstad 2016) that operates on the front-end using graphical interfaces and is relatively easy to implement. The informants found it difficult to draw a line between this type of lightweight automation and the more heavyweight automation requiring modifications to back-end systems.

> Still I must say that it is quite difficult to think when it is a robot and when it is a rule programmed into the system; it is difficult to grasp [the difference]. [I mean] I am unsure if I am thinking of the wrong thing [type of automation] when talking about software robots. (Accountant Susan)

4.3 Summary

The idea of RPA elicited both positive and negative responses among our informants. In different ways, RPA was seen as a force that may change the nature of workers' organizational role and status (e.g., positively through upgrading their jobs or negatively through threatening unemployment). It was also seen more concretely as a mechanism that changes particular work tasks (e.g., positively, as a means to reduce errors, or negatively, as a change agent that fragments one's work). In the following discussion, we will provide interpretive lenses that help analyze these differential responses and consider some of the implications of such effects.

5 Discussion

In this paper, we present a study that probed the initial reactions of knowledge workers to a planned introduction of RPA. Our study contributes to both theory and practice, and we next turn to the theoretical and managerial contributions.

5.1 Theoretical Implications

We theorize our findings of the case study through two lenses. First, we theorize on the emerging positively dispersed uncertainty concerning the nature of RPA, whether symbiotic or augmentative. Second, we claim that the relativistic nature of worker reactions potentially has an impact on workplace atmosphere.

5.1.1 Metaphors for Human-RPA Cooperation

When interpreting the observations made in the empirical study, an important theme pertained to the uncertainty on what RPA actually is and how much artificial intelligence (AI) it entails. Perplexity about RPA's true meaning is an important issue, as our findings (see Sect. 4.2.3 on perplexity) show.

Uncertainty about the future often leads to anxiety. In the case of RPA, uncertainty exists on two levels: on the level of the meaning of a "robot" and on the level of AI that automation can have, independently of whether it is robot-like. In both cases, workers are under-equipped with the knowledge and competence needed to understand the technology they are soon about to start closely interacting with. Educating workers to be more knowledgeable about AI, thus debunking myths around it, may be one solution for decreasing the perplexity; however, we think that a more useful approach may be to develop easily understandable yet accurate enough metaphors for human–automation (or human–RPA) cooperation.

Human–computer interaction (HCI) research offers two classic metaphors for human–computer cooperation, and these can be analyzed as candidates for human–RPA automation as well. Licklider's *symbiosis* metaphor (Licklider 1960) presented humans and computers as co-dependent entities but with different roles. In this metaphor, humans define the goals and make the decisions, while computers carry out routine work that is needed to prepare for insights. This idea of division of tasks based on a "humans are good at, computers are good at" principle pervades much of HCI thinking even today. Engelbart's *augmentation* metaphor (Engelbart 1962) is another prominent classic in HCI research. Instead of starting from a premise of labor division, this metaphor presents computers as a means to improve humans' senses, cognitive capacity and execution of actions. Compared to the symbiosis metaphor, the augmentation metaphor posits more agency for the user.

RPA's vision, according to which work that is currently carried out by humans would be automated, seems more closely tuned in with the symbiosis view. This involves, however, a twist: RPA absorbs an increasingly larger scope of tasks in a human–RPA dyad's total work content, without offering reciprocal benefits for the humans. This shakes Licklider's original idea of a balanced reciprocal symbiotic relationship. In the absence of other strong positive metaphors for human–automation relationships, one approach for increasing the meaning of humans in future work is to combine the two metaphors. The imbalance in the symbiosis could then be counterbalanced with more active augmentation of intellect on the human side (Asatiani et al. 2019).

However, IS and HCI researchers should also seek to develop new metaphors. Mixed-initiative interfaces (Horvitz 1999) where humans and computers act as equal partners offers one such metaphor. This metaphor would also be compatible with a concept of cooperation with an intelligent robot. However, as of now, the metaphor does (yet) not match reality and results in false perceptions. More work is needed for better conceptualizing and communicating to workers how their work may change with an introduction of RPA.

5.1.2 Workplace Atmosphere Implications

Although the participants were receptive to the idea of RPA, the findings also revealed a rich variety of negative perceptions. The topics that emerged reflect very well the open questions that have generally been asked about the increase in automation and for which conclusive settled answers have not been found. Questions such as whether automation leads to unemployment in some profession and whether they lead to a loss of control of one's job, for example, are being heavily debated.

It seems to us that in the valence dimension (i.e., whether the perception of RPA is positive or negative), the differences can be interpreted more deeply through the concept of coping strategies. The theory of coping, developed in psychology (Lazarus and Folkman 1984), posits that humans have a tendency to react to hardships and challenges actively or passively, depending on their sense of locus of control. Persons who find themselves helpless (i.e., who perceive a lack of control) are more likely to adopt passive strategies that seek adaptation to surrounding conditions. High levels of locus of control, correspondingly, are related strategies where people seek possibilities to change their surroundings.

This theory seems to apply to our data in a similar manner as has been reported in other contexts of information systems use (Beaudry and Pinsonneault 2005). The positive perceptions, such as those expressed in relation to the upgrade of jobs (Sect. 4.1.1) or possibilities for more detailed analyses (Sect. 4.1.3), seem to be related to the respective workers' beliefs that they have the power to craft their jobs with RPA's help. The negative perceptions, in turn, such as job loss (Sect. 4.2.1) and loss of control (Sect. 4.2.2), have a very clear connection to a sense of powerlessness. While the active coping strategies were related to reorganizing one's job, upskilling it, or spreading one's work so that it will include deeper analyses, the workers who expressed negative perceptions adopted more adaptation-oriented passive strategies, such as what one could call "damage control" or "bounded acceptance". For example, they described the limits that RPA should have in their work roles. Thus, RPA would be welcome if it took away "routine stuff" (Accountant Helen, Sect. 4.2.1). Beyond that, however, the effect could be seen as negative due to fragmentation of one's work. In these cases, RPA would therefore be resisted.

According to the coping theory, a person's coping strategy is determined at an appraisal stage, where the person weighs the effects of an event on one's personal life. In our case, the appraisal's outcome seems to be affected by *relativistic response*: an evaluation of whether the effects of RPA (and automation in general) are going to hit oneself harder than other people. Thus, some of our participants welcomed backshoring (Accountant Elizabeth; Sect. 4.1.1) because they felt that they would be on the winning side of its effects. Similarly, in the perceptions of possible job losses (Sect. 4.2.1), the defining question seemed to be whether workers considered themselves as those who are going to be “pushed aside” and replaced.

If relativistic response is indeed an important factor in workers’ evaluations of RPA’s and automation’s effects, it may lead to an increased sense of a competitive atmosphere in professions where such peer comparisons have not been commonplace. This relationship could be an interesting topic for future research.

Increased peer competition may foreground tougher values in workplaces. Whether RPA will decrease the peer support, collegiality and generally positive atmosphere in workplaces is a question whose answer remains largely unknown because RPA and automation in general are very recent developments. Earlier research on other forms of automation provides some insights to this question. As discussed in Sect. 2, the workers’ reactions to automation are impacted by the context. Workers who consider themselves to have an advantage (e.g., young workers who are comfortable with modern technology (Chao and Kozlowski 1986) or top performers (Herold et al. 1995; Keillor et al. 1997) tend to be enthusiastic towards automation. On the other hand, unskilled and older workers (Chao and Kozlowski 1986; Herold et al. 1995) tend to perceive automation as a threat. This would suggest that automation could potentially influence the workplace atmosphere negatively and divide workers into opposing camps. However, as of yet, there is no evidence to suggest that RPA may lead to similar effects.

The positive reception and curiosity that our study revealed about workers’ reactions attests that the impact of RPAs and software robots depends on a wide variety of worker characteristics. The negative impacts on workers, although easy to picture, need empirical verification. The paper is one of the first evaluations of this important research path.

5.2 Managerial Implications

Our results offer several potentially valuable recommendations for managers contemplating the introduction of automation tools in knowledge work and specifically accounting. We next discuss these managerial implications on three fronts: (1) automation tools and their fit with the seasonality of accounting, (2) indications of peer competition, and (3) opportunities to harness positively dispersed curiosity concerning RPA.

Accounting is a notoriously *seasonal work domain*, with the end of month and the beginning of the year being typically more hectic than other periods. Several times in the interviews, it was stated that automation tools can provide relief from this seasonal work stress by offering to offload certain manual tasks to automation at peak workload times. Managers can use this kind of argumentation and reasoning when communicating about the introduction of RPA to their workers. Overall, accountants welcome the possibility to focus and double check their work, and automation tools are key in providing these kinds of opportunities to accountants.

Our results suggest that within the domain of accounting, not all tasks and workers are hit by automation in a similar way, resulting in an effect called relativistic response, which we discussed in the previous section. This, in turn, might lead to increased levels of competition among workers that managers need to be aware of (*indications of peer competition*). Within accounting, there exist a myriad of different types of work tasks, ranging from manual, routine tasks of invoice (both sales and purchase) handling to tasks requiring more cognitive capabilities, such as payroll processing and tax management. Managers need to take our findings into account when implementing automation tools in accounting, conduct a careful task-level analysis of the potential impacts of automation on each worker and interpret the associated responses.

Finally, overall, our results highlight workers' curiosity towards automation tools rather than fear and anxiety. Managers should try to harness this *positively dispersed curiosity* of workers to their advantage. They could, for example, develop a careful RPA implementation strategy through which they would clearly articulate the capabilities and deficiencies of the planned automation tools to workers.

5.3 Limitations and Further Research

Like all empirical studies, ours is not without its limitations. First and most importantly, our data corpus consists of a limited number of interviews with informants from one company and one geographical area: Finland. Finland is a very advanced country in terms of accounting software implementation (firms are used to using cloud-based AIS) and penetration of standards associated with structured data (e.g., Finland has the highest penetration of electronic invoices[4]). This maturity in using sophisticated cloud-based AIS coupled with advanced standards might have distorted our findings. Further research could examine whether similar findings can be found through empirical research in less advanced countries. Additionally, our sample consisted of accountants who were not strongly against technological changes in their environment (see Appendix 2 for details). They had become accustomed to technological change by being exposed to several AIS and having gone through several system transitions. Second, while we aimed to provide neutral videos to trigger initial perceptions, the choice of video material might have primed our informants. While the first video was a neutral informative video on the functioning of the RPA tool, the second one portrayed RPA as an assistant to the accountant in a rather positive light. Different video choices might have yielded different initial reactions among our informants.

[4]See, for example, Penttinen et al. (2018b) or the e-invoicing market reports in Koch (2014) or the Eurostat statistics in https://ec.europa.eu/eurostat.

6 Conclusions

In this paper, we set out to investigate the initial reactions and perceptions of knowledge workers to a planned implementation of RPA, responding to calls for research on artificial intelligence in knowledge work (Sutton et al. 2016). We studied an accounting firm that was planning to introduce RPA to its core accounting processes. Based on earlier academic literature and popular press on automation, we expected the reactions to be guided by fear and anxiety. While our informants did raise the expected concerns about job security and loss of control over their work, their initial reactions to the technology were surprisingly positive. The informants even expressed enthusiasm and genuine curiosity towards the capabilities of RPA. Based on our results, we discussed two main theoretical implications: we first theorized on the emerging positively dispersed uncertainty concerning the nature of RPA and its effects on human–computer interaction. Then, we theorized on the relativistic nature of worker reactions potentially having an impact on the workplace atmosphere. Finally, we provided guidance for managers on issues to consider when contemplating the potential introduction of automation tools within the domain of accounting.

Appendix

Appendix 1: Interview Questionnaire

Interview guide for specialist interviews

Respondent's background information

- Age and education?
- Prior work experience?
- Positions held at AccComp?
- Current responsibilities at AccComp?

General questions of daily work

- What systems are you currently using in your work and for what purposes? Do you move information from one system to another manually?
- AccComp implemented a new accounting information system recently. Has your work changed after the implementation of the new system? How has it changed?
- Does the system(s) that you use have the necessary features to be able to carry out your work?
- What is your estimate of the ratio of data that you receive in paper or digital format?

Initial reactions to and acceptance of new technology

- Tutorial videos of RPA:
- Thinking about the tutorial videos about a software robot, have you heard of or used that kind of software robot before?
- What kind of initial feelings do software robots evoke in you?
- Did your understanding of what software robots are and how they work change after you watched the video?
- Do you see the software robot as a positive or negative thing regarding your work, and if so, why?
- Let's look at this list of different financial accounting processes. What processes in this list do you handle at work?
- Can you take me through the [specific financial accounting from the list] process step-by-step and describe it in as much detail as possible? For example, how you enter information into the system, what problems you might encounter, what is repetitive and routine in the process where you can just "switch your brain off," when do you need to focus to get the information right in the system, and so on.
- *Follow up:* Do you follow a clear workflow list written by someone else, or have you yourself formed an informal workflow in your work?
- How often do you react to unpredictable anomalies in your work that require a lot of thinking and attention? Can you handle those situations alone, or do you need help from someone else?
- How many clients do you have at the moment, and do you feel you have enough time to handle your work without feeling overloaded at all times?
- Can you describe how you solved a problematic situation(s) that you encountered in the accounting system(s) or overall?
  - Have you noticed that a small error would lead to a bigger error? Can you give an example?
- Do you report errors forward, and if so, how do you do it?
- What parts of your work tasks do you especially enjoy?
- If you consider that a software robot can take over some of your tasks, what would you do with the time that is left over? Would you, for example, want to take more clients or concentrate more deeply on current clients, aim for a better work position that has a higher salary, or possibly something else?
- Is it a good thing if repetitive mechanical work decreases or is even eliminated from your work? If so, why?
- Are you more willing to embrace a new, more efficient accounting system that you can set to automate some tasks, or a software robot? For what reasons?
- Do you consider software robots to be a progressive, innovative solution to be used in knowledge work?
- If you think on a more general level, how easy or hard do you find it to adapt to technological changes?

- How do you feel about learning new skills, for example, learning how to use a new software to set up software robots to run certain tasks, as in the videos?
- Would you be ready to change your work description and take new tasks to handle, for example, teaching a software robot how to carry out work tasks?
- Would it be harder to keep up your skills that you need in work if a software robot handled some of your tasks in a [specific financial accounting process]?
- Overall, do you feel that you are informed well enough of these software robots and their actual impacts on work? What would you want to know more about?
- Would it be important for you that you can participate in the software robot design process?
    - *If yes*: In what ways would you want to participate, and what kind of impact would you aim to have by participating in the design process?

Interview guide for manager interviews

Respondent's background information

- Education?
- Prior work experience?
- Positions held at AccComp?
- Current responsibilities at AccComp?

General questions

- Does AccComp follow certain management principles, for example, Lean, Six Sigma or TQM?
- What kind of role does technology have at AccComp, and do you follow your field of work's latest technological developments?
- What kind of message does the C-suite aim to give about the role of technology within the organization?
- Does AccComp have an automation strategy?
- Do the business and IT functions work together? If so, how much and in what kind of matters?

Initial reactions to and acceptance of new technology
Tutorial video of RPA

- If you think of the tutorial video about software robotics, have you heard of or seen that kind of software robot in action before?
- Did your image of what software robots are and how they work change after you watched the video?
- What kind of initial feelings software robots evoke in you?
- When you think of the video, do you see the software robot as a positive or negative thing regarding financial management work in the company? If so, why?

- If you consider that a software robot can take over some financial management tasks, how would you allocate the financial management professionals' time that is left over?
- Do financial accounting specialists have the expertise to, for example, teach software robots how to handle work tasks or the willingness to learn new skills?
- Are your workers aware of the possibilities and changes that come with software robots? Have you heard them talking about these software robots or similar topics?
- Do you have a roadmap for implementing RPA yet?
  - *If yes*: Have you done an RPA proof of concept yet?
- Do you have a communications plan to inform about the software robots?
- *If yes:* How long before you decide to start implementing them are you going to begin communicating about software robots?
- Do you already know what you want to achieve with robotic process automation in your unit?
- How easy or hard do you find it to adapt to new technological changes from your own and the organization's point of view?
- Do you consider software robots to be a progressive, innovative solution to be used in knowledge work?
- When you think of recruitment for financial management positions (e.g., accountants, payments receivable clerks), what characteristics and/or skills do you emphasize in recruiting for these positions?
- How much do you anticipate the changing needs in skill sets when recruiting new personnel?
- How have the specialist teams been composed? Does your team have diverse expertise?
- How independent are, for example, accountants and payments receivable clerks in their work?
- Do the specialists follow a clear workflow list written by someone else, or have they formed an informal workflow when they carry out their tasks?
- Do you rotate clients or tasks from time to time between your subordinates, or do they work on the same client and tasks all the time?
- Do you take pre-emptive actions to prevent errors from happening in work?
- Have you noticed that a small error could lead to a bigger error? Can you give an example?
- Do you think that errors could be better anticipated or avoided with training?
- What kind of training or retraining do you provide for workers?

Appendix 2: Informant Profiles

Name	Age; education	Familiarity with RPA	Attitude towards technology	Pre-implementation perceptions of RPA
Specialist Susan	41; business college graduate	Had heard about RPA around a year ago, examples of purchase invoice process development. Not familiar with true implications on work processes	Technological change does not frighten Susan; she has cautious attitude, "let's see [what it can do]."	Positive codes: enhanced productivity Negative codes: job security, technical difficulties, loss of control over work
Specialist Elizabeth	63; business college graduate	Not familiar with RPA prior to interview	Is glad to learn new skills, even coding	Positive codes: new opportunities at work, upgrade of jobs, reduced errors Negative codes: job security, fragmentation of work processes, loss of control over work, potential deskilling Neutral codes: RPA's cognitive capabilities, expansion of responsibilities
Senior specialist Jane	56; BBA	Not familiar with RPA prior to interview	Works primarily in development, so the threshold for taking on new technologies is not so high	Positive codes: reduced errors, upgrade of jobs, new opportunities at work Negative codes: job security, fragmentation of work processes Neutral codes: expansion of responsibilities

(continued)

(continued)

Name	Age; education	Familiarity with RPA	Attitude towards technology	Pre-implementation perceptions of RPA
Specialist Emily	32; BBA	Not familiar with RPA prior to interview	Is quite knowledgeable about systems (other than RPA), follows developments with interest, and finds it easy to adopt new information technology and systems	Positive codes: upgrade of jobs, reduced errors, new opportunities at work Negative codes: job security
Specialist Margaret	56; business college graduate	Not familiar with RPA prior to interview	Does not mind learning to use new systems; on the contrary, she finds it interesting	Positive codes: even out peaks in workload Negative codes: job security, simplification of work tasks (neg)
Specialist Jenny	35; business college graduate	Other than having noticed some articles in the popular press, was not familiar with RPA prior to interview	Positive attitude towards technological change. Is cautious about the long-term impacts [of technology implementation], but adopting new technologies is not a problem	Positive codes: upgrade of jobs, simplification of work tasks (pos)
Specialist Helen	45; BBA	Not familiar with RPA prior to interview	When prompted about her attitude towards technology, Helen stated that "You get used to everything, everything changes."	Positive codes: upgrade of jobs, enabling more in-depth analysis of accounting Negative codes: job security Neutral codes: RPA's cognitive capabilities

(continued)

(continued)

Name	Age; education	Familiarity with RPA	Attitude towards technology	Pre-implementation perceptions of RPA
Specialist Sarah	29; BBA	Has read news about RPA replacing workers, e.g., in claims handling. Not very familiar with the technology; however, understands that they are rules based	Adapts well to technological changes; the older she gets, the more training she feels is necessary	Positive codes: upgrade of jobs, RPA driving BPR, enabling more in-depth analysis of accounting. Negative codes: job security, loss of control over work.
Specialist Christine	51; vocational school graduate	Not familiar with RPA prior to interview	Adapts relatively well to new systems. Has been involved in many system changes during the last 10 years	Positive codes: enhanced productivity, upgrade of jobs, enabling more in-depth analysis of accounting.
Team lead John	48; BBA	Has heard the term RPA but is not familiar with the technology	Some technological changes are easier than others. If there exist good documentation and guidelines, then it is easy. Depends much on the user interface, as most of accounting software has the same functionalities and just the user interface varies	Positive codes: even out peaks in workload, simplification of work tasks (pos), enhanced productivity Negative codes: hazardous work processes.

(continued)

(continued)

Name	Age; education	Familiarity with RPA	Attitude towards technology	Pre-implementation perceptions of RPA
Director Lisa	52; MSc Econ	Has read white papers and attended seminars on RPA. Understands the business case but is not familiar with RPA in practice	Depends on the usability, if it is intuitive and Lisa does not need to spend time searching for functionalities [then it is easy]. If there is a testable prototype of a system, then Lisa is interested; if not, then she might feel reserved	Positive codes: reduced errors, RPA driving BPR, RPA as marketing tool Negative codes: need to reallocate workers to new tasks
Vice president Robert	60; MSc Econ	Somewhat familiar with RPA.	In Robert's position, he feels that he needs to actively adopt new technologies. Overall, he is excited about them but would not want to adopt beta versions of systems; he likes to adopt mature, established systems	Positive codes: upgrade of jobs, easy implementation, heavyweight vs lightweight automation Negative codes: job security
Manager Mary	50; MSc Econ	Understands quite well what RPA is	Takes what is coming at her. Is used to system changes. Finds it natural, in her position, to learn how to use new systems	Positive codes: upgrade of jobs, even out peaks in workload, enhanced productivity, reduced errors

Appendix 3: Coding Scheme

		Coding stages				
Description	Example quote	Open coding	#	Axial coding	#	Thematic coding (reference)
Informant expresses enthusiasm regarding RPA	"Software robots create suppliers in the system, receive invoices, do the postings, interpret the content of the invoice and send them out for approval. So in my mind, there are endless opportunities, and that we are in the forefront thinking about this and taking them into use." (Team lead John)	Enthusiasm	4	Upgrade of jobs	28	Upgrade of jobs (Blaker et al. 2013; Chao and Kozlowski 1986; Herold et al. 1995)
Informant feels that RPA will lead to an expansion of responsibilities	"How can one person work with the robot? [I mean] the human expert would need to master large and wide work entities if the robot replaced much of the manual work." (Specialist Elizabeth)	Expansion of responsibilities	2			Opportunities to move to managerial and supervisory roles (Chao and Kozlowski 1986)
Informant feels that RPA could provide new opportunities at work	"With RPA in place, I could take more customer companies." (Specialist Elizabeth)	New opportunities at work	3			

(continued)

(continued)

		Coding stages				
Informant feels that RPA might lead to simplification of work tasks with positive consequences	"[the implementation of RPA] could generate insights on how to "straighten out" our processes … then, RPA could take care of the routine tasks." (Specialist Jenny)	Simplification of work tasks (pos)	2			
Informant feels that using RPA will lead to upgrade of jobs	"Now it feels that we don't have time to analyze the numbers; [with RPA] we could also analyze book-keeping data, and we could serve our clients better by saying, hey, this is your solvency ratio, I prepared this for you." (Specialist Sarah)	Upgrade of jobs	17			
Informant views RPA as a way to even out peaks in workload	"… we need to get things done by the third day of any given month … so the first week is very hectic … [the work process] would be smoother with RPA." (Specialist Margaret)	Even out peaks in workload	3	Even out peaks in workload	13	Enhanced productivity enabled by automation of labor-intensive tasks (Blaker et al. 2013)

(continued)

(continued)

		Coding stages				
Informant feels that productivity would be increased through RPA implementation	"I think the time required for one client would decrease, so you would need to take something to compensate for that. You cannot just think that, ok, I am going to take a bit longer coffee break. So you would take more clients." (Specialist Susan)	Enhanced productivity	6			
Informant feels that RPA can ignite business process development (BPR) and digitization initiatives	"[With RPA], we would like to improve our pace in developing automation. … we have the as-is situation and then to-be situation, and we would like to get to the to-be situation quicker with RPA." (Director Lisa)	RPA driving BPR and digitization of work processes	4			

(continued)

(continued)

		Coding stages				
Informant perceives RPA as something that will enable him/her to conduct more thorough analysis of accounting data	"I could use the time that is freed up to search for discrepancies in the figures because now we have had to leave the small differences hanging there in the balance sheet, because we have not had time [to correct them], and they may have been there since 2012 when I was not even the book-keeper for this client." (Specialist Sarah)	Enabling more in-depth analysis of accounting	3	Enabling more in-depth analysis of accounting	6	Informating (Zuboff 1988)
Informant discusses RPA as a means to document work processes	"… we talk about work documentation. Often, accountants have some documentation for their own tasks … sometimes we need to shift work between our two offices, and then, we have at least two invoice processing systems in place." (Senior specialist Jane)	Documentation tool	3			

(continued)

(continued)

		Coding stages				
Informant feels that the amount of errors would be reduced with RPA	"Some of my work, such as processing these energy invoices, is very routine-like. You type in the invoice number and first find the contract number, then you type it in, dates, due dates, reference number, amount, VAT code and VAT, then row information, basic fee, energy fee, and then you accept. That's routine. And prone to errors because you do it by hand. Now that's being developed [through RPA]." (Specialist Emily)	Reduced errors	8	Reduced errors	8	Reduced errors (Blaker et al. 2013; Herold et al. 1995)
Informant fears that RPA might have detrimental effects on job security	"I am not sure if I am correct, but these software robots will probably remove a lot of jobs, and the residual work is expert work that requires higher education." (Specialist Elizabeth)	Job security	9	Job security	10	Job security (Blaker et al. 2013; Chao and Kozlowski 1986; Herold et al. 1995)

(continued)

(continued)

		Coding stages				
Informant feels that RPA implementation might require the reallocation of workers to new tasks	"There might be challenges in relocating the persons whose jobs are being automated. Will all of them be able to become, through retraining, experts on RPA?" (Director Lisa)	Need to reallocate workers to new tasks	1			
Informant feels that RPA will lead to fragmentation of work processes	"… no, I don't think it would be difficult [to learn to interact with RPA], but I do think that the nature of the work would change a lot. The work would be fragmented and divided such that in the end, humans would just search for errors in the system. This is a big prejudice." (Specialist Elizabeth)	Fragmentation of work processes	2	Loss of control over work through fragmentation of work	22	Black boxing (Argote et al. 1983; Gohmann et al. 2005; Majchrzak and Cotton 1988)

(continued)

(continued)

		Coding stages					
Informant expresses fear that RPA might lead to hazardous work processes	"In accounting, a person handling purchase invoices and doing book-keeping and handling payments, that person cannot be the same [due to the Finnish accounting legislation]. … this would be a type of dangerous work task combination if robots were handling all of these [tasks]." (Team lead John)	Hazardous work process - lack of validation	1				
Informant fears that RPA might result in a loss of control over work	"I think it would be more difficult to track down if the robot has made a mistake – more difficult to track down those mistakes than the ones that I make." (Specialist Sarah)	Loss of control over work	4				

(continued)

(continued)

		Coding stages				
Informant expresses concerns on potential deskilling resulting from RPA implementation	"With RPA in place, you would not learn how to do accounting in the same way [as without RPA]. You would need to get that learning experience from elsewhere." (Specialist Elizabeth)	Potential deskilling	12			
Informant feels that RPA might lead to simplification of work tasks with negative consequences	"I would not want to just monitor what the RPA does. That's not my… if I am at work, I need to have something [concrete] to do. Otherwise, I get bored." (Specialist Margaret)	Simplification of work tasks (neg)	2			
Informant fears that RPA might cause technical difficulties	"As someone working in accounting, you are of course worried that are they [RPA rules] correct, and what if something goes wrong? When will we catch it and notice it?" (Specialist Susan)	Technical difficulties	1			

(continued)

(continued)

		Coding stages				
Informant discusses RPA's cognitive capabilities	"It was shown that the RPA could learn the task that is given and programmed. I don't know what happens if there is a problem. … Can the RPA go forward and navigate and search somewhere else?" (Specialist Helen)	RPA's cognitive capabilities	2	Perplexity of what a "robot" is	15	
Informant questions whether he/she can trust the capabilities of RPA	"I feel more comfortable trusting the data that I have entered into the system … maybe if I would use robot and see with my own eyes on the screen [what it does], then that would increase my level of trust [in RPA]. (Specialist Susan)	Trust	2			
Informant is unsure about the capabilities of RPA	"If we start from there, where robots have traditionally been implemented in manufacturing, the contrast to these kinds of robots that would "think," it is difficult to grasp." (Manager Mary)	Uncertainty - curiosity of the capabilities of RPA	11			

(continued)

(continued)

		Coding stages				
Informant discusses his/her attitude towards technology in general	"I am positive towards new technology. I want to stay on top of things." (Specialist Jenny)	Attitude towards technology	19	Respondent background	41	N/A
Informant discusses his/her familiarity with RPA prior to the research project	"I am not well aware of [what] RPA [is]. What is the true impact. I have not followed companies that have implemented it or how it has made processes quicker." (Specialist Susan)	Familiarity with RPA	22			N/A
Informant discusses an application area that he/she finds suitable for RPA	"I would implement RPA in the purchase invoice handling process, in the front part of that process. On the video, RPA went into the e-mail, so it should be able to retrieve [purchase invoice] data from there." (Manager Mary)	Application area	4	N/A		N/A

(continued)

(continued)

		Coding stages				
Informant finds RPA easy to implement	"In that robotics example, when there are several software programs, they do not need to be integrated; instead, you operate on top of the software, which has not been possible before." (Vice president Robert)	Easy implementation	1	N/A		N/A
Informant considers his/her preferences over front-end vs. back-end automation	"[In back-end automation], the challenge is integration, which is a pain. [RPA is a delight]. The robot sits on top of existing IT infrastructure and starts move between systems without [requiring heavy] integration. That, in my mind, is the biggest issue changing the landscape." (Vice president Robert)	Front-end vs. back-end automation	9	N/A		Lightweight vs. heavyweight automation (Bygstad 2016)

(continued)

(continued)

		Coding stages				
Informant feels the need to ensure domain knowledge in the development of RPA	"[When developing RPAs], it would be good to have someone involved who really understands accounting and its requirements. If it is developed simply by engineers who have not done accounting, then it might not work as they initially planned." (Specialist Helen)	Need to ensure domain knowledge in development of RPA	1	N/A		N/A
Informant feels that RPA could be used as a marketing tool towards customers	"Our vision includes the digital dimension, and automation is related to this digitalization. It is an important part of our strategy and customer promise. And I see that these software robots are a part of digitalization." (Director Lisa)	RPA as marketing tool	2	N/A		N/A

References

Abdinnour-Helm, S., Lengnick-Hall, M. L., & Lengnick-Hall, C. A. (2003). Pre-implementation attitudes and organizational readiness for implementing an enterprise resource planning system. *European Journal of Operational Research, 146*(2), 258–273.

Akst, D. (2013). Automation anxiety. *The Wilson Quarterly Summer, 3,* 1–13.

Argote, L., Goodman, P. S., & Schkade, D. (1983). The human side of robotics: How workers react to a robot. *Sloan Management Review, 16*(1), 1–33.

Asatiani, A., & Penttinen, E. (2016). Turning robotic process automation into commercial success - Case OpusCapita. *Journal of Information Technology Teaching Cases, 6*(2), 1–8.

Asatiani, A., Penttinen, E., Rinta-Kahila, T., & Salovaara, A. (2019). Implementation of automation as distributed cognition in knowledge work organizations: Six recommendations for managers. In *40th International Conference on Information Systems,* Munich (pp. 1–16).

Atkinson, R. D. (2012). US manufacturing decline and economic development prospects. *Economic Development Journal, 11*(3), 5.

Autor, D. H. (2015). Why are there still so many jobs? The history and future of workplace automation. *Journal of Economic Perspectives, 29*(3), 3–30.

Bainbridge, L. (1983). Ironies of automation. *Automatica, 19*(6), 775–779.

Beaudry, A., & Pinsonneault, A. (2005). Understanding user responses to information technology: A coping model of user adaptation. *MIS Quarterly, 29*(3), 493–524.

Blaker, K., White, L., & Poyser, W. (2013). Dispensary assistants' attitudes and perceptions regarding automated dispensing machines in community pharmacies. *International Journal of Healthcare Technology and Management, 14*(1/2), 90.

Bryant, A., & Charmaz, K. (2007). *The sage handbook of grounded theory*. Sage.

Brynjolfsson, E., & McAfee, A. (2014) *The second machine age: Work, progress, and prosperity in a time of brilliant technologies*. WW Norton & Company.

Bygstad, B. (2016). Generative innovation: A comparison of lightweight and heavyweight IT. *Journal of Information Technology, 32*(2), 180–193.

Cain Miller, C. (2016). The long-term jobs killer is not China. It's automation. [WWW Document]. The New York Times. Retrieved April 17, 2018, from https://www.nytimes.com/2016/12/21/upshot/the-long-term-jobs-killer-is-not-china-its-automation.html.

Carr, N. (2015). *The glass cage: How our computers are changing us,* 1 edn. W. W. Norton & Company.

Chao, G. T., & Kozlowski, S. W. J. (1986). Employee perceptions on the implementation of robotic manufacturing technology. *Journal of Applied Psychology, 71*(1), 70–76.

Charmaz, K. (2006). *Constructing grounded theory: A practical guide through qualitative analysis*. London, United Kingdom: SAGE Publications.

Davis, L. E. (1962). The effects of automation on job design. *Industrial Relations: A Journal of Economy and Society, 2*(1), 53–71.

Delehanty, G. E. (1966). Office automation and the occupation structure. *IMR; Industrial Management Review* (pre-1986) *7*(2), 99.

Engelbart, D. (1962). Augmenting human intellect: A conceptual framework. *Contract, 49*(3578), 80.

Faunce, W. A., Hardin, E., & Jacobson, E. H. (1962). Automation and the employee. *The ANNALS of the American Academy of Political and Social Science, 340*(1), 60–68.

Ford, M. (2015). *Rise of the robots: Technology and the threat of a jobless future*. Basic Books.

Frey, C. B., & Osborne, M. A. (2017). The future of employment: How susceptible are jobs to computerisation? *Technological Forecasting and Social Change, 114,* 254–280.

Fung, H. P. (2014). Criteria, use cases and effects of information technology process automation (ITPA). *Advances in Robotic and Automation, 3*(3), 1–11.

Gnambs, T., & Appel, M. (2019). Are robots becoming unpopular? Changes in attitudes towards autonomous robotic systems in Europe. *Computers in Human Behavior, 93,* 53–61.

Gohmann, S. F., Guan, J., Barker, R. M., & Faulds, D. J. (2005). Perceptions of sales force automation: Differences between sales force and management. *Industrial Marketing Management, 34*(4), 337–343.

Hallikainen, P., Bekkhus, R., & Pan, S. L. (2018). How OpusCapita used internal RPA capabilities to offer services to clients. *MIS Quarterly Executive, 17*(1), 41–52.

Herold, D. M., Farmer, S. M., & Mobley, M. I. (1995). Pre-implementation attitudes toward the introduction of robots in a unionized environment. *Journal of Engineering and Technology Management, 12*(3), 155–173.

Hoos, I. R. (1960). The impact of office automation on workers. *International Labour Review, 82,* 363.

Horvitz, E. (1999). Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems the CHI is the Limit—CHI'99* (pp. 159–166).

Jacobson, E., Trumbo, D., Cheek, G., & Nangle, J. (1959). Employee attitudes toward technological change in a medium sized insurance company. *Journal of Applied Psychology, 43*(6), 349.

Jones, E. L. I., Sundaram, S., Chin, X. Y., & Chin, W. W. (2002). Factors leading to sales force automation use: A longitudinal analysis. *Journal of Personal Selling & Sales Management, 22*(1993), 145–156.

Keillor, B. D., Bashaw, E. R., & Pettijohn, C. E. (1997). Salesforce automation issues prior to implementation: The relationship between attitudes toward technology, experience and productivity. *Journal of Business & Industrial Marketing, 12*(3), 209–219.

Keynes, J. M. (1933). Economic possibilities for our grandchildren. essays in persuasion, 358–373.

Lacity, M., Scheepers, R., & Willcocks, L. P. (2018). Cognitive automation as part of Deakin University's digital strategy. *MIS Quarterly Executive, 17*(2), 4.

Lacity, M., & Willcocks, L. (2016). Robotic process automation at Telefónica O2. *MIS Quarterly Executive, 15*(1), 21–35.

Lazarus, R. S., & Folkman, S. (1984). Stress, appraisal, and coping, stress, appraisal, and coping.

Licklider, J. C. R. (1960). Man-computer symbiosis. *IRE Transactions on Human Factors in Electronics HFE-1* (1), 4–11.

Majchrzak, A., & Cotton, J. C. (1988). A longitudinal study of adjustment to technological change: The case of job transfers from mass to computer-automated batch production. *Journal of Occupational Psychology, 61*(1), 43–66.

Olson, M. H., & White, N. H. (1979). Impact of office automation on society: Implications for education, policy and research.

Patton, M. Q. (2001). *Qualitative evaluation and research methods, qualitative evaluation and research methods* (3rd ed.). London: SAGE Publications.

Penttinen, E., Kasslin, H., & Asatiani, A. (2018a). How to choose between robotic process automation and back-end system automation? In *26th European Conference on Information Systems, ECIS 2018* (pp. 1–14).

Penttinen, E., Halme, M., Lyytinen, K., & Myllynen, N. (2018b). What Influences Choice of Business-to-Business Connectivity Platforms? *International Journal of Electronic Commerce, 22*(4), 479–509.

Polkinghorne, D. E. (2005). Language and meaning: Data collection in qualitative research. *Journal of Counseling Psychology*.

Salovaara, A., Lyytinen, K., & Penttinen, E. (2019). High reliability in digital organizing: Mindlessness, the frame problem, and digital operations. *MIS Quarterly, 43*(2), 555–578.

Sills, A. (2016). ROSS and Watson tackle the law [WWW Document]. IBM Watson Blog. Retrieved April 17, 2018, from https://www.ibm.com/blogs/watson/2016/01/ross-and-watson-tackle-the-law/.

Strauss, A., & Corbin, J. (1998). *Basics of qualitative research techniques*. Thousand Oaks, CA, USA: SAGE Publications.

Sutton, S. G., Holt, M., & Arnold, V. (2016). "The reports of my death are greatly exaggerated"—Artificial intelligence research in accounting. *International Journal of Accounting Information Systems, 22,* 60–73.

The Economist. (2016). Automation and Anxiety [WWW Document]. The Economist. Retrieved April 17, 2018, from https://www.economist.com/news/special-report/21700758-will-smarter-machines-cause-mass-unemployment-automation-and-anxiety.

Vaughan, K., & MacVicar, A. (2004). Employees' pre-implementation attitudes and perceptions to e-learning a banking case study analysis. *Journal of European Industrial Training, 28*(5), 400–413.

Yin, R. K. (2013). *Case study research: Design and methods*. SAGE Publications.

Zuboff, S. (1988). *In the age of the smart machine: The future of work and power*. New York: Basic Books.

Towards an Understanding of Scaling the Software Robot Implementation

Corinna Rutschi and Jens Dibbern

Abstract The implementation of software robots is based on the often time-consuming work carried out by the project team, which often leads to higher than expected costs and time delays. This can be made more efficient by scaling the extension of the robot's functionalities. However, scaling can only take place once one has understood what can be scaled and to what extent. Therefore, based on an empirically illustrated theoretical conceptualization of scaling the software robot implementation, in this chapter we elaborate how scaling can be approached when implementing software robots.

1 Introduction

With the increasing potential to automate business processes using software robots, organizations face the challenge of scaling the implementation of such robotic systems in order to enable their efficient evolution. Generally, a robotic system or a software robot can be any machine replacing work performed by humans (Willcocks and Lacity 2016) while gathering information and following instructions to execute tasks (Tirgul and Naik 2016). Examples for robotic systems are robotic process automation (RPA) (Willcocks and Lacity 2016), chatbots (Sengupta and Lakshman 2017) and self-learning systems (Bostrom 2014). The main advantage of introducing software robots is that they help companies improve quality or efficiency by outperforming people in executing certain tasks and processes (Fung 2014; Guzman and Pathania 2016; Sengupta and Lakshman 2017; Sharma et al. 2016; Slaby 2012). Today, standard robotic implementation solutions are available that provide a kind of toolbox, whereby the robot can be built with the help of the elements contained therein. An example of this is IBM Watson Conversation Services, which make it

C. Rutschi (✉) · J. Dibbern
University of Bern, Bern, Switzerland
e-mail: corinna.rutschi@iwi.unibe.ch

J. Dibbern
e-mail: jens.dibbern@iwi.unibe.ch

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_18

possible to create a chatbot by modeling decision trees. However, the implementation of software robots is based on the often time-consuming work carried out by the project team, which may lead to higher than expected costs and time delays. Ideally, robots may be implemented with foresight of extending their scope and reach, which equals the challenge of scaling the robot implementation process. Such digital scaling may be described as a dynamic reinforcing process by which the reach of a software robot is extended either through expanding its functionalities (mutation) or by transferring its functionalities to additional software robots (inheritance) that may reuse part of its components (Henfridsson and Bygstad 2013). While the introduction of the first robots in organizations often mirrors innovation processes of exploration and experimentation, subsequent robots should be developed more efficiently drawing on scaling mechanisms. Such scaling should allow for a more efficient implementation of software robots since it allows for extending robots or developing additional robots in novel context, without incurring significant additional costs. To gain insights on how scaling can be achieved, we need to understand what can actually be scaled, and how, and to what extent it can be scaled. In the realm of the broader topic of digital transformation, scaling has been analyzed on an infrastructure level (Henfridsson and Bygstad 2013) but not so much with regards to process automation and robot implementation. Thus, the foundational knowledge around scaling of robot implementations is missing. Accordingly, within this chapter we aim at investigating how efficient and effective scaling of the robot implementation process can be achieved. We seek to address this research problem by taking a dynamic perspective and exploring the generative mechanisms that help scaling the robot implementation process. Our overarching objective is to explain what digital scaling means in relation to the implementation of software robots and how scaling can be achieved. Therefore, we formulate the following research question (RQ):

- RQ1: How can the software robot implementation process be scaled?

2 Conceptual Foundation

In this chapter, we have made first steps towards a theoretical understanding and empirical illustration of the scaling of the implementation of software robots. We have chosen a reciprocal approach that has enabled us to conceptualize a model in which we have derived theoretical elements of digital scaling deductively from theory and inductively from empirical data of a case study. However, the model developed in this chapter represents only a first draft and needs to be further refined and substantiated with additional data. Two concepts are foundational for our theorizing effort, i.e. software robots and digital scaling. These two concepts allow us to better understand the evolution of software robots.

2.1 Software Robots

A software robot describes a system that replaces work that was formerly performed by humans (Willcocks and Lacity 2016). Examples for such robots are robotic process automation (RPA) (Willcocks and Lacity 2016), chatbots (Sengupta and Lakshman 2017) and self-learning systems (Bostrom 2014). Thereby, robots gather information and follow instructions to execute tasks (Tirgul and Naik 2016). By introducing software robots, companies can better achieve quality or efficiency goals, while software robots are able to outperform people in executing certain tasks and processes. In contrast to human employees, software robots are permanently available; they can execute certain processes around the clock and at much higher speeds than humans. In addition, they support companies in reducing error rates and thus increasing customer satisfaction (Fung 2014; Guzman and Pathania 2016; Sengupta and Lakshman 2017; Sharma et al. 2016; Slaby 2012). However, as already mentioned, software robots can only execute very specific processes. To date, processes that are particularly suitable to be taken over by a software robot are digitizable, rule-based, uncomplicated, and standardized processes that are executed in large volumes (Asatiani and Penttinen 2016; Fung 2014; Guzman and Pathania 2016; Sengupta and Lakshman 2017; Sutherland 2013; Willcocks and Lacity 2016). In order for a successful evolution of software robots the process of enlarging the reach and functionalities over time has to be understood. Thus, a successful evolution of software robots may be associated with scaling the whole robot implementation process.

2.2 Digital Scaling

In an information systems (IS) perspective, scaling means extending an IS in size and/or scope within the same or a new environment. In relation to software robots, an environment could describe the setting in which a software robot acts involving all surrounding actors. Thus, scaling describes practices that allow a technology to be "spread, enhanced, scoped, and enlarged" (Sahay and Walsham 2006, p. 43). In contrast, the term scale refers to the size and scope of an IS that can be achieved by scaling (Sahay and Walsham 2006). Up to now, scaling has mainly been used to achieve economies of scale through standardization (Chandler 1990). Scaling may thereby lead to different outcomes, such as an increased user base (Huang et al. 2017) or a successfully evolved digital infrastructure (Henfridsson and Bygstad 2013). In order to ensure a successful evolution of software robots, it is essential to understand what scaling means in this context. This requires an understanding of what is scalable, i.e. of what should be scaled (scale) and to what extent this can be done (scaling) (Sahay and Walsham 2006). Robots are designed to perform certain tasks by following certain behavior patterns or rules. In addition, robots include features such as "adaptivity, robustness, versatility and agility" (Pfeifer et al. 2007, p. 1088). Thus, scaling in the sense of robots might be described as engineering robots "capable of performing a large variety of tasks" (Pfeifer et al. 2007, p. 1091). However, scaling does not refer to the extent to which a system can be configured, customized, parameterized (Sahay and Walsham 2006) or adapted. Adaptation may be necessary in the case of environmental changes so that a system can perform processes exactly as it did before the change. However, this does not mean that its functionalities are extended and therefore cannot be called scaling. What can be described as scaling is the step-by-step process in which technology changes into a more complex form (Henfridsson and Bygstad 2013). The flexibility of technology can be innovatively exploited by extending functionality within the same or a new setting. Thus, the addition of new functionalities (mutation) to an IS can describe one mode of scaling. Another mode of scaling describes the transfer of functionalities (inheritance) to a new IS (Huang et al. 2017; Svahn et al. 2017; Yoo et al. 2012). However, scaling does not only describe a technical problem, but rather a socio-technical problem. This can be explained by the fact that the social and technical aspects of an information system are not separate, but interact and influence each other (Henfridsson and Bygstad 2013; Sahay and Walsham 2006; Star and Ruhleder 1996). Generative mechanisms can then be described as "causal structures that generate observable events" (Henfridsson and Bygstad 2013, p. 911). Thus, scaling can be described as a generative process, which requires actions taken by actors such as the developer. Such actions can be associated with reuse. Reuse enables the development and implementation of IT systems in a more efficient way. Reuse of already created elements can be considered as a mechanism that triggers scaling by enabling the addition or transfer of functionality (Banker and Kauffman 1992; Basili et al. 1996). Thus, scaling requires that certain elements can be reused. However, not everything can be reused directly, but certain elements may first have to be modified so that they can then be reused to extend or transfer functionality. Reuse may sometimes be restricted due to impediments that arise and need to be mitigated. The reuse mechanism can therefore not always be applied directly, but depends on context factors that reflect impediments (see Adler et al. 1999). Digital scaling can therefore be described as a generative process that depends on contextual factors and associated impediments, and where various mitigating factors need to be actualized to enable reuse and therefore scaling. It is important to understand what the impediments are that prevent reuse and how mitigating factors can make it possible to overcome them (see Adler et al. 1999). In order to better understand the scaling of the software robot implementation, it is necessary to analyze which components can be reused to what extent, and whether there are impediments and how they can be overcome to enable an addition or transfer of functionality (Henfridsson and Bygstad 2013; Huang et al. 2017; Svahn et al. 2017; Yoo et al. 2012).

3 Conceptualization of Scaling Software Robot Implementation

Implementing software robots by transforming human-executed processes into robots can be done more efficiently as the implementation process scales. We make first steps towards a theoretical understanding and provide an empirical illustration of the scaling of the implementation of software robots. We have chosen a reciprocal approach that has enabled us to conceptualize a model in which we have derived theoretical elements of digital scaling deductively from theory and inductively from empirical data of a case study. However, scaling does not refer to the extent to which a system can be configured, customized, parameterized (Sahay and Walsham 2006) or adapted. Thus, based on the theoretical concepts of digital scaling and our preliminary data, we have developed an initial model of scaling the implementation of software robots (Fig. 1). Digital scaling allows us to theoretically open up the black box of a successful evolution of software robots and how the associated scaling can be approached.

When scaling the implementation of software robots, various aspects must be understood, such as the reusable components, the reuse mechanism, potential impediments and related mitigating factors, and the different scaling modes (Henfridsson and Bygstad 2013; Huang et al. 2017). These aspects in turn have a decisive influence on the successful evolution of software robots or intermediate scaling outcomes. The model describes a scaling phase in which a software robot is further developed from stage n (*scaling stage*) to stage n + 1 (*scaling stage*) or newly developed between both stages. To actually scale, it is necessary to understand how reuse can be tackled and what components can be reused to what extent. When software robots are implemented, processes can be automated whereby certain components are created that can be reused (*scaling trigger*) to extend particular processes (*mutation*) or transfer similar processes to new robots (*inheritance*). However, the potential of drawing on existing robots through reuse (*scaling trigger*) may be hindered due to certain impediments that are grounded in the *scaling context*. This means that depending on the context, certain reusable components may need to be substantially modified before they can actually be reused which can significantly impede the scaling process

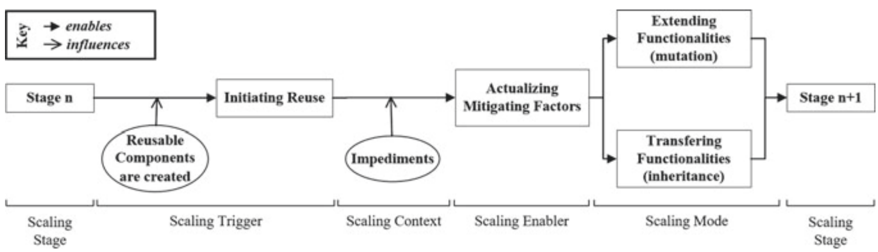

Fig. 1 Phase of scaling

(impediments). If there is such a misfit between the context of the reusable components and the context in which they are to be reused, mitigating factors should help correct this misfit (see Adler et al. 1999). An example of such an impediment could be if a chatbot was built in English and now a German copy of it is to be made. One could assume that the English content can simply be translated to German. However, generally different languages are connected with different cultures and thus different ways of communication. This means that a direct translation is often not possible, because users interact differently with the chatbot in English than other users do in German. The mitigating factor here would therefore be that the content must not only be translated into the required language (i.e. German) but also the associated form of communication. By actualizing mitigating factors, reusable components can be used to extend or transfer functionality to enhance the robot. Digital scaling in the sense of software robots could be assigned to two different scaling modes depending on the reusable components, the context and actualized mitigating factors. The first mode describes the scaling when functionalities of a software robot are extended in a new domain. Thus, the goal here is not to expand the robot so that it can do things better than it has already done, but that it can do new things. This reflects the notion of mutation. The second mode describes scaling when functionalities of one software robot are transferred to another software robot. This reflects the notion of inheritance. Thus, the goal here is to expand to new robots. The extension and transfer of functionalities therefore describe scaling modes, which in turn may lead to the evolution of software robots (Huang et al. 2017; Svahn et al. 2017; Yoo et al. 2012).

4 Illustration Through Chatbot Case

With the help of theoretical sampling, we identified a case that seemed to contribute to empirically illustrate our conceptualized model. The case describes a chatbot project at a Swiss bank and shows how our model can be instantiated. The aim is not to test the model but to illustrate it while some aspects of the model have also been derived from the case data (see e.g. Leonardi 2011). We conducted seven semi-structured interviews with people in different roles within the project team between October and November 2017, and in a second round in September 2018. This helped us to obtain a holistic picture of the case (Miles and Huberman 1994; Yin 2003). In addition, we also analyzed other data such as robot software suit manuals. Given that our key objective is to build theory, the research thrust is exploratory in nature (Benbasat et al. 1987). Qualitative research methods are suitable for “generating novel theory” (Eisenhardt 1989, pp., p. 546)—in particular theory that aims at answering “how” and “why” questions (Yin 2003). This is true for our study as the key objective is to understand how to scale the robot implementation process.

The case describes a bank that wanted to optimize its contact center (CC) in terms of efficiency and in terms of improving performance and reducing costs. Therefore, chatbots were deemed suitable to automate processes. A chatbot represents a virtual assistant (Shawar and Atwell 2007) that imitates human conversations and thus enables the automation of conversational processes (Heller et al. 2005). After releasing a chatbot into the live system, the human user can interact with it via a user interface (UI), such as a pop-up window integrated on a website (Sengupta and Lakshman 2017), Facebook Messenger, Skype or Slack (Patil et al. 2017). Thereby, a chatbot is able to gather knowledge during each interaction with a human user and improve its accuracy of mapping incoming questions to correct answers within a corresponding decision tree (Hildebrand et al. 2003; Sengupta and Lakshman 2017). However, chatbots need training to learn and improve their accuracy (Sengupta and Lakshman 2017). The project regarded was initiated in October 2016 and a German version of a chatbot was released in November 2017. By evaluating the project, different scaling phases could be identified. In the project analyzed, however, it was only possible to digitally scale after the chatbot had been developed and implemented to a certain extent, i.e. the German version had been created. As already mentioned above, scaling does not refer to the extent to which a system can be configured, customized, parameterized (Sahay and Walsham 2006) or adapted. The basic prerequisite for scaling the implementation of the chatbot was therefore the previous development and implementation process of the chatbot in a first phase. The development of the chatbot basically meant to model conversational processes within decision trees and to implement variations and synonyms. Decision trees were modeled around one main question, which constituted the root, while possible direct answers and follow-up questions formed the branches of a decision tree. One main question then required about 100 variations, so that the chatbot was able to answer accurately. "*Still, if there is a 101st question and the syntax is wrong, we are pretty sure the chatbot is going to map the question to the right main question.*" (External Partner). Initially, the chatbot could answer simple questions that contained general information; occurred in high volumes; contained self-service components or aspects that the end user could handle him or herself; and referred to a non-value-adding process. The chatbot was therefore initially able to conduct simple conversations in German, which did not require any system integration. As long as decision trees were extended and new variations and synonyms were added that allowed the chatbot to run processes more accurately in the same domain, i.e. around the same topic, no scaling was performed. Thus, the initial development and implementation phase of the chatbot cannot be referred to as scaling. However, what can be referred to as scaling is when new functionality was added in new domains (i.e. mutation) or when a new chatbot could be created more quickly based on the knowledge gained from the initial development and implementation process (i.e. inheritance). By evaluating the project, different scaling stages could be identified. In summary, three different incidents which each reflect one stage of scaling within the robot development in the analyzed case study could be identified that together represent an evolutionary path that the robot development went through. These are the scaling from the German to the French chatbot, the scaling to the e-banking chatbot, and the scaling to the voicebot. In all scaling stages certain elements could be reused and thus functionalities transferred. Table 1 illustrates the different scaling phases including the corresponding scaling trigger,

Table 1 Scaling phases in Chatbot case

Scaling Phase	Scaling trigger	Scaling context/impediments	Scaling enabler/mitigating factors	Scaling mode	Scaling outcome
French Chatbot	– Modelled conversations in German	– Language misfit	– Language translation	– Transfer functionalities	Evolution of French chatbot
E-Banking Chatbot	– Modelled conversations – Structure of modelled conversations	– Integration misfit – Content specification misfit	– System integration – Deep links	– Transfer functionalities – Extend functionalities	Evolution to e-banking integration
Voicebot	– Modelled conversations – Structure of modelled conversations	– Form of communication misfit – Content specification misfit	– Transformation of form of communication – N.A. yet	– Transfer functionalities – Extend functionalities	Evolution from chat- to voicebot

the scaling context, mitigating factors, the scaling mode, and outcome, which we will explain in more detail below.

French Chatbot. In the course of the implementation of the German chatbot it was determined that another customer request was also a French chatbot. After the official release of the German chatbot in November 2017, the French chatbot was scheduled to go live in October 2018. Thus, when the German chatbot had reached a certain level of maturity, i.e. the chatbot was able to answer questions that met the above criteria with high accuracy, the implementation of the French chatbot was initiated. The already built functionalities, i.e. conversations, of the German chatbot could therefore be reused (*scaling trigger)* and transferred (*scaling mode: transfer functionalities*) to the French chatbot. *"This meant that a new IT instance had to be created."* (IT Project Manager). The decision trees, and implemented variations and synonyms that had already been modelled had to be translated (*scaling context: language misfit*) in order to be reused for the French chatbot. *"This is just a clone for which the content has been translated."* (Business Project Manager 2). Thus, the conversations had to be translated from German into French (*mitigating factor: language translation*). Overall, the implementation of the French chatbot (*scaling outcome*) can be described as the first phase of scaling in which certain elements already created could be reused in a translated form.

E-Banking Chatbot. Alongside the implementation of the French chatbot, an e-banking integration of all chatbot versions, i.e. German and French, was initiated in summer 2018. The integration into the e-banking system was intended to support the customer in handling his or her e-banking activities in the best possible way. *"So that we can offer you help in e-banking. For example, if a customer asks "Where can I find my standing orders?" and the chatbot can then display the page or help the customer get there."* (IT Project Manager). This meant that the respective chatbot should no longer just answer simple questions according to the criteria mentioned above, but should also access customer-specific information and thus conduct more complex conversations. *"In the e-banking system, we know whether the customer is already logged in or not and can then give the appropriate answer without having to give a standard answer."* (Content Manager). In addition to the already modeled conversations, which could be reused (*scaling trigger*) and integrated into e-banking (*scaling mode: transfer functionalities*), new conversations had to be modeled and thus new functionalities had to be implemented in a new domain (*scaling mode: extend functionalities*). *"There are then additional contents that are only developed for e-banking."* (Application Manager). Therefore, it was not possible to directly reuse conversations that had already been modelled. However, what could be reused for the implementation of new conversations was the structure of the existing conversations, i.e. decision trees (*scaling trigger*). In order to enable the integration into the e-banking system, the chatbot had to be connected via suitable interfaces. Thus, a system integration was required (*scaling context: integration misfit*). Therefore, it was necessary to clarify which system dependencies arose in the organization and which interfaces existed (*mitigating factor: system integration*). A sensitive point was that certain customer-specific information could not be processed directly in the chatbot because it would otherwise have been stored in the cloud abroad, which did not comply with Swiss data guidelines (*scaling context: content specification misfit*). To ensure this, a workaround had to be created. *"But it's not about someone saying "what's my balance?" and the chatbot then says "10'000 francs", but he [the chatbot] says "on this page you'll find the balance"."* (IT Project Manager). So-called deep links were used for this, with the help of which the chatbot could refer to the corresponding tiles within the e-banking system (*mitigating factor: deep links*). Thus, the integration of the French and the German chatbots into the e-banking system (*scaling outcome*) can be described as the second phase of scaling in which certain elements already created could be reused and others had to be newly created relying on the structure of reusable components or scaling triggers.

Voicebot. Likewise in summer 2018 the first ideas related to a voicebot were considered. The vision around the voicebot was to enable voice banking. This means that a customer can talk to the bot and the bot can trigger commands in the background. This requires converting speech to text when the customer makes an input and converting text to speech again when the bot makes an output (*scaling context: form of communication misfit*). Conversations that had already been modelled could be converted into the voicebot (*scaling mode: transfer functionalities*), while new functionalities had to be implemented, based on the structure of the existing conversations (*scaling trigger*), that would then enable voice banking activities (*scaling mode: extend functionalities*). Originally, the project team assumed that the already modelled conversations of the chatbot could be reused (*scaling trigger*) for this purpose. This could be done in part, but the conversations must be transformed into a form in which people speak (*mitigating factor: transformation of form of communication*). *"But you don't speak the same way you write."* (Business Project Manager 2). Additionally, in contrast to the integration in e-banking, the problem of not having sensitive data in the chat protocol, i.e. on the cloud, cannot be bypassed so easily (*scaling context: content specification misfit*). *"It's different with voice, he [the voicebot] doesn't tell you which page to visit, he [the voicebot] has to give you a number there."* (IT Project Manager). However, this problem has not yet been solved and no mitigating factor has yet been identified. The transformation of the text-based bot into a voice-based bot was therefore associated with various challenges that the project team is currently working on. Thus, the transformation from a chatbot to a voicebot (*scaling outcome*) can be described as the third phase of scaling in which certain elements already created could be reused but had to be transformed into a form in which people speak.

In summary, three different scaling phases could be identified, the scaling to the French chatbot, the scaling to the e-banking chatbot, and the scaling to the voicebot. In all scaling phases certain elements could be reused and thus functionalities transferred. In phases two (e-banking chatbot) and three (voicebot) not only functionalities were reused and transferred, but also new functionalities were added, whereby the already defined structure of existing functionalities, i.e. conversations, could be reused. While mutation therefore occurred in all identified scaling stages, inheritance occurred in the e-banking and voicebot stages. Mutation and inheritance describe scaling modes that are based on the reuse of certain already created components. However, the more such reusable components (e.g. decision trees) are embedded into a certain context, the more likely it may be that there is an impediment preventing reuse. In order to overcome these impediments, mitigating factors must be actualized (e.g. translation from one language context into another language context). Thus, depending on the scaling context (language misfit, integration misfit, content specification misfit, form of communication misfit), the actualization of different mitigating factors helped to overcome impediments and accordingly to scale in different modes (transfer and extend functionalities), which resulted in different scaling outcomes (evolution of French chatbot, evolution to e-banking integration, evolution from chat- to voicebot).

5 Discussion

Software robots are expected to dramatically improve the efficiency of companies and disrupt the way humans and machines work, and collaborate (Schwab 2017; Willcocks and Lacity 2016). It is critical to understand how companies can successfully transform processes into robots and how to scale such robot implementations. We contribute to digital scaling literature by examining how the implementation of software robots can be scaled. For this purpose, we conceptualized existing scaling constructs from the literature, while some aspects of the model were also derived from the data. We aim at showing exemplarily how scaling was approached within the case study, i.e. the chatbot case. It could be shown that digital scaling can be divided into different phases. Within these phases, different mitigating factors can be actualized in order to overcome certain impediments, which results in different scaling modes and thus creates different scaling outcomes. The implementation of software robots is associated with high costs and time expenditure. These can be reduced by scaling and therefore the implementation of software robots can be made more efficient. Thereby, the implementation of software robots could also be described as outsourcing to machines as opposed to outsourcing to near or offshore centers. Outsourcing is one form of sourcing and defines an agreement with an external service provider to manage and complete a specific work for a specific time period, cost level and level of service (Dolgui and Proth 2013; Oshri et al. 2015). In general, the main advantage of outsourcing is that processes can be transferred from local employees with higher wages to employees near or offshore with lower wages. This cost advantage can be further extended if processes are transferred to robots which do not receive wages instead of to employees with lower wages. If the outsourcing of processes to robots can be scaled, then not only lower costs, but also greater efficiency can be achieved. However, in order to be able to scale at all, it must be understood what and to what extent this can be scaled. Thus, an understanding of how to scale the software robot implementation process is of great interest to both research and practice.

6 Outlook

We have developed a first descriptive model of the process of scaling as the basis for our deeper analysis of scaling and the evolutionary process of implementing software robots. Our model describes a first steps towards a theoretical understanding and empirical illustration of the scaling of the implementation of software robots. Further investigations are necessary to test and further develop the model. The analogy between traditional outsourcing and the implementation of software robots can also help to better understand the dynamics of digital scaling. In addition, we are currently only concerned with the extent to which digital scaling is approached by humans. According to the current state of the technology, software robots seem not yet to be able to scale their functionalities themselves or only to a very limited extent. Thus, it is difficult for software robots to deal with possible impediments or changes in the environment. In contrast, however, self-learning systems may be able to extend their functionalities over time and thus scale them. Forthcoming it would be interesting to better understand which scaling possibilities there are with regard to the implementation of robots or self-learning systems and to what extent these can be best realized and used. In addition, when considering reuse as a scaling mechanism, certain impediments may not always allow reuse to be realized directly. In order to reuse components, they must first be created (Banker and Kauffman 1992; Basili et al. 1996). In the chatbot example, this was done by initially defining and modeling decision trees, and variations and synonyms. However, once such reusable components are embedded in the robot and thus in a particular environment, they may need to be modified before they can be reused. For this reason, the embedding of reusable components in robots can cause certain impediments, whereupon mitigating factors must be actualized in order to overcome these impediments and thus enable reuse. In fact, the more components are embedded, the more likely it may even be that there is an impediment preventing reuse. The processes that software robots perform could also be described as routines, while routines describe a series of interdependent actions performed on a pattern basis (Feldman et al. 2016). Thus, if robots take over certain processes, this could be described in such a way that routines previously executed by humans must be converted into a form so that they can be executed by the robot. Routines are thus embedded in the robot. By embedding routines into robots, routines become stable (D'Adderio 2011). There seem to be some analogies between routine embedding and occurring impediments. Thus, a better understanding of the relationship between routine embedding and related impediments might be of high value.

References

Adler, P. S., Goldoftas, B., & Levine, D. I. (1999). Flexibility versus efficiency? a case study of model changeovers in the Toyota production system. *Organization Science, 10*(1), 43–68.

Asatiani, A., & Penttinen, E. (2016). Turning robotic process automation into commercial success—Case Opuscapita. *Journal of Information Technology Teaching Cases, 6*(2), 67–74.

Banker, R. D., & Kauffman, R. J. (1992). Reuse and productivity in integrated computer-aided software engineering: An empirical study. *MIS Quarterly, 14*(3), 374–401.

Basili, V. R., Briand, L. C., & Melo, W. L. (1996). How reuse influences productivity in object-oriented systems. *Communications of the ACM, 39*(10), 104–116.

Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *Management Information Systems Quarterly, 11*(3), 369–386.

Bostrom, N. 2014. *Superintelligence: Paths, Dangers, Strategies*. OUP Oxford.

Chandler, A. D. (1990). *Strategy and structure: Chapters in the history of the industrial enterprise*. MIT press.

D'Adderio, L. (2011). Artifacts at the centre of routines: Performing the material turn in routines theory. *Journal of Institutional Economics, 7*(2), 197–230.

Dolgui, A., & Proth, J.-M. (2013). Outsourcing: Definitions and analysis. *International Journal of Production Research, 51*(23–24), 6769–6777.

Eisenhardt, K. (1989). Building theories from case study research. *Academy of Management Review, 14*(4), 532–550.

Feldman, M. S., Pentland, B. T., D'Adderio, L., & Lazaric, N. (2016). Beyond routines as things: Introduction to the special issue on routine dynamics. *Organization Science, 27*(3), 505–513.

Fung, H. P. (2014). Criteria, use cases and effects of information technology process automation (Itpa)," *Advances in Robotics & Automation* (3).

Guzman, I., & Pathania, A. (2016). Chatbots in customer service. Retrieved October 25, 2017, from https://www.accenture.com/t00010101T000000__w__/br-pt/_acnmedia/PDF45/Accenture-Chatbots-Customer-Service.pdf.

Heller, B., Proctor, M., Mah, D., Jewell, L., & Cheung, B. (2005). "Freudbot: An investigation of Chatbot technology in distance education. In *EdMedia: World Conference on Educational Media and Technology*: Association *for the Advancement of Computing in Education* (AACE), pp. 3913–3918.

Henfridsson, O., & Bygstad, B. (2013). The generative mechanisms of digital infrastructure evolution. *MIS Quarterly* 907–931.

Hildebrand, M., Eliëns, A., Huang, Z., & Visser, C. (2003). Interactive agents learning their environment. In *International Workshop on Intelligent Virtual Agents* (pp. 13–17). Berlin: Springer

Huang, J., Henfridsson, O., Liu, M. J., & Newell, S. (2017). Growing on steroids: Rapidly scaling the user base of digital ventures through digital innovation. *MIS Quarterly, 41*(1).

Leonardi, P. M. (2011). When flexible routines meet flexible technologies: Affordance, constraint, and the imbrication of human and material agencies. *MIS Quarterly, 35*(1), 147–167.

Miles, M., & Huberman, A. (1994). *Qualitative data analysis: An expanded sourcebook*. London: Sage Publications.

Oshri, I., Kotlarsky, J., & Willcocks, L. P. (2015). *The handbook of global outsourcing and offshoring,* 3rd Edn. Springer.

Patil, A., Marimuthu, K., & Niranchana, R. (2017). Comparative study of cloud platforms to develop a Chatbot. *International Journal of Engineering & Technology, 6*(3), 57–61.

Pfeifer, R., Lungarella, M., & Iida, F. (2007). Self-organization, embodiment, and biologically inspired robotics. *Science, 318*(5853), 1088–1093.

Sahay, S., & Walsham, G. (2006). Scaling of health information systems in India: Challenges and approaches. *Information Technology for development, 12*(3), 185–200.

Schwab, K. (2017). *The fourth industrial revolution*. Crown Business.

Sengupta, R., & Lakshman, S. (2017). Conversational Chatbots—Let's Chat. Retrieved October 25, 2017, from https://www2.deloitte.com/content/dam/Deloitte/in/Documents/strategy/instrategy-innovation-conversational-chatbots-lets-chat-final-report-noexp.pdf.

Sharma, P., Southern, R., & Dalton, D. (2016). The discruptive Chat Bots—Sizing up real opportunities for business. Retrieved 25 Oct, 2017, from https://www2.deloitte.com/content/dam/Deloitte/ie/Documents/ie-dispruptivechat-bots.pdf.

Shawar, B. A., & Atwell, E. (2007). Different measurements metrics to evaluate a Chatbot Syste. In *Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies* (pp. 89–96). Rochester, NY, USA: Association for Computational Linguistics.

Slaby, J. R. (2012). Robotic automation emerges as a threat to traditional low-cost outsourcing, *HfS Research Ltd*.

Star, S. L., & Ruhleder, K. (1996). Steps toward an ecology of infrastructure: Design and access for large information spaces. *Information Systems Research, 7*(1), 111–134.

Sutherland, C. (2013). Framing a constitution for robotistan. *Hfs Research*.

Svahn, F., Mathiassen, L., & Lindgren, R. (2017). Embracing digital innovation in incumbent firms: How Volvo cars managed competing concerns. *MIS Quarterly* (41:1).

Tirgul, C. S., & Naik, M. R. (2016). Artificial intelligence and robotics. *International Journal of Advanced Research in Computer Engineering & Technology, 5*(6), 1787–1793.

Willcocks, L., & Lacity, M. C. (2016). *Service automation: Robots and the future of work*. Steve Brookes Publishing.

Yin, R. (2003). *Case study research*. California: Thousand Oaks.

Yoo, Y., Boland, R. J., Jr., Lyytinen, K., & Majchrzak, A. (2012). Organizing for innovation in the digitized world. *Organization Science, 23*(5), 1398–1408.

Final Synthesis

Synthesis: Outsourcing of Information Services: Where Are We?

Rajiv Sabherwal

Abstract In this paper, I share some insights obtained from the papers from the fifth International Conference on Outsourcing of Information Services (ICOIS), while drawing upon the broader literature on outsourcing. I examine the current status, changes over time, and potential future directions in the area of outsourcing. As discussed in the paper, the outsourcing of information services (OIS) seems to offer greater potential benefits, with new industries being rooted on outsourcing and significant societal impacts. However, risks associated with OIS have increased as well, making it imperative that firms consider both aspects and not outsource IT when it increases risk and lowers returns. OIS has also become more diverse and more complex, with the broadening of actors involved from one client and one or a few vendors to a large number of individual and organizational external agents, potentially including some central and some peripheral actors as well as crowds, small cloud-based firms, robots, and marginalized individuals. As expected, the governance of outsourcing arrangements has undergone major shifts, from contracted and collaborative arrangements to arms-length and digitally-mediated structures. The research on outsourcing parallels this trend, with greater diversity in both theoretical foundations and research methods, and seems to permeate research in other areas such as business value of IT and design science. Perhaps even more drastic changes lie ahead in OIS, amidst emerging ITs, such as "big data," blockchains, social media, cloud computing, and artificial intelligence.

1 Introduction

The first academic Conference on Outsourcing of Information Systems Services was held in May 1993. Subsequently, the name of this conference was changed to International Conference on Outsourcing of Information Services (ICOIS) when the second conference was held in June 2001. This led to an edited book "Information Systems Outsourcing in the New Economy: Enduring Themes, Emergent Patterns

R. Sabherwal (✉)
University of Arkansas, Fayetteville, AR, USA
e-mail: RSabherwal@walton.uark.edu

R. Hirschheim et al. (eds.), *Information Systems Outsourcing*, Progress in IS,
https://doi.org/10.1007/978-3-030-45819-5_19

and Future Directions" (2002). Subsequently, ICOIS was held in 2007, 2013, and most recently in 2019. In this paper, I share some insights I have obtained from attending this fifth ICOIS at Mannheim and reading the associated papers, while drawing upon the broader literature in the area including my own research over time.

Following Kodak's landmark information technology (IT) outsourcing decisions in 1989, the global IT outsourcing market was estimated to have grown to $76 billion by 1995 (Lacity and Willcocks 1998) and to $270 billion by 2010 (Willcocks et al. 2011). The industry focus during that period was on the outsourcing of IT and processes related to it, such as development of information systems (IS). Not surprisingly, research on the outsourcing of information services also started with a somewhat simple, perhaps narrow view of outsourcing, focusing on the outsourcing of information systems, with interest in the associated governance processes, risks, and consequences.

According to Wikipedia (https://en.wikipedia.org/wiki/Outsourcing, accessed December 14, 2019), the term "outsourcing" is based on the combination of "outside" and "resourcing." Accordingly, all types of outsourcing have one aspect in common: they involve an organization using an external party to handle certain business activities. More specifically, outsourcing of information services (OIS) is the use of certain external resources to handle certain aspects of information services, including either ongoing service or a one-off project (Kotlarski et al. 2018). This definition of OIS raises four broad questions:

(1) what aspects of information services are being outsourced (i.e., the "what" question)?
(2) to which external resources is the focal organization outsourcing (i.e., the "who" question)?
(3) what are the distinguishing features of this practice (i.e., the "how" question)? and
(4) what are the implications of such outsourcing (the "so what" question)?

Moreover, two additional questions arise regarding research on ISO: (5) which literature areas are being used in such research; and (6) which empirical methods are being used. Recent research, including papers presented at the Fifth ICIOS (2019), indicates that ISO research has become significantly diverse in terms of each of the above six aspects. In the next six sections, I examine these changes in each of these aspects, while drawing upon the papers from the 2019 ICOIS conference.

2 What Is Being Outsourced?

Beginning with the outsourcing of the development of IS, OIS moved to the outsourcing of the maintenance and support of a firm's IT infrastructure. Subsequently, some firms moved on to outsource not just their information systems or technologies but also their governance of the entire IS function (e.g., Sabherwal et al. 2001). This diversification and growth in outsourcing is not limited to information services;

indeed, outsourcing of business activities across various areas, including manufacturing, led to concerns in 1980's regarding a "hollow corporation" (Business Week 1986; Pastin and Harrison 1987). For example, Pastin and Harrison (1987, p. 54) remarked:

> … when you peek inside many corporations, especially manufacturing corporations, you find that no one is home. The factories are empty and the offices sparsely populated with fearful managers.

The diversification and growth in the scope of outsourcing has only accelerated in recent years, with the emergence of "sharing economy" and concerns about "hyper-outsourcing" (Srnieck 2016). The 2019 ICOIS highlights this increasing diversity of aspects being outsourced; the focus of outsourcing in the 22 papers varies considerably, including the outsourcing of IT governance (Ravindran 2019), individuals' tasks (Asatiani et al. 2019; Rutschi and Dibbern 2019), data (Jarvenpaa and Markus 2019), cybersecurity (Benaroch 2019), micro services (Bozan et al. 2019), the generation of design ideas (He et al. 2019; Hurni et al. 2019), and key business activities (Wiener et al. 2019). Moreover, there is a significant shift in the scope of OIS, with new businesses being entirely built on using IT to outsource business activities (Wiener et al. 2019; Asatiani et al. 2019; Rutschi and Dibbern 2019). Wiener et al. (2019) discuss how individuals are no longer employed in traditional, full-time jobs, and instead often work, either full-time or part-time, as independent contractors who are paid for performing an assigned task, or 'gig.' Indeed, information sharing is critical to new business models such as those based on the outsourcing of labor and vehicles (as in Uber; Wiener et al. 2019) and the outsourcing of household assets in Airbnb (Srnicek 2016; Mead 2019). These trends are not altogether surprising in the current era of digital transformation (Dibbern and Hirschheim 2020).

3 To Whom Is It Being Outsourced?

Initial OIS efforts involved outsourcing activities to vendor organizations who could perform those activities at lower cost or with greater innovation. This was followed by the outsourcing of activities to multiple organizations with carefully nurtured (through relationship-building and contractual mechanisms) interorganizational relationships (e.g., Choudhury and Sabherwal 2003; Dibbern et al. 2015) and to individuals who were typically hired as "contractors" and charged with either developing software or, in some cases (e.g., Sabherwal 1999), overseeing the development of systems by a vendor firm. In some subsequent cases, activities were outsourced to marginalized individuals or groups (i.e., "impact sourcing"; Lacity and Rottman 2012; Malik and Nicholson 2019), including in villages (i.e., "rural sourcing", Lacity and Rottman 2012) and prisons (i.e., "prison sourcing"; Lacityet al. 2015).

This trend toward engaging a variety of entities—including individuals and organizations—has continued in the recent times. Activities continue to be outsourced to other firms (Hurni et al. 2019; Maier et al. 2019) and individuals (He et al. 2019;

Malik and Nicholson 2019; Wiener et al. 2019), but to newer kinds of both. For example, Maier et al. (2019) discuss the outsourcing to cloud-based software Small and mid-size enterprises (SMEs), whereas Malik and Nicholson (2019) discuss "impact sourcing" to marginalized individuals—women in economically-backward area in Pakistan. Even earlier, organizations would sometimes outsource to multiple other organizations or to multiple individuals, but the progress in cloud computing has made it much easier to outsource activities to multiple distributed organizational actors (Hurni et al. 2019), or independent contractors (Wiener et al. 2019; He et al. 2019). He et al. (2019) use the label of "crowdsourcing" to characterize this shift, Hurni et al. (2019) view it as distributed innovation, whereas Bozan et al. (2019) view it as a shift from using thoroughbreds to using a school of goldfish. In addition to this shift toward a significantly larger number of external individual or organizational agents, an entirely new kind of target entity to which tasks can be outsourced has gained prominence; fueled by the recent development in artificial intelligence and machine learning, tasks are being outsourced to robots as discussed in two of the papers in the 2019 ICIOS (Asatiani et al. 2019; Rutschi and Dibbern 2019). Although the automation of information-related tasks has long been possible through IT (Zuboff 1988), the emergent robots and chat-bots behave in a much more human fashion, thereby opening up new possibilities for outsourcing to these emergent tools (developed within the organization or by external vendors).

4 How Is It Being Outsourced?

Given the changes in what is being outsourced and to whom, it is no surprise that the practices associated with how the outsourcing is done have been changing as well. Considerable work on OIS over time has focused on aspects such as the decision regarding whether or not to outsource (Lacity and Hirschheim 1993), selecting the vendor and establishing the contractual and relational arrangement, and subsequently managing the relationship (Sabherwal 1999; Choudhury and Sabherwal 2003; Lacity and Hirschheim 1993), with the selected vendors. Along similar vein, in a recent curation reviewing sourcing studies published in *MIS Quarterly* since 1990s, Kotlarski et al. (2018) identify three broad clusters in IS sourcing literature: (1) making the sourcing decision; (2) designing contractual structures; and (3) managing the sourcing relationship.

Early OIS work focused on enabling collaboration with the vendor while recognizing and addressing potential opportunism, this was usually in close relationships with one or few vendors. As the number of external agents—organizational or individual—involved in outsourcing has grown, and we even encounter situations where the sources may be unknown (Jarvenpaa and Markus 2019), the practices associated with managing them have inevitably changed too. New ways to control projects (Huber et al. 2019) and mitigate risk (Gozman et al. 2019) have been discussed. Moreover, we have seen a shift from trust building with one vendor (Sabherwal 1999) to triads (client, client's rival, and a common vendor) (Kishore et al. 2019), bridging (Su 2019), such as through awareness building, (Malik and Nicholson 2019) the relationships with a large number of external agents.

The changes mentioned in the last paragraph focus on the client perspective. By contrast, Meiser and Beimborn (2019) focus on the vendor. They identify 22 categories of innovations specifically by vendors in OIS arrangements, and divide them into innovation outcomes and innovation enablers. They then classify the innovation enablers into four dimensions: collaboration (wherein the vendor works with others such as associations, startups, and academia), processes (i.e., vendor's internal procedures such as employee coaching, talent management, and design thinking), structures (including establishing new units, launching incubator programs or innovation labs, and acquiring innovative firms), and events (which are one-time or periodic, such as conferences and hackathons).

In addition to such changes from the client and vendor perspectives of OIS governance, the overall view of the OIS arrangement is changing as well. We have seen shifts from from partnerships in IS development to value co-creation (He et al. 2019) through innovative ecosystems (Hurni et al. 2019) that involve simultaneous cooperation and competition (i.e., "coopetition," Hurni et al. 2019), and from internal integration (within large software vendors) to external integration (with multiple vendors) (Maier et al. 2019).

Departing from the 2001 focus on TCE, organizations, and strategy, the authors in the 2019 ICOIS pay considerable attention to the individuals involved in outsourcing. For example, Su (2019) examines cultural sense-making by such individuals. Two examples of the above innovation enablers for vendors (Meiser and Beimborn 2019) focus on the coaching of employees and the management of talent. We have also seen other innovative individual-level practices, such as allowing women workers in Pakistan to bring their mothers to trips (Malik and Nicholson 2019), and recognizing the importance of new aspects such as welfare (Malik and Nicholson 2019) and autonomy and privacy (Wiener et al. 2019) of independent contractors.

The above organizational and individual changes in how OIS is managed have been complemented by technical changes that enable sourcing from numerous diverse sources, including cloud computing. Infrastructure providers like Microsoft, Amazon and Google have been introducing new business models such as cloud services, which have clear effects on how OIS is being governed and will be in the future. Another important change we are seeing in certain situations, such as Uber (Wiener et al. 2019), is that some components of the software remain on-premise while other software resides on the cloud (Hoffmann et al. 2019; Maier et al. 2019) or on distributed devices.

Thus, we are seeing several changes in how OIS is being done. These changes are occurring at the levels of organizations (client, vendor, and their relationship), individuals (either contractors or employees of the client or the vendor), as well as technologies (including the enabling role played by cloud computing and the external agent role played by robots).

5 What Are the Implications of Outsourcing?

The impacts of OIS are naturally being affected by the above shifts. Whereas the focus of OIS was initially on reducing costs and somewhat later on promoting innovation (OIS), we see OIS as enabling growth (Ravindran 2019) and being the basis for value generation by firms (He et al. 2019). Ravindran (2019) finds that firms with low R&D investment and firms with low industry concentration use OIS to pursue growth, whereas firms with high R&D investment and firms with high industry concentration use it to pursue efficiency. Moreover, He et al. (2019) distinguish between expected and realized value from outsourcing, and between value to the firm and value to the external party (the crowd in this paper). We also see the societal impacts of OIS, such as through companies that provide marginalized individuals with work opportunities (Malik and Nicholson 2019), create entirely new forms of employment (e.g., Uber, as in Wiener et al. 2019), and offer new ways to rent resources (e.g., Airbnb).

The new benefit areas from OIS are also accompanied by new kinds of risk. The papers in ICOIS 2019 discuss several such risks, including a loss of transparency (Bozan et al. 2019), data breaches (Benaroch 2019), project drift (Huber et al. 2019), knowledge leakage (Kishore et al. 2019), concentration risk (Gozman et al. 2019), and risk of contagion (Gozman et al. 2019). The last two of these, discussed by Gozman et al. (2019), may need further explanation. The trend toward outsourcing to multiple vendors or individuals, and to a crowd or the cloud, creates a potential "concentration risk" when a large number of clients rely on a small number of vendors, thereby making that large number of clients susceptible to any infrastructural support issues at those few vendors. It also creates a potential "risk of contagion," wherein a large number of external vendors being involved, but they themselves depend on a small number of vendors for the cloud infrastructure.

In the last four sections, I have examined how the field of OIS has changed in terms of what is being outsourced, to whom, and why, and the implications of these changes in terms of resulting benefits and risks. Next, let's examine how OIS researchers are responding to these changes. More specifically, how has the OIS research changed in terms of the literature areas being drawn upon and the research methods being used? The next two sections examine these two aspects.

6 What Literature Informs Outsourcing Research?

Following the 2001 ICOIS, Klein (2002, p. 24) wrote in his review chapter (similar to this):

> Outsourcing research has flourished since the nineties and with this has come an increasing variety of theoretical perspectives that inform current outsourcing research. However, no satisfactory coherent theory is currently proposed and the selection of theoretical bases (reference theories and methods) for outsourcing research still appears to be more ad hoc than systematic (p. 24).

> I was also surprised to find an unusually large number of reference theories represented in this conference. Most likely the twelve reference theories and frameworks identified in Sect. 2 do not even exhaust all of the theoretical reference theories and frameworks that have been used in the outsourcing literature elsewhere. Therefore it is fair to say that outsourcing research experiments with an unusually large set of theoretical perspectives. While this testifies to the openness and rigor of the field, it is nonetheless surprising that none of the twelve identified here overlaps with those that are most commonly used to explore social issues in other areas of IS research (p. 30).

My review of the papers from the 2019 ICOIS suggests that things have remained similar to 2001 in terms of the diversity of theoretical perspectives. The 22 papers in the 2019 ICOIS have drawn upon 15 different theoretical perspectives, ranging from requirements engineering (Hoffmann et al. 2019) to community logic (Malik and Nicholson 2019). However, there are several differences from 2001. First, only one paper (Chang et al. 2019) in the current 22 explicitly draws upon transaction cost economics (TCE), a sharp drop from eight of the 21 in 2001. Two of the 22 in the current set use strategy literature, specifically on value creation (He et al. 2019) and value chains (Gopal 2019). This is about the same as in 2001, where Klein viewed 3 of the 21 papers as using strategy theory. Klein found two papers to focus on relational exchange, whereas two papers in 2019 draw upon the literature on trust (Benaroch 2019; Kishore et al. 2019), and another on the literature on culture (19). Economic risk literature was used in one paper in 2001, whereas three papers draw upon the literature on broader (not just economic) set of risks in 2019. Some of the literature areas from 2001—game theory, the legalistic perspective and the software capability maturity model—do not seem to be used in the 2019 papers. By contrast, the current papers use literature on several areas, including requirements engineering (Hoffmann et al. 2019), project drift (Huber et al. 2019), control (Wiener et al. 2019), organizational integration (Maier et al. 2019), knowledge management (Kishore et al. 2019; Krancher and Dibbern 2019), that were not evident in the 2001 set.

The theoretical foundations for the current set of papers, as well as the broader contemporary OIS literature, can be viewed in terms of four broad categories:

(a) theories focused on the project, including requirements engineering (Hoffmann et al. 2019), risk (Benaroch 2019; Chang et al. 2019; Gozman et al. 2019), control (Wiener et al. 2019), and drift (Huber et al. 2019);
(b) theories focused on the organizational or interorganizational context and actions, including literature on trust (Benaroch 2019; Kishore et al. 2019), culture (19; Su 2019), organizational integration (Maier et al. 2019), knowledge management (Kishore et al. 2019; Krancher and Dibbern 2019), strategy, specifically value-creation (Gopal 2019; He et al. 2019), and TCE (Chang et al. 2019);
(c) literature focused on technology, specifically robots (Asatiani et al. 2019; Rutschi and Dibbern 2019); and
(d) theories beyond traditional organizations, including institutional logic (Malik and Nicholson 2019), community logic (Malik and Nicholson 2019), and the literature on privacy (Wiener et al. 2019), crowdsourcing (He et al. 2019), and ecosystems (Hurni et al. 2019).

Whereas theories related to (a) and (b) were used in 2001 as well, the specific theories and their relative use (for example the decreased reliance on TCE) have changed. But the most significant change is in the emerging use of theories in categories (c) and (d), i.e., those beyond human beings (i.e., robots) or beyond traditional organizations, respectively. This change can be directly linked to shifts I discussed earlier (in Sect. 3)—toward outsourcing to robots and to a significantly larger number of external individuals or organizations.

The theoretical shift can also be viewed as a shift in the value perspective of the studies. In his 2002 article, Klein noted: "The value perspective underlying the twelve theoretical perspectives that were identified appears to be oriented toward management." The current set of papers show a clear shift. Whereas categories (a) and (b) above to be seem to be generally oriented toward management, theories in category (c) are oriented more toward individuals beyond traditional organizational boundaries (e.g., Uber drivers, crowds, and marginalized sections of the society).

7 What Research Methods Are Being Used to Study Outsourcing?

Klein noted a clear use of interpretivist research methods in the papers from the 2001 ICOIS. His following remarks are relevant:

> Most papers show interpretivist influence. The question here is, to what extent is this part of a planned research strategy and to what extent is it simply in the nature of exploratory, preliminary research, on which conferences tend to focus in contrast to premier journal publications? (Klein 2002, p. 27).

> … most authors appear to incorporate or are at least open to incorporating elements of interpretivism in their research methods. The observation brings two comments to mind. First, in that sense, current outsourcing research does not share the positivist bias that Orlikowski observed for the premier research journals in 1991. It needs to be checked further, whether this is true only for outsourcing conference papers, which often are more avant-garde than journal papers. However, if we take the observation of prevailing openness to mixing elements from positivist and interpretive research methods as a serious trend, then methodologically something very interesting is going on in outsourcing research. (Klein 2002, p. 27).

Although the use of interpretivist methods is less apparent in the 2019 papers, qualitative methods continue to be predominant. 12 of the 21 papers used qualitative methods, including field interviews (Asatiani et al. 2019; Bozan et al. 2019), a single case study (Malik and Nicholson 2019), a case study with one vendor and clients from three countries (Su 2019), multiple case studies within the same company (Hoffmann et al. 2019), multiple case studies across different companies (Huber et al. 2019; Krancher and Dibbern 2019; Maier et al. 2019), longitudinal qualitative studies (He et al. 2019; Hurni et al. 2019), and qualitative analysis of secondary data (Meiser and Beimborn 2019; Gozman et al. 2019). However, six of the papers are quantitative, including three event studies (Benaroch 2019; Chang et al. 2019; Kishore et al. 2019), and one each based on a survey (Wiener et al. 2019), agent-based modeling

(Gopal 2019), and secondary panel data (Ravindran 2019). Although four papers in 2001 used surveys, none used event studies, agent-based modeling, or secondary panel data. The remaining four papers (Jarvenpaa and Markus 2019; Aubert and Rivard 2019; Rutschi and Dibbern 2019; Zimmermann 2019) are conceptual in nature (compared to five in 2001).

Thus, although qualitative methods and conceptual articles continue to play an important role in OIS research, there seem to be some shifts in research methods: (a) the qualitative methods seem to be used in a more positivist fashion than before; (b) quantitative methods seem to be used to a greater extent than before; (c) both qualitative and quantitative methods seem to be using more diverse techniques, with qualitative studies benefitting from multiple cases within the same organization, longitudinal designs, or qualitative analysis of secondary data (in addition to single or multiple case studies that were also used in 2001), and quantitative studies benefitting from event studies, agent-based modeling, or secondary panel data (in addition to questionnaire survey, which was also used in 2001); (d) there seems to be greater use of secondary data than before. The diversification of research techniques could to some extent be due to a natural methodological progress in research methods, but it also reflects the diversification of the OIS field in terms of what is being outsourced, to whom, and how. The greater use of secondary data could be related to the greater use of secondary data in IS research in general (in part due to the progress in the literature on economics of IS), as well as to the greater availability of secondary data sources, especially with the progress in cloud computing and social media.

8 Conclusions

The state of practice and research on OIS has changed significantly over time, as the papers in ICOIS 2019 depict. Below are some overall comments regarding these changes.

First, ***the overall pattern*** that emerges from the changes discussed above, and summarized in Table 1, is that the outsourcing phenomenon has become more important, with new industries being rooted on outsourcing and significant societal impacts, more diverse (in terms of both what is outsourced, and to whom), more complex (in terms of how outsourcing is done and managed), and more risky as well. The research on outsourcing seems to be paralleling the phenomenon, as evident from the greater range of theoretical foundations as well as research methods.

Second, ***the nature of the actors*** in OIS arrangements seems to be undergoing a transformation. Whereas such arrangements traditionally involved one client and one or a few vendors, now we increasingly encounter outsourcing to a large number of external agents, both individual and organizational. This creates situations where there are some central and some peripheral actors in OIS arrangements (Hurni et al. 2019). With crowds, small cloud-based firms, robots, and marginalized individuals being now among potential external agents to whom information services can be outsourced, individuals managing such arrangements have to be aware of

Table 1 Outsourcing and outsourcing research: the major emergent changes

Focal aspects	Emergent changes
The Phenomenon	
What?	• Increasing diversity (e.g., outsourcing cybersecurity) • Increasing importance (e.g., Uber and other businesses built entirely on outsourcing)
To whom?	• Increasing diversity (e.g., to marginalized individuals) • Increasing breadth of sources (e.g., crowdsourcing) • Increased use of IT itself as the source (e.g., robots)
How?	• Increasing uncertainty tolerance by clients (e.g., outsourcing to unknown agents) • Increasing use of innovative mechanisms by vendors (e.g., launching incubator programs) • Increasing complexity of sourcing arrangements (e.g., simultaneous cooperation and competition) • Increasing distribution of ITs and data across clients, vendors, and others (e.g., cloud services)
So what?	• Rising societal impact (e.g., through businesses founded on outsourcing) • Increasing range of risks (e.g., loss of transparency)
The Research	
Which literature?	• Reduced reliance on transaction cost economics (only 1 of the 21 papers in ICOIS 2019) • Four broad categories of theoretical perspectives, including theories focused on: (a) the project; (b) the organizational context and actions; (c) technology; and (d) aspects beyond the organization • Increasing range of theoretical perspectives (15 across the 21 papers in ICOIS 2019) across the above four categories, especially with new theories from aove categories (c) and (d) being used
Which research methods?	• Continued predominance of qualitative methods (12 of the 21 papers in ICOIS 2019) • Greater use of positivist (rather than interpretivist) qualitative methods • Increasing use of quantitative methods (6 of the 21 papers in ICOIS 2019) • Increasing diversity of research techniques (e.g., agent-based modeling)

the marked distinctions among such actors. For example, outsourcing to technology (robots) is currently being done based on task division and fragmentation, based on the view that humans are good at certain things and computer are good at other things, but robots are absorbing increasingly larger scope of tasks, raising concerns of deskilling and job degradation (Asatiani et al. 2019). I hope, and expect, this to be only a short-term phenomenon, especially considering the increasingly smart nature of humans consuming IT. As Aubert and Rivard (2019) find, we are moving toward "deconstructed" information systems (data, algorithms, platforms, etc.,) that can be assembled or disassembled as needed, and human skills, such as those discussed in the next point, will continue to be important.

Third, the way in which ***outsourcing arrangements*** are governed is undergoing major changes as well. We are seeing a shift from contracted and collaborative arrangements to arms-length and digitally-mediated structures (Bozan et al. 2019). We are also seeing partners in outsourcing arrangements place greater emphasis on the level of employee coaching by each other (Meiser and Beimborn 2019), as well as on cultural sense-making and bonding by individuals (Su 2019). Moreover, while managers' technical, business, and application knowledge clearly matter in OIS, their cognitive frames are being highlighted as well (Krancher and Dibbern 2019). Given the changes in the nature of the OIS, it is no surprise that the environment, and changes in it, affect the nature of the benefits from outsourcing (Ravindran 2019). Governance is thus becoming increasing dynamic in nature (Huber et al. 2019), leading to a temporal view of emerging value as well (He et al. 2019).

Fourth, one shift in terms of ***the research on outsourcing*** (beyond those discussed above and summarized in Table 1) seems to be that outsourcing is becoming embedded in research in other areas such as business value of IT and design science. Indeed, some of the papers in the 2019 ICOIS could be viewed as belonging to these streams, rather than "management of IS," wherein IS outsourcing research has historically been viewed. Although this permeation of outsourcing research across the field of IS might make it more difficult to delineate "outsourcing research," I consider it to be an encouraging trend, both for outsourcing researchers and the field of IS.

Finally, the emerging trends discussed in this article have some clear ***implications for executives*** charged with managing OIS projects. OIS seems to increase both firm returns and risks, making it imperative that firms consider both aspects and not outsource IT when it increases risk and lowers returns, as may be the case with strategic IT (Chang et al. 2019). Moreover, when Firm A terminates contract with a vendor and that vendor enters into an outsourcing contract with Firm's competitive rival (say Firm B), stock market reacts negatively to Firm A's stock. This implies that firms should exercise caution when considering the termination of outsourcing relationships, especially ones that are long term (Kishore et al. 2019). Executives in charge of OIS efforts should also carefully consider the implications of how their firms move across the value chain in the OIS context (Gopal 2019). Moreover, they should recognize the complementarity between the macro- (firm's strategy) and micro- (employees' goals) perspectives of outsourcing (Zimmermann 2019).

In conclusion, much has changed in the world of OIS. It has become bigger (not just in the number or cumulative dollar amounts of outsourced contracts, but also the number of agents involved as sources), better (as seen in its foundational role in new businesses such as Airbnb and Uber), and more complex (with a plethora of options to which outsource to, including firms, individuals, crowds, and robots). Perhaps even more drastic changes lie ahead, as some of the other emerging ITs, such as "big data," blockchains, and social media join the party that has started with cloud computing and artificial intelligence. How far are we from firms being able to leverage value by using blockchains to quickly and anonymously outsource IT services to unknown vendors monitored through robots?

References

Asatiani, A., Penttinen, E., Ruissalo, J., & Salovaara, A. (2019). Knowledge workers' reactions to a planned introduction of robotic process automation—Empirical evidence from an accounting firm. In *International Conference on Outsourcing of Information Services.*

Aubert, B.A., & Rivard, S. (2019). The outsourcing of IT governance. In *International Conference on Outsourcing of Information Services.*

Benaroch, M. (2019). Cyber security and IT outsourcing—Challenges and emerging realities. In *International Conference on Outsourcing of Information Services*.

Bozan, K., Lyytinen, K., & Rose, G. (2019). Outsourcing and Microservice architectures (MSA): What do you buy when you move from raising thoroughbreds to feeding a school of gold fish? In *International Conference on Outsourcing of Information Services*.

Business Week (1986). Special report: The Hollow corporation. March 3, 56–85.

Chang, Y., Han, K., & Hahn, J. (2019). Risk and return implications of it outsourcing: The moderating role of contract duration and prior relationship. In *International Conference on Outsourcing of Information Services.*

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. *Information Systems Research, 14*(3), 291–314.

Dibbern, J., & Hirschheim, R. (2020). Introduction: Riding the waves of outsourcing change in the era of digital transformation.

Dibbern, J., Goles, T., Hirschheim, R., & Jayatilaka, B. (2015). Information systems outsourcing: A survey and analysis of the literature. *ACM SIGMIS Database: The DATABASE for Advances in Information Systems, 35*(4), 6–102.

Gopal, A. (2019). Is the grass greener? On the strategic implications of moving along the value chain for IT service providers. In *International Conference on Outsourcing of Information Services.*

Gozman, D., Machaiah, T., & Willcocks, L. (2019). When it rains it pours: Cloud sourcing and the risk of contagion in the financial sector. In *International Conference on Outsourcing of Information Services.*

He, R., Kotlarsky, J., & Nevo, D. (2019). Understanding how value emerge between a firm and a crowd in crowdsourcing projects: A process perspective. In *International Conference on Outsourcing of Information Services.*

Hoffmann, P., Mateja, D.N., Sporer, K., & Heinzl, A. (2019). Bridging the vendor-user gap in enterprise cloud software development through data-driven requirements engineering. In *International Conference on Outsourcing of Information Services.*

Huber, T., Dibbern, J., Fischer, T., & Kirsch, L. J. (2019). How and why software projects drift: A process view. In *International Conference on Outsourcing of Information Services.*

Hurni, T., Huber, T., & Dibbern, J. (2019). Emerging innovation ecosystems: The critical role of distributed innovation agency. In *International Conference on Outsourcing of Information Services.*

Jarvenpaa, S. L., & Markus, M. L. (2019). Data sourcing and data partnerships: Illustrations from biodata. In *International Conference on Outsourcing of Information Services*.

Kishore, R., Iriyama, A., & Gunupudi, L. S. V. (2019). Vendor opportunistic behavior in IT outsourcing post relationship termination. In *International Conference on Outsourcing of Information Services.*

Klein, H. (2002). On the theoretical foundations of current outsourcing research. In R. Hirschheim, A. Heinze, & J. Dibbern (Eds.), *Information systems outsourcing: Enduring themes, emergent patterns and future directions* (pp. 24–43). Berlin: Springer.

Kotlarsky, J., Oshri, I., Dibbern, J., & Mani, D. (2018). MIS Quarterly Research Curation on IS Sourcing.

Krancher, O., & Dibbern, J. (2019). Unravelling specific-knowledge transfer: A multiple-case study exploring the what, the how, and the challenges of knowledge transfer to outsourcing vendors. In *International Conference on Outsourcing of Information Services.*

Lacity, M. C., Hirschheim, R. A. (1993). *Information systems outsourcing; myths, metaphors, and realities*. Wiley: Chichester.
Lacity, M. C., & Willcocks, L. (1998). An empirical investigation of information technology sourcing practices: Lessons from experience. *MIS Quarterly, 22*(3), 363–408.
Lacity, M. C., & Rottman, J. W. (2012). *Emerging ITO and BPO markets: Rural sourcing and impact sourcing.* IEEE Computer Society Press.
Lacity, M. C., Rottman, J. W., & Carmel, E. (2015). Prison sourcing: 'Doing good' or 'good for business'? *Journal of Information Technology Teaching Cases, 4,* 99–106.
Maier, T., Heinzl, A., & Kude, T. (2019). Challenges in deploying SaaS solutions: Who provides the cloud infrastructure service? In *International Conference on Outsourcing of Information Services.*
Malik, F., & Nicholson, B. (2019). The role of institutions and corporate social strategy in impact sourcing. In *International Conference on Outsourcing of Information Services.*
Mead, R. The Airbnb Invasion of Barcelona. *The New Yorker.* April 22.
Meiser, S., & Beimborn, D. (2019). Innovation in outsourcing—An empirical analysis of outsourcing vendors' innovation approaches. In *International Conference on Outsourcing of Information Services*.
Pastin, M., & Harrison, J. (1987). Social responsibility in the Hollow corporation. *Business and Society Review, 87*(63), 54–58.
Projects, Information Systems Research, 14(3), 291–314.
Ravindran, K. (2019). Information technology outsourcing: An ambidextrous perspective. In *International Conference on Outsourcing of Information Services.*
Rutschi, C., & Dibbern, J. (2019). From human-executed to robot- automated routines. In *International Conference on Outsourcing of Information Services.*
Sabherwal, R. (1999). The role of trust in managing outsourced IS development projects. *Communications of the ACM, 42*(2), 80–87.
Sabherwal, R., Hirschheim, R., & Goles, T. (2001). The dynamics of alignment: Insights from a punctuated equilibrium model. *Organization Science, 12*(2), 179–197.
Srnicek, Nick. (2016). *Platform capitalism*. Cambridge and Malden: Polity Press.
Su, N. (2019). Cultural sensemaking in digital outsourcing. In *International Conference on Outsourcing of Information Services.*
Wiener, M., Cram, W. A., & Benlian, A. (2019). Perceptions of control legitimacy in the gig economy: Conceptual dimensions and empirical illustrations. In *International Conference on Outsourcing of Information Services.*
Willcocks, L., Oshri, I., Kotlarsky, J., & Rottman, J. (2011). Outsourcing and offshoring engineering projects: Understanding the value, sourcing models, and coordination practices. *IEEE Transactions on Engineering Management, 58*(4), 706–716.
Zimmermann, A. (2019). A systemic understanding of micro level mechanisms and firm level design of global sourcing. In *International Conference on Outsourcing of Information Services*.
Zuboff, S. (1988). *In the age of the smart machine: The future of work and power* (p. 1988). New York: Basic Books.