JYVÄSKYLÄ STUDIES IN COMPUTING
212

Yikun Lu

An Information System Design Product Theory for the Class of eSourcing Requirements, Delivery and Completion Management Systems for eSourcing Service Providers

JYVÄSKYLÄN YLIOPISTO

JYVÄSKYLÄ STUDIES IN COMPUTING 212

Yikun Lu

An Information System Design Product Theory for the Class of eSourcing Requirements, Delivery and Completion Management Systems for eSourcing Service Providers

Esitetään Jyväskylän yliopiston informaatioteknologian tiedekunnan suostumuksella julkisesti tarkastettavaksi yliopiston Agora-rakennuksen auditoriossa 2 maaliskuun 13. päivänä 2015 kello 12.

Academic dissertation to be publicly discussed, by permission of the Faculty of Information Technology of the University of Jyväskylä, in building Agora, auditorium 2, on March 13, 2015 at 12 o'clock noon.

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2015

An Information System Design Product Theory for the Class of eSourcing Requirements, Delivery and Completion Management Systems for eSourcing Service Providers

JYVÄSKYLÄ STUDIES IN COMPUTING 212

Yikun Lu

An Information System Design Product Theory for the Class of eSourcing Requirements, Delivery and Completion Management Systems for eSourcing Service Providers

UNIVERSITY OF JYVÄSKYLÄ

JYVÄSKYLÄ 2015

Editors
Seppo Puuronen
Department of Computer Science and Information Systems, University of Jyväskylä
Pekka Olsbo, Ville Korkiakangas
Publishing Unit, University Library of Jyväskylä

URN:ISBN:978-951-39-6081-0
ISBN 978-951-39-6081-0 (PDF)

ISBN 978-951-39-6080-3 (nid.)
ISSN 1456-5390

Jyväskylä University Printing House, Jyväskylä 2015

ABSTRACT

Lu, Yikun
An Information System Design Product Theory for the Class of eSourcing Requirements, Delivery and Completion Management Systems for eSourcing Service Providers
Jyväskylä: University of Jyväskylä, 2014, 76 p. (+ included articles)
(Jyväskylä Studies in Computing
ISSN 1456-5390; 212)
ISBN 978-951-39-6080-3 (nid.)
ISBN 978-951-39-6081-0 (PDF)

Information and Communications Technology (ICT)-enabled international sourcing of software-intensive systems and services (eSourcing) is increasingly used as a means of reducing costs, increasing quality, searching missing skills and capabilities, and achieving strategic objectives. To reap maximal benefits from eSourcing and mitigate the risks, service providers and clients have to be aware of and build capabilities for the entire eSourcing life-cycle. China is growing fast in the international eSourcing service provisioning market, but most Chinese eSourcing service providers are small or medium-sized and typically work for larger intermediaries instead of end-clients, limiting their business and capabilities development. The current literature does not extensively address this business model and ways to overcome its limitations. To overcome this gap in research, this doctoral thesis focuses on eSourcing life-cycle management from service providers' perspective and probes the eSourcing life-cycle management in Information and Communications Technology Sourcing (ICTS) and Business Process Sourcing (BPS) contexts. The specific research domains are sourcing services for software testing and logistics. This thesis presents the best practices for eSourcing service providers and creates an Information System Design Product Theory (ISDT) for each domain. The thesis also develops an ISDT for the class of eSourcing requirements, delivery and completion management systems, which offers as generalizable scientific knowledge as possible concerning the most important business practices for eSourcing service providers from the viewpoint of service provisioning, breakdown recovery, and the redesign of the eSourcing life-cycle. eSourcing service providers can use it to establish domain-specific design product theories and to instantiate them into information systems that support the design, service provisioning, and breakdown recovery within the eSourcing life-cycle, which could help eSourcing service providers to overcome the mentioned limitation for Chinese sourcing providers. While the research focuses on Chinese service providers, the results can be generalizable to service providers in other nations with powerful eSourcing industries.

Keywords: information system design product theory, eSourcing life-cycle, software testing, logistics

Author's address Yikun Lu
Department of Computer Science and Information Systems
University of Jyväskylä
P.O.Box 35 (Agora)
FIN-40014 Jyväskylä, Finland
yikun.y.lu@jyu.fi, yikun_l@hotmail.com

Supervisors Timo Käkölä
Department of Computer Science and Information Systems
University of Jyväskylä

Reviewers Anol Bhattacherjee
College of Business
University of South Florida
Tampa, Florida, USA

Bill Hefley
Katz Graduate School of Business & College of Business Administration
University of Pittsburgh
Pittsburgh, Pennsylvania, USA

Opponents Professor Matti Rossi
School of Business
Aalto University

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest gratitude to Dr. Timo Käkölä, my advisor and dear friends. Without his continuous encouragement and support in the last five years it would have been impossible for me to grow and pursue my goals in the academic community and in my life. He helped me a lot in my academic objectives and career development. His profound knowledge and industry experience helped me to organize my study in software engineering and engineering business management and to build my research in information systems.

I am grateful to all the employees in the case companies who participated in this research. They answered questions, participated in interviews, provided feedback and offered all the help they could. I especially want to thank Xin He, the CEO of the testing case company, and Yousan Tang, the CIO of the logistics case company. They accepted my visit request, offered significant support for data collection and interviews, provided feedback and reviewed the research papers. Without their patient help, this research would not have achieve progressed very far.

I would like to thank my parents who supported me all throughout and encouraged me to carry on with my life.

My special thanks to my dear girlfriend, Han, who made me feel warm in the cold of Finland for the last three years. Her persistence, encouragement and considerateness made it possible for me to pursue my dreams.

In the past four years, the research has received financial support from CIMO, COMAS, eBEREA and LIIKESIVISTYSRAHASTO. Without this funding support, it would have been impossible for me to collect the data and interview stakeholders during the past four years and complete this research.

Jyväskylä 25th November 2014
Yikun Lu

FIGURE

TABLE

CONTENTS

This dissertation includes five research papers, which are referred to in the text by their roman numerals.

I Lu, Y. and Käkölä, T. (2011) Which Test Artifacts Testing Service Providers should Reuse and How? Experiences from a Case Study in the Chinese ICT Sourcing Market. In R.H. Sprague Jr. (Ed.), proceeding of 44th Hawaii International Conference on System Sciences (HICSS). Kauai, Hawaii, IEEE Computer Society, 1-10.

II Lu, Y. and Käkölä, T. (2014) A Dynamic Life-cycle Model for the Provisioning of Software Testing Services. Journal of Systems Science & Control Engineering, 2(1), 549-561.

III Lu, Y. and Käkölä, T. An Information Systems Design Product Theory for Integrated Requirements, Test and Defect Management Systems. Journal of Information Systems and e-Business Management. (will submit it soon)

IV Lu, Y. and Käkölä, T. (2013) An Information System Design Product Theory for Integrated Order, Transportation and Warehouse Management Systems. In R.H. Sprague Jr. (Ed.), proceeding of 46th Hawaii International Conference on System Sciences (HICSS). Wailea, Hawaii, IEEE Computer Society, 3717-3726.

V Lu, Y. and Käkölä, T. (2014) An Information System Design Product Theory for the Abstract Class of Integrated Requirements and Delivery Management Systems. In R.H. Sprague Jr. (Ed.), proceeding of 47th Hawaii International Conference on System Sciences (HICSS). Waikoloa, Hawaii, IEEE Computer Society, 3677-3686.

The first author did most of the work in each conference article. The first author is responsible for collecting and analyzing the data in all the articles that form part of this research. The second author assisted in revising and provided overall guidance for each conference article. Both authors contributed equally to the writing of both journal papers.

1 INTRODUCTION

International sourcing of software-intensive systems and services (eSourcing) enabled by information and communications technology is used worldwide as a means to reduce costs, improve quality, share risks, access skills and achieve strategic aims (Barthelemy & Geyer, 2005; Gorla & Somers, 2014). Already in the beginning of this millennium, over 50% of the American Fortune 500 firms and a significant proportion of Western European and Japanese firms leveraged offshore software sourcing (Carmel & Agarwal, 2002; Sahay, Nicholson, & Krishna, 2003). International sourcing is used not only by large enterprises but also by small and medium-sized enterprises, who also try to use this strategy to reduce costs and provide high-quality services (Carmel & Agarwal, 2002; Carmel & Nicholson, 2005). India, Russia, Philippines and China are important nations for service provisioning (Carmel & Nicholson, 2005; Carmel, Gao & Zhang, 2008). The use of Information and Communications Technology (ICT) is crucial in eSourcing because "it is expected to be a key factor in the remodeling of how and where business is done and how organizations create value. As such, ICT contributes to the appearance of networked organizations, the shaping of virtual enterprises, the development of partnerships and the formation of strategic alliances." (Van der Zee & Ribbers, 2000, 1)

International sourcing has been available for many years, but the reported success rate of the projects is still comparatively lower than expected (Herath & Kishore, 2009). To enhance the success rate, considerable amount of research on multiple aspects is conducted. For example, clients and service providers need to manage their relationship effectively with appropriate risk mitigation, coordination, and control strategies (Carmel & Nicholson, 2005; Levina & Ross, 2003; Sabherwal, 1999, 2003), which help them to cooperate closely and execute their engagement effectively (Gottschalk & Solli-Sæther, 2006). Selecting the providers cautiously helps clients to control sourcing projects and establish trust with sourcing service providers (Earl, 1996; Lacity & Hirschheim, 1993; McFarlan & Nolan, 1994). Trust is important for clients and sourcing service providers to establish long term strategic relationship and cooperation (Hart & Saunders, 1997; Ring & Van de Ven, 1994). The types of contracts in different scenarios have been analyzed (Wang, Barron & Seidmann, 1997; Lee, Miranda & Kim, 2004; Aron, Clemons & Reddi, 2005; Herath & Kishore, 2009) to formal processes in relationships management and facilitate long-term relationships for clients and service providers (Kern, Willcocks & Van Heck, 2002; Shi, Kunnathurb & Ragu-Nathan, 2005; Lee, Miranda & Kim, 2004). Sourcing projects need to be governed through structural mechanisms, such as deliverables, penalty clauses, and reporting arrangements (Sabherwal, 1999). However, most research in sourcing focuses on clients (Plugge, Bouwman & Molina-Castillo, 2013), among them the clients from the US and Europe (Koveos & Tang, 2004). The service providers' perspective has not been studied sufficiently (Feeny, Lacity & Willcocks, 2005; Borman, 2006; Plugge, Bouwman & Molina-Castillo, 2013). In fact, service providers are important in international sourcing because offshoring of services is critically dependent on a supply of providers with operational and strategic capabilities to offer comparative cost advantages, satisfactory quality, and on-time delivery despite the differences in distance, time zones, and culture (Carmel & Tjia, 2005). However, international sourcing research is limited to the service model based on India and to their American clients (Kaiser & Hawk, 2004; Oshri, Kotlarsky & Willcocks, 2007). American and European enterprises are familiar with the large Indian sourcing service providers and their sourcing services. For example, Indian companies gained 90% of US enterprises' sourcing business in 2002 (Schniederjans, Schniederjans & Schniederjans, 2007, 20). In 2008, India had some 65 per cent of the ICT sourcing and 43 per cent of the business process sourcing market (Oshri, Kotlarsky, Rottman, & Willcocks, 2009).

India has dominated the global sourcing market for several years, but its role will change (Rottman & Lacity, 2004; Oshri, Kotlarsky, Rottman, & Willcocks, 2009). China's national policies and its growing economy are encouraging multinational firms to seriously consider China as a major complement in their sourcing strategy (Oshri, Kotlarsky, Rottman, & Willcocks, 2009). The government helps the software industry and companies to enhance the importance and prevalence of exporting IT services in order to enhance Chinese role in the global sourcing market. Chinese sourcing service providers initiated a strategic shift from domestically focused markets to international markets in 2000, and the main export markets have expanded from Japan and South Korea to Europe and North America (Carmel, Gao, & Zhang, 2008). However, the Chinese sourcing market is less mature than the Indian sourcing market, and unlike the Indian market, it is not dominated by several big sourcing providers. The top ten Chinese sourcing providers account for less than 30 per cent of the Chinese sourcing market (Ziqun et al., 2007). Most Chinese providers leverage the mediated offshore outsourcing business model, whereby a small or a medium-sized Chinese provider delivers offshore software services to a larger foreign ICT client that contracts and interfaces with the actual end-clients onshore (Järvenpää & Mao, 2008). This business model usually restricts the providers to small, low-value projects and hampers the sharing of knowledge with end-clients, severely impeding the capability and business development of Chinese providers. Järvenpää and Mao (2008) focus on the client-specific, process, and human resources capabilities development, but their research does not cover the entire eSourcing life-cycle and cannot enhance service providers' capabilities comprehensively. The current literature does not extensively address this business model and ways to overcome its limitations. Rottman and Lacity (2004) describe the twenty important practices for CIOs to help them to work with sourcing service providers and mitigate risks. This research, however, does not cover the entire eSourcing life-cycle. Momme (2002) describes a comprehensive framework for the eSourcing life-cycle from clients' perspective. The framework includes the main activities and performance measures for each phase. Based on this framework, Käkölä (2008) presents the best practices framework for international eSourcing and proposes the supporting ICT tools for each phase of the sourcing life-cycle to manage the eSourcing life-cycle comprehensively. In his research, the eSourcing life-cycle was divided into seven phases from clients' perspectives, starting from clients making a strategic sourcing analysis and decision. The best practices framework includes main activities, supporting ICT tools, performance measures and expected outcomes in each phase, but the introduced supporting ICT tools are for clients, not for providers. The eSourcing Capability Model for Service Providers (eSCM-SP, 2010) is the most comprehensive eSourcing model available for service providers, but this model does not include the ICT tools to support providers' practices during the eSourcing life-cycle. At the end of Chapter 3 of this doctoral thesis, the best practices framework for international eSourcing service providers is presented. Chapter 4 creates an information system design product theory for the class of eSourcing requirements, delivery and completion management systems to help sourcing service providers to manage the entire sourcing life-cycle and support their activities in the sourcing life-cycle. The best practices framework for eSourcing service providers includes the main activities in each phases, the supporting ICT tools, performance measures and expected outcomes. Current literature provides little theory-based guidance to help companies to design and use information system design product theory for the class of eSourcing requirements, delivery and completion management systems to achieve the goals of improved product quality, cycle time reduction, service delivery, and overall effectiveness. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems and the other supporting ICT tools work together to support the entire eSourcing life-cycle.

"Design theories, unlike other theories, support the achievement of goals (Hevner, March, Park & Ram, 2004; Markus, Majchrzak & Gasser, 2002; Van Aken, 2004; Walls et al., 1992, 2004). An information systems design theory (ISDT) is "a prescriptive theory based on theoretical underpinnings which says how a design process can be carried out in a way which is both effective and feasible" (Walls et al., 1992, 37). It prescribes both the design product and process aspects of a class of information systems, namely, (1) the value propositions to be fulfilled by implementing an instance of the class, (2) meta-requirements describing the problem(s) to be solved by the class, (3) the meta-design prescribing the class of artifacts used to solve the problem(s), and (4) applicable kernel theories from social and natural sciences for understanding, governing, and/or solving the problem(s) shared across all products within the class, and how the products should be built (Walls et al., 1992, 2004)."(Käkölä et al, 2010, 3)

Following Walls et al. (1992), the doctoral thesis creates information system design product theories for the classes of eSourcing requirements, delivery and completion management systems that support eSourcing life-cycle from providers perspective and help clients and providers to establish dynamic capabilities and enable them to (1) interact transparently, (2) monitor the sourcing life-cycle in real time, (3) identify communication and breakdowns, (4) reconfigure resources flexibly, and (5) create knowledge to recover from breakdowns and to redesign routines in order to proactively eliminate similar breakdowns in future. Breakdowns in work routine impacts service progress and quality, which requires stakeholders to create solutions or new knowledge to get services back to work routine (Heidegger, 1977; Bell & Zemke, 1987).

eSourcing can be divided into two categories: ICT sourcing (ICTS) and business process sourcing (BPS), and in the global market there are some large eSourcing service providers, such as Flextronics, Infosys, and Wipro (Manning, Massini & Lewin, 2008; Engardia & Einhorn, 2005). ICT sourcing occurs when an organization contracts one or more providers to perform an ICT function instead of performing the function themselves. The service provider could be a third party or another division or a subsidiary of a single corporate entity (ITAC, 2003). "ICT has had a major impact on the way business is conducted, especially with traditional value chains disintegrating and organizational boundaries becoming blurred." (Cachia & Kruger, 2007, 33) BPS involves the sourcing of non-core business processes, along with their ICT support, to internal or external service providers. It enables clients to focus on their primary business operations and to achieve a combination of lower costs, improved productivity, and flexible staffing options (Schniederjans, Schniederjans & Schniederjans, 2007, 245-246). The major reasons for BPS are to reduce costs associated with the business process as well as access to quality services (Kim & Kim, 2008; Lacity & Willcocks, 2014). The usually sourced processes have ranged from sales/marketing, accounting/finance and human resource to call centers (UN, 2003). ICTS and BPS have been among some of the biggest business trends and highest IT growth sectors in developed economies in the beginning of this millennium (Willcocks, Hindle, Feeny & Lacity, 2004). Clients make the sourcing decision not only to reduce labor costs, but also to access pools of highly talented individuals around the world (Manning, Massini & Lewin, 2008; Lewin & Peeters, 2006; Farrel, Laboissiere & Rosenfeld, 2006). Corresponding to this, the sourcing business has increasingly involved product development functions, such as engineering, research and development, and production (Manning, Massini & Lewin, 2008; Subramaniam & Venkatraman, 2001; Patel & Vega, 1999).

This research uses the case study research methodology (Eisenhardt, 1989; Cunningham, 1997; Runeson & Höst, 2009) for studying one service provider in each of these two contexts to offer scientific knowledge concerning the most important business practices, activities, and classes of information systems for eSourcing (1) ICT services and (2) business processes. The research investigates the two contexts, respectively, through the following two eSourcing domains in the Chinese eSourcing market: (1) software testing services and (2) third-party logistics services (hereafter called logistics services). The two research cases are Ltesting (http://en.ltesting.com.cn/) and PG Logistic (PGL) (http://www.pgl-world.com/).

The chosen two research domains and the two case firms have close relationships with the Chinese government's development strategy. In 2006, the Ministry of Commerce set up the 1000-100-10 Project to nurture 1000 outsourcing companies, to attract 100 major foreign companies to use offshore services in China, and to establish 10 outsourcing parks (Carmel, Gao & Zhang, 2008). Testing services are a part of this 1000-100-10 Project because a majority of Chinese providers can offer them. To support software industry development, government-issued policies including tax incentives, tax-free zones, local incentives to ease creation of software parks, and special tax rates for employers of IT staff (Oshri, Kotlarsky, Rottman, & Willcocks, 2009). To overcome the global financial crisis in 2009, the Chinese government set up a plan to restructure and revitalize the top ten industries, among them the logistics industry (XNA, 2009).

Modern logistics services need to successfully align the ICT infrastructure and business strategy (Hughes & Kaplan, 2009). It is a proper research domain which studies how ICT-enabled business processes build business value and improve service quality, offering a comprehensive perspective for the eSourcing of logistics services from the service provider's viewpoint. However, most of the literature neither covers ICT-enabled logistics services holistically (Gunasekaran & Ngai, 2003a; Zhang & Han, 2010) nor takes the perspective of the eSourcing service providers (Lai, Zhao & Wang, 2006; Zhang & Han, 2010). For example, integrated logistic information system management (Chiu, 1995) focuses on electronic data interchange (EDI) among the distribution systems, but the relevant transportation and warehouse management in the logistics life-cycle are not considered in this research. Other early research on integration logistics services focuses on seamless integration of the logistics functions among various stakeholders (Stock et al,. 1998, 2000; Childerhouse and Towill, 2003). Frohlich and Westbrood (2001) proposed five different logistics integration strategies for clients and service providers, but the research does not explain how to integrate the logistics information in the logistics process integration for clients and service providers, the relevant transportation management and warehouse management having not been integrated with order management. The research of Sheu et al. (2006) concluded that better IT capabilities and information sharing between clients and service providers help them to achieve better coordination and problem-solving activities. The research of Zhou et al. (2007) and Li et al. (2009) found that information sharing has significant impact on integration of logistics systems and delivery performance, but their research has no direct effect on IT implementation performance or on integration order, transportation and warehouse management (Prajogo & Olhager, 2012). The existing literature offers the service providers limited guidance to design and execute integrated ICT-enabled logistics services supporting the entire logistics life-cycle from product or service design to delivery.

Ltesting is a medium-sized (less than 50 employees) software testing services provider (Ltesting, 2014). It is suitable for this research because it has established a leading position in the Chinese testing service market and has testing experience from multiple domains (e.g., banking, insurance and telecommunications) and professional services. It can offer various types of testing-related services. There are two reasons for selecting PG Logistics (PGL) as the second research case company. First, PGL is the leading and most influential third-party logistics enterprise in the Chinese market. It is also the first Chinese company to provide clients with integrated logistics services. Second, PGL has developed its own flexible and scalable third-party logistics information integration platform.

Therefore, this research examines software testing services and logistics services to identify the most crucial classes of information systems and to integrate them into the class of eSourcing requirements, delivery and completion management systems that support the entire eSourcing life-cycle. While this doctoral thesis focuses on Chinese eSourcing service providers, it seems likely that the results are generalizable to providers in other nations with powerful eSourcing industries.

The rest of this introductory chapter is organized as follows. Section 1.1 presents the research questions addressed in this dissertation. Section 1.2 briefly describes the theoretical foundations of this research: the eSourcing Capability Model for Service Providers (eSCM-SP) and the Information Systems Design Theory (ISDT) for Dual Information Systems (DIS). Section 1.3 describes the included original articles and their relationships. The last section describes the structure of the dissertation.

1.1 Research questions

This doctoral thesis focuses on the abilities of Chinese eSourcing service providers to (1) establish and implement efficient practices and business models throughout the eSourcing life-cycle, (2) recover from unanticipated coordination breakdowns quickly and effectively, and (3) design and use technology infrastructure and information systems that enable effective enactment, breakdown recovery, and redesign of the eSourcing life-cycle. Therefore, this research addresses the following research question: Which eSourcing practices, associated activities, and enabling classes of information systems are the highest priority ones for eSourcing service providers from the viewpoint of executing the eSourcing life-cycle, recovering from coordination breakdowns during execution, and redesigning the life-cycle practices, and which are the systems to ensure organizational long-term effectiveness?

This doctoral thesis focuses on the following four sub-questions from service providers' perspective during the case studies and cross-case research (1) What are the major activities in each phase of the eSourcing life-cycle and how are these activities managed? (2) What are the performance measures of each phase? (3) Which information and communication technology tools best support the eSourcing life-cycle and each phase? (4) What are the internal relationships between the organizational structure, enabling classes of information systems, information systems architecture, business strategy, and the eSourcing life-cycle?

It is important to establish and execute effective service delivery processes and business models throughout the eSourcing life-cycle. Routine practices and enactment will work under normal conditions, and this is important for achieving efficient organizational performance and service delivery (Käkölä and Taalas, 2008). Recovering from unanticipated coordination breakdowns quickly and effectively is also important (Käkölä and Taalas, 2008). By analyzing breakdowns and the underlying causes, stakeholders in the workplace can identify the problems that are not easily visible in normal routines and create new knowledge to solve such problems. Redesigning the eSourcing life-cycle when necessary ensures organizational survival, proactive elimination of some breakdowns, and effective long-term enactment of routines (Käkölä and Taalas, 2008). Analyzing specific eSourcing activities and enabling information systems holistically as work systems (Alter, 2006) help uncover: (1) which eSourcing practices and associated activities have the highest priorities for service providers in ensuring service and organizational long-term effectiveness and (2) which enabling classes of information systems help the most in accomplishing the business objectives. Therefore, this doctoral research uses the sub-questions in both case studies and in the cross-case study to guide data collection and analysis.

1.2 Theoretical foundations

The eSourcing Capability Model for Service Providers (eSCM-SP) is the foundation of this research because it is the most comprehensive eSourcing model available from service providers' perspective. According to eSCM-SP, a life-cycle (Table 1) has three phases, (1) an engagement is carried out and involves gathering and negotiating requirements with a client, contracting, designing, resourcing, and deploying a service, (2) the service is delivered according to the commitments established for the engagement, and (3) the engagement is completed primarily by transitioning the resources from the provider to the client or to a third party (eSCM-SP, 2010).

TABLE 1 The eSCM-SP V2.01

Ongoing practices represent management functions that need to be performed during the entire eSourcing life-cycle in order to satisfy the intent of these practices.		
Initiation	Delivery	Completion
Practices focus on the capabilities needed to effectively prepare for service delivery. The practices are concerned with collecting and analyzing service requirements, negotiating, contracting, and designing and deploying the services, including the transfer of the necessary resources.	Practices focus on service delivery capabilities, including the ongoing management of service delivery, on the verification that commitments are being met, and on the management of the finances associated with service provision.	Practices focus on the capabilities needed to effectively close an engagement with particular client(s) at the end of the eSourcing life-cycle. They include the capture of the lessons learned from the engagement and the transition of resources to the client, or to a third party, from the provider.

Ongoing management function practices take place throughout the life-cycle. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, performance management), and there are specific practices in each phase. This research uses eSCM-SP v2 (eSCM-SP, 2010) as the reference model, the capability areas of this version including a total of 84 specific practices. eSCM-SP prescribes five capability levels. Therefore, certified assessors can use eSCM-SP as a reference model to determine the capability levels of service providers, and clients can use the certifications to find and select service providers. Service providers can use eSCM-SP as a roadmap to improve their capabilities to higher levels. This investigation collected data and compared the practices of the case firms to eSCM-SP based on the three phases and specific practices. Therefore, the eSCM-SP helps this research to analyze the eSourcing life-cycle services from the perspectives of service processes.

The authors of eSCM-SP convincingly argue that (1) eSCM-SP is applicable to both ICT and business process sourcing and (2) it can help service providers improve their capabilities related to both ongoing, phase-specific, and engagement-specific eSourcing practices throughout the eSourcing life-cycle (eSCM-SP, 2010). Yet, eSCM-SP has not been used or studied extensively in China. This doctoral research employed only a relatively small subset of the relevant best practices envisioned in eSCM-SP for Chinese service providers, mainly because most providers are in relatively early phases of eSourcing capability development (Oshri, Kotlarsky, Rottman, & Willcocks, 2009) and thus cannot use the most advanced practices of eSCM-SP. For example, the practices of *encourage innovation* and *career development* are not used in either of the two case companies. Providers with the capabilities of maturity managing applied to organization performance who proactively enhance value could use these practices.

Information Systems Design Theory (ISDT) for Dual Information Systems (DIS) is also the kernel theory for this research. As the businesses of the case companies are growing, the existing information systems in the case companies restrict them from improving service quality and sharing information with clients. For example, to meet client complex requirements and share important information timely, the logistics case company bought a new warehouse management system to replace the original one developed by themselves, because the new warehouse management system can offer more detailed information (e.g., inventory information, cargo type, size) on service and share the information with clients timely. Information systems design theory for DIS helps bridge the design/use dualism, because the conceptual design of most information systems reflects a design/use dualism of technology, making it difficult for users to feel responsible for the computerized aspects of work (Käkölä & Taalas, 2008). Information Systems Design Theory (ISDT) for Dual Information Systems (DIS) has three layers: knowledge-base layer, business-system layer and project-system layer. The knowledge-base layer stores and makes accessible reusable current and historical work and information system designs and their justifications and performances in the organizational units. "The knowledge-base layer of Dual Information Systems is a repository of explicit work and IS design knowledge in the knowledge-base layer of a hypertext organization." (Käkölä & Taalas, 2008, 4) "Hypertext organization is a dual organization structure coordinating the allocation of time, space, and resources so an organization can achieve high performance in routines and ensure long-term survival. Hypertext organization is comprised of knowledge-base, business-system, and project-system layers." (Käkölä & Taalas, 2008, 4) Work routines are enacted in the business-system layer. The project-system layer provides a field of interaction where project teams create knowledge. Knowledge is also created through the circular movement of actors amongst the layers (Käkölä & Taalas, 2008). Information Systems Design Theory (ISDT) for Dual Information Systems (DIS) helps the doctoral research to systematically analyze the eSourcing life-cycle services from the viewpoints of the different layers.

1.3 The relationships among the included articles

This dissertation contains five articles focusing on the research objectives from different aspects of the research questions. The relationship between the articles and the structure of the doctoral research project is as follow:

Articles I, II, and III focus on software testing services. Article IV focuses on logistics services. A cross-case analysis is conducted in Article V and in Chapter 4.

Article I describes the reusable test artifacts for testing service providers, which helps testing service providers to meet client requirements such as lower costs, shorter service times, and to improve software quality and productivity. This article finds test plans, test cases, and test reports to be the most important test artifacts that can be reused in the testing services. Lessons learned from the testing services need to be accumulated so as to improve providers' services capability in the future. Test environments and test procedure documents can be reused in specific domains. Test case patterns and test plan templates are high-level artifacts, which can save time and help novices to perform test services effectively. Significant reuse work is implemented in the initiation phase when test managers and test analysts choose suitable reusable test artifacts from the test database and define the best practices to meet client requirements. Testers provide valuable feedback concerning the reused artifacts, thus improving the quality of domain-specific test artifacts and the usefulness of test artifact database. Test tools can be reused too. For example, HP Quality Center offers test management services which can be reused across various types of testing projects. Reusable test artifacts research is helpful for the industry and service providers. The created Information System Design Product Theory for the class of Requirements, Test and Defect Management Systems (RTDMS) (Article 3), as well as its meta-requirements and meta-design are reusable artifacts. Testing service providers should be able to reuse them for process redesign and acquisition and/or development of the information system design product theory for the class of RTDMS instances.

Article II creates a comprehensive eSourcing life-cycle model for testing services, enabling service providers and clients to manage the sourcing life-cycle effectively. This article also presents the most important class of information systems for testing service providers, that is, the class of Requirments, Test, and Defect Management Systems (RTDMS), which helps testing service providers to transcend the limitations of the mediated sourcing business. The most important testing practices are requirements analysis and test planning, primarily conducted in the initiation phase.

Article III presents an Information Systems Design Product Theory (ISDT) for the class of Requirements, Test and Defect Management Systems (RTDMS). The theory helps clients and testing service providers to manage and control the testing process, carry out the service process standardization, and improve service effectiveness. This paper establishes the meta-requirements and the meta-design of the the Information System Design Product Theory for the class of RTDMS. Its instances can be used for testing services by any testing service provider. These Information System Design Product Theory for the class of RTDMS instances are expected to enforce standardized processes for testing service providers and implementation of best practices across testing projects. The Information System Design Product Theory for the class of RTDMS instances send timely information on services to clients and service providers, which make the testing services process transparent and seamless. Test analysts, developers, and testers share and reuse artifacts from the project database across projects to raise productivity and quality. Developers can share knowledge about defects across projects to increase efficiency and reduce risks. Test managers and clients can aggregate quality metrics across projects. This article presents the stakeholders of comprehensive testing and their performance in the testing life-cycle. Measurement management, configuration management and quality assurance management teams work with the testing team to execute the testing services.

Article IV focuses on the logistics services research, creating an Information System Design Product Theory for the class of Order, Transportation and Warehouse Management Systems (OTWMS), which helps clients and service providers to acquire and design information systems for designing managing and controlling logistic processes and improving service effectiveness. This article established the meta-requirements and the meta-design of the design product theory for the class of OTWMS. The theory helps logistics providers design, acquire, and use its instances for providing logistics services. These instances are expected to enforce standardized processes for logistics providers and the implementation of best practices across services. Order management, transportation management, and warehouse management teams can share and reuse artifacts from the database to raise productivity and quality. Clients can share logistics knowledge across service engagements to increase efficiency and reduce risks. Logistics providers and clients can aggregate quality metrics across service engagements.

Article V and Chapter 4 focus on the cross-case research. Article V presents an Information Systems Design Product Theory (ISDT) for the class of Requirements and Delivery Management Systems (RDMS). This theory is partly derived (1) deductively from comprehensive kernel theories such as eSourcing Capability Model for Service Providers (eSCM-SP) and (2) inductively from the domain specific design product theories for the classes of RTDMS and OTWMS. The theory prescribes an abstract class of systems because instances of the class need not be built. The theory is primarily used to create more detailed domain-specific design product theories. The design product theories for the classes of OTWMS and RTDMS are domain-specific theories used to prescribe information system subclasses of the class prescribed by the information system design product theory for the class of RDMS. The theory is expected to help eSourcing service providers and commercial software vendors to design domain-specific integrated systems for service provisioning and breakdown recovery throughout the eSourcing life-cycle in a variety of ICT and business process sourcing domains, helping clients and service providers to manage and control the eSourcing life-cycle and to make the process transparent and seamless. The doctoral research found that the completion phase is also important for the information system design product theory for the class of eSourcing requirements and delivery management systems (RDMS). Therefore, RDMS name has been changed to information system design product theory for the class of eSourcing requirements, delivery and completion management systems in this dissertation.

Chapter 4 compares the common and variable parts of the information system design product for the classes of RTDMS and OTWMS and generalizes an information system design product theory for the class of eSourcing requirements, delivery and completion management systems for eSourcing service providers. This chapter also compares the information system design product theory for the class of eSourcing requirements, delivery and completion management systems with kenrel theories to clarify the contributions of this doctoral research.

1.4 The structure of the dissertation

This dissertation is organized as follow. Chapter 2 presents the research methodology, data collection and analysis processes. Chapter 3 describes the results of the testing and logistics case study. An information systems design product theory has been created, respectively, for each domain. Based on the two case studies and literature reviews, this research generalizes the best practices framework for international eSourcing service providers and presents this best practices framework in the end of this chapter. Chapter 4 describes the cross-case analysis and creates an information system design product theory for the class of eSourcing requirements, delivery and completion management systems. The last chapter presents the contributions and limitations of this doctoral thesis.

2 RESEARCH METHODOLODY

This chapter describes the research methodology, data collection and analysis processes. The team structure of software testing and logistics services providers is also explained. The team members' performance and their responsibilities in the eSourcing life-cycle are described as well.

2.1 Research methodology

This thesis classifies eSourcing into ICT sourcing (ICTS) and business process sourcing (BPS) categories and studies each category through a case study. This research used a single qualitative case study to provide a holistic, systemic understanding of the phenomenon of eSourcing services provisioning in the context of testing services and logistics services (Eisenhardt, 1989; Runeson & Höst, 2009; Cunningham, 1997). The data collected for the research covers a complete life-cycle for service providers, including the most important business practices, the artifacts reused in these practices, and the employee responsible for these tasks. The common and variable aspects among the two case studies are summarized and a generalizable result from the viewpoint of eSourcing service providers is drawn up. The eSourcing Capability Model for Service Providers (eSCM-SP) was chosen as the reference model in the research. It has been demonstrated to help various types of providers to improve their capabilities related to both ongoing, phase-specific, and engagement-specific sourcing practices throughout the sourcing life-cycle. The eSCM-SP life-cycle involves three phases from the provider's viewpoint: initiation, delivery, and completion. Ongoing practices are run throughout the life-cycle to perform management functions. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, and performance management). The investigation collected data and compared the practices of the two case organizations to eSCM-SP based on the three phases and specific practices.

During both case studies and the cross-case study, the research analyzed the data by iterating between two phases. First, the information systems used by them and the information on the routines, the most significant breakdowns in routines, and the processes and information systems used for recovering from breakdowns were compared to the eSourcing phases and practices prescribed by the eSCM-SP. Service breakdowns caused by poorly designed, poorly used, and/or entirely missing information systems were of special interest to the research. Interactions between the eSourcing processes, practices, organizational structures, and information systems were analyzed to define the most important information systems for the entire eSourcing life-cycle and service providers. Second, the results were shown to key managers and stakeholders in the case companies, feedback from the managers and the stakeholders was collected, and the results were revised as necessary and summarized. In the cross-case analysis, the common and variable parts of the two case studies and the created information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS were compared and analyzed with respect to each phase of the eSourcing life-cycle.

For the first testing case study, the research conducted two rounds of investigation in the case company in Beijing. For the first round (from the end of 2009 to the beginning of 2010), the author spent over three weeks observing life in the case company, analyzing documents and memoranda, and interviewing key personnel. The interviewees involved the CEO, all testing managers, and a number of test analysts to uncover the routine practices and information systems associated with testing work and major breakdowns disrupting work. The research used a questionnaire for each interview and concluded with an open discussion to address emerging issues. Interviews were summarized and sent to the interviewees, who verified them and provided feedback as necessary. If the analysis indicated that major deviations existed or information was missing, clarifications were requested from informants through email. The data collection and analysis process continued for several months using the internet to collaborate with the case company. A year after the first round of interviews, the first author performed a second round (from the end of 2010 to the beginning of 2011) in the case company to collect supplementary data related to breakdowns and workarounds. For the second round, the quality assurance manager, the measurement process manager, and other people supporting the test teams were also interviewed.

For the second logistics case study, the research also conducted two rounds of investigation in the case company. For the first round (in the end of 2010), a subsidiary of the corporate entity was investigated on-site in Beijing. It delivers logistics services based on the Total Order Management (TOM) information systems instance developed by the case company. The instance includes three main subsystems: Order Management System (OMS), Warehouse Management System (SMS), and Transportation Management System (TMS), enabling the offering of an integrated solution for clients' supply chain management. They support service delivery and proactively help to avoid service breakdowns in order to meet client requirements. SMS offers basic warehouse management functionality. To meet complex requirements, the case company bought another Warehouse Management System (WMS). The investigation found that (1) ways the workers in the company use the subsystems for providing logistics services and (2) the types of breakdowns occurring. The second round of data collection (from the end of 2011 to the beginning of 2012) was conducted by visiting the headquarters of the company on-site in Guangzhou to probe how TOM’s subsystems supported each service phase of eSCM-SP.

This doctoral thesis sets out to span two different eSourcing domains to organize a cross-case research, the most challenging part of this research. Therefore, the cross-case research results need to generalize the common and variable aspects of the eSourcing life-cycle among the two different eSourcing domains. Domain engineering literature (Akoka, 2005, 461-463) and Alter's research (2006, 2008) on service system fundamentals were used as a theoretical basis to overcome the challenge. Domain engineering was used to identify, model, construct, catalog, and disseminate a set of software artifacts to enable the development and maintenance of software in each eSourcing service domain (Akoka, 2005, 461-463). Domain engineering includes three main activities: (1) domain analysis identifies a domain and captures its ontology. It should specify the basic elements of the domain, identify the relationships among these elements, and represent this understanding in a useful way, (2) domain design and (3) domain implementation are concerned with mechanisms for translating requirements into systems that are made up of components with the intent of reusing them to the highest extent possible. Service systems produce all services of significance and scope. For understanding and analyzing service systems, Alter’s research (2008) presents three frameworks: (1) the work system framework provides a system-oriented view of any system that performs work within or across organizations, (2) the service value chain framework augments the work system framework by introducing functions that are specifically associated with services, and (3) the work system life cycle model focuses on work systems’ change and evolution over time, treating the life cycle of a system as a set of iterations involving service routines and unplanned change. A service system is thus a useful fundamental unit of analysis for understanding, analyzing, and designing services and better service systems in marketing, operations, and information systems research (Alter, 2008). Therefore, these two bodies of literature help the research to create abstractions that enable the comparison of the two eSourcing domains. For example, in the initiation phase of eSourcing, through comparing the service providers’ practices and the support offered by the instances of information system design product theory for the class of RTDMS and by information system design product theory for the class of OTWMS, the common and variable parts were analyzed to draft the requirements management services for information system design product theory for the class of eSourcing requirements, delivery and completion management systems.

In the eSourcing service, providers use sub-teams to execute specific services. For example, in the testing case company, test teams are responsible for testing service delivery. Usually the teams have four roles: test manager, test analyst, tester, and coordinator (Table 2). Coordinators are like boundary spanners between clients and service providers. They are especially important in solving communication challenges in international eSourcing when clients and service providers use different languages and have different cultures (Ma, Li, Chen, Conradi, Ji & Liu, 2008; Poston, Simon & Jain, 2010). Additionally, test teams can be organized flexibly based on the project characteristics, personnel workloads, and client requirements. For example, when the projects are small, testers do not need to be involved in test teams because test managers and test analysts can do their work. Article 2 and 3 present more details on the team members performance in the testing life-cycle.

TABLE 2 Job Descriptions of Key Roles in Test Teams

Title	Responsibility
Test Manager	Test managers are responsible for test project planning, management, risk evaluation, and report review. During project completion, test managers summarize the projects and the lessons learnt. They have at least five years of relevant work experience.
Test Analyst	Test analysts analyze test requirements, design test plans (together with the test manager), and design test cases. They have from three to four years relevant work experience.
Tester	Testers conduct the specific test assignments. They have more than one year’s work experience.
Coordinator	Coordinators communicate with clients, acting as bridges between clients and service providers. They need to have comprehensive testing knowledge, because they attend the testing service life-cycle from early bidding and negotiation through to service completion.

To deliver comprehensive testing sourcing services, test teams still need support from other teams, namely, measurement management teams, quality assurance management teams and configuration management teams (Table 3). The three teams work with test teams and clients throughout the testing life-cycle. They provide relevant information and support to help test teams and clients to track the service processes, control the service quality, and manage the testing life-cycle. Article 2 and 3 give more information on their performance in the testing life-cycle.

Logistics teams are responsible for delivering specific logistics services. Usually the most important services are delivered by three sub-teams: order management team, transportation management team, and warehouse management team (Table 4). These teams need to work together to process orders, analyze client requirements, draft logistics plans and deliver services. Article 4 presents more details on their performance in the logistics life-cycle.

TABLE 3 Job Descriptions of Teams Involved in Testing Sourcing Services

Title	Responsibility
Test Team	Test teams are responsible for test assignments, which include drafting test plans, executing tests, finding and reporting defects, and cooperating with other teams to deliver test services that meet clients' requirements.
Measurement Management Team	Measurement management teams are responsible for collecting and analyzing metrics data to help test teams and clients to improve and manage the quality of test services and to monitor the progress of testing services delivery. Clients and test teams can adjust or even redesign service processes based on measurement results.
Quality assurance Management Team	Quality assurance management teams are responsible for ensuring that testing services can meet clients' requirements and are run in the right way. Appropriate quality criteria can help to ensure that test teams focus on the most important tasks that need more resources.
Configuration Management Team	Configuration management teams are responsible for managing the configurations of different versions of test artifacts during the services. They deal with version conflicts, and manage changes of test artifacts.

TABLE 4 Responsibilities of the Key Logistics Teams

Title	Responsibility
Order management team	The team is responsible for order validation, entry, and processing. It makes the logistics plan for the order together with the transportation team and the warehouse management team. During the delivery phase, it also coordinates with the transportation and warehouse management teams. It is solely responsible for communicating with clients. During order completion, it documents the services and the lessons learnt.
Transportation management team	The team defines the most efficient transportation schemes according to client requirements and drafts the logistics plan with the order management team. It executes the delivery service and reports any transportation breakdowns (e.g., delays, accidents, and unforeseen stops).
Warehouse management team	The team uses Warehouse Management System to manage the cargo and support the transportation team (e.g., ensuring the cargo is ready for delivery when the transportation team arrives).

Based on the two case studies and eSourcing Capability Model for Service Providers (eSCM-SP), this thesis assigns the eSourcing services providers to two separate service delivery sub-teams: the requirements management team and the delivery management team (Table 5). These teams need to work together, for example, to process and analyze client requirements, to draft eSourcing service plans, to deal with service breakdowns and to deliver services. In order to track the service progress and to monitor the service providers' performance in the service, clients need to get relevant information on the services during the eSourcing life-cycle.

TABLE 5 Responsibilities of the Key eSourcing Service Provisioning Teams

Team	Responsibility
Requirements management team	The team is responsible for transforming clients' requirements to executable requirements, requirements prioritization, and management. This team makes the eSourcing service plan together with the delivery management team. During the delivery phase, it coordinates with the delivery management team and deals with service breakdowns. It is responsible for communicating with clients.
Delivery management team	The team executes the service delivery according to client requirements, drafts, together with the requirements management team, the eSourcing service plan and reports service breakdowns (e.g., delays, accidents, and out-of-budget events). It collects metrics and deals with breakdowns during the service. During the service completion, it documents the services and the lessons learnt and transfers resources to clients or third parties.

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems should be abstract and generic enough so that eSourcing service providers could use it to design domain-specific information systems and improve their processes and information systems regardless of their current practices and systems. Providers can thus use even separate management systems (e.g., requirements management systems and delivery management systems) to better integrate and manage their systems for enabling the end-to-end eSourcing life-cycle. For example, information system design product theory for the class of eSourcing requirements, delivery and completion management systems can track the requirements execution process against the service plan and report execution results and breakdowns. It does not need to help execute specific tasks, but it needs to track and report the results of the tasks. Specific tasks can be run by using other management tools. For example, the testing case company uses RequisitePro to manage requirements and CVS (Concurrent Versions System) or SVN (Apache Subversion) to manage artifacts versions, but they use HP Quality Center (QC) to manage the entire testing life-cycle.

"*For example, the requirements management tool is not as good as RequisitePro. The version control feature is quite basic compared to CVS or SVN. Therefore, sometimes we need to use more than one tool in the services, including specific tools for specific tasks (e.g., test execution or version management). Then we will transfer the results to QC for comprehensive test management.*" -A test manager

Therefore, the analysis of the practices and information systems of the case companies has helped this research to scope the information system design product theory for the class of eSourcing requirements, delivery and completion management systems appropriately.

3 THE CLASSES OF INFORMATION SYSTEMS FOR ESOURCING SERVICE PROVIDERS OF SOFTWARE TESTING AND THIRD PARTY LOGISTICS SERVICES

This chapter presents the most important classes of information systems for software testing services and logistics services. The information system design product theories for the class of RTDMS and OTWMS are described, respectively, in section 3.1 and 3.2. Based on the case studies and literature reviews, a best practices framework for international eSourcing service providers is generalized and presented in section 3.3.

3.1 The most important classes of information systems for software testing service providers

To effectively manage the entire eSourcing life-cycle, eSourcing service providers need to closely co-operate with geographically distributed clients. They need to develop effective communications to deal with a wide range of issues (e.g., differing organizational cultures between clients and service providers) that can hamper collaboration (Poston, Simon and Jain, 2010). Effective communication between clients and service providers and within and between different sub-teams of service providers helps the service providers to enhance competitiveness, add economic value, and improve service quality and market branding (Sharma, Apoorva, Madireddy & Jain, 2008; Barthelemy, 2001; Constantinescu, 2005).

To support collaboration and knowledge management throughout the testing life-cycle, sophisticated information systems are needed. Yet, most commercially available systems that focus on the testing life-cycle only provide limited support for clients and service providers. For example, they may use separate requirements management, test execution management, or defect management for the entire life-cycle. For this reason, it is difficult to support standardized data transfer between the different information systems during specific life-cycle phases, which reduces service effectiveness and increases the risk of failure.

Salo and Käkölä (2005) found that requirements management systems (RMS) need to be designed and used to redesign and effectively enact the earliest phases of software development in geographically distributed organizations. However, RMS would only offer limited benefits, if RMS were not integrated with the systems used in the down-stream phases. For example, client representatives responsible for entering requirements could not use RMS to track whether the requirements would be fulfilled, which decreased their interest in that task. Therefore, systems need to be designed to support the life-cycle more comprehensively. Käkölä et al. (2010) found that integrating RMS with software release management systems improves life cycle support the most. After all, value can only be created when the right requirements are met in the right releases. Käkölä et al. (2010) identified test and defect management systems as the next most important system classes that should be integrated with the requirements and release management systems, because a piece of software can be effectively released only when the requirements it realizes have been verified and validated and when the expected number of remaining, undetected defects is acceptable.

Article 2 found that the class of requirements, test and defect management systems is the most important class of information systems for testing service providers, the initiation phase proved most important in the testing sourcing life-cycle. It was also found that requirements analysis and test planning are the most important testing practices, primarily conducted in the initiation phase. Lu and Käkölä (Article 2) created a comprehensive ICT-enabled sourcing life-cycle model for testing services in order to help clients and service providers to manage the testing sourcing life-cycle effectively. However, instances of information system design product theory for the class of requirements, test and defect management systems that successfully enable the entire life-cycle are scarcely available. HP Quality Center, Microsoft Visual Studio Test Professional, and Rational Quality Manager are the most successful integrated requirements, test, and defect management products commercially available in the market.

3.1.1 An information system design product theory for the class of Requirements, Test, and Defect Management System

The existing literature provides little theory-based guidance to help companies to design and use Requirements, Test and Defect Management Systems to achieve the goals of improved product quality and cycle time reduction, service delivery, and overall effectiveness. Therefore, Article 3 creates an information system design product theory for the class of Requirements, Test and Defect Management Systems (RTDMS). The article prescribes the product aspects of information system design product theory for the class of RTDMS, the existing literature does not provide such a theory. Moreover, instances of information system design product theory for the class of RTDMS can be built in many ways, and it is thus not as fruitful to prescribe the process aspects as the product aspects. An instance of information system design product theory for the class of RTDMS helps providers to manage the entire testing life-cycle from test planning and requirements management through test execution and defect tracking and management to project closure, which builds upon the baseline set by Käkölä et al (2010), Article 2 and Lu and Käkölä (2011). A database supports each information system design product theory for the class of RTDMS instance by retrieving and storing important artifacts, such as requirements artifacts, test artifacts, and defect artifacts. During the testing execution process, the database also stores the automated test scripts and other test execution artifacts. The database includes reusable and other artifacts produced during all the projects that the service provider has served. Information system design product theory for the class of RTDMS instances use the database to help projects manage newly-created test artifacts and reuse test artifacts from previous projects. They also support testing team members' work and help the clients to monitor and track the testing service progress (Article 3). Clients and service providers can obtain timely information from information system design product theory for the class of RTDMS instances, which help them to manage the service progresses and organize the schedules. Article 3 presents more details about this.

In order to achieve such integration, several factors need to be considered. Firstly, client requirements and system requirements need to be transformed to detailed and executable test requirements. Specific test executions may uncover defects and other problems needing quick communication between clients and test teams and/or between test team members, but the clients and test teams may be globally distributed, making it difficult to conduct face-to-face meetings (Article 3). Changing client requirements or revising test plans may require revising contracts between clients and testing service providers and raises project risks. Moreover, financial, technological, business, and other risks for clients and service providers need to be considered before initiating a project (Elitzur & Wensley, 1997; Tafti, 2005).

The main contribution of testing services research is the information system design product theory for the class of Requirements, Test and Defect Management Systems (RTDMS). Open source communities and commercial product providers can build software products that meet the meta-requirements and the meta-design of the design product theory to enable effective testing sourcing services. The design product theory is validated based on (1) a case study focusing on a leading testing sourcing service provider that implements integrated organization-wide requirements, test and defect management processes and systems, (2) a literature review in the fields of test requirements management, test execution management, and defect management (Prasanna et al. 2005; Hong, Kim & Lee, 2010; Lazic, 2010), and (3) a study of HP Quality Center and other testing service management tools used in the market.

"RTDMS describes the service processes and our team members' (e.g., testers, quality assurance managers) responsibilities in the services, which are exactly the processes we are carrying out in our services. QC as an instance of RTDMS is used a lot in our services. Other testing management products have functions similar to QC, but QC is the most popular product in the market, and it can offer comprehensive management to cover the entire testing life-cycle. The other testing management products focus on defect management a lot, they cannot offer management for the entire testing life-cycle as comprehensive as QC." -CEO

The design product theory is partially based on the experiences of the case company and its use of HP Quality Center to ensure that the meta-design is flexible, comprehensive, and scalable, that is, the instances of information system design product theory for the class of RTDMS can deal with large numbers of test artifacts (e.g., test cases, test requirements and test scripts) and their relationships and support the teams involved such as test teams, measurement management teams, and quality assurance management teams (Article 3). The meta-design proposed in the testing services research is lightweight and easy to use, so that small and medium-sized companies can use (parts of) it (e.g., test requirements management) to execute information system design product theory for the class of RTDMS solutions.

Information system design product theory for the class of RTDMS instances help test managers and test analysts to prioritize and valuate client requirements. The project database stores the requirements and requirements' interdependencies. The prioritization and valuation methods are beyond the scope of the testing services research. The highest priority requirements have the maximum priority in using testing resources, and most testing resources are allocated to deal with the highest priority requirements (Article 3). Instances of information system design product theory for the class of RTDMS make it easy to track test executions and defects, because all test artifacts and defects are bidirectionally linked to requirements. Each test execution clearly indicates which test requirements provide the purpose for the execution and which defects have been detected by which specific test execution. Version management and baselining help identify the stakeholders involved with different versions of different artifacts and the actions the stakeholders have taken. Therefore, stakeholders can control requirements changes, analyze the impacts of requirements changes, and revise the test processes and test plans as necessary to meet the most important requirements and service breakdowns, as detailed in Article 3, which has more information on this.

3.2 The most important classes of information systems for logistics service providers

Logistics information systems facilitate logistics management. They allow clients to acquire logistics services and conduct business transactions via the internet (Liu, So, Choy, Lau & Kwok, 2008). The main purposes of logistics information systems integration are to achieve real-time capturing and sharing of key information along the logistics service processes, process unexpected coordination breakdowns quickly, and make logistics decisions (Prajogo & Olhager, 2012).

Investments in IT may fail to produce expected benefits unless clients and logistics service providers are willing to share logistics information (Fawcett, Osterhaus, Magnan, Brau & McCarter, 2007). Effective logistics can typically be achieved only when companies exchange not only transactional data (e.g., material or product orders) but also strategic logistics information helping stakeholders to make important decisions in their operations (Li, Ragu-Nathan, Ragu-Nathan & Rao, 2006). For example, the sales history information helps providers to forecast demand, improving service quality and efficiency, real-time inventory information helps providers to plan replenishment and delivery schedules, improving service quality and reducing inventory costs (Seidmann & Sundararajan, 1997; Lee, So & Tang, 2000). Effective knowledge sharing necessitates frequent and intense communication between clients and service providers and contributes to establishing cooperative relationships between partners. Therefore, high degrees of symmetrical flows of strategic information between partners will be likely (Klein, Rai & Straub, 2007).

A number of studies have demonstrated various benefits from information sharing with logistics partners in terms of inventory management (Cachon & Fisher, 2000; Lee, So & Tang, 2000; Yu, Yan & Cheng, 2001; Zhao, Xie & Zhang, 2002), improved agility and flexibility (Swafford, Ghosh & Murthy, 2008), and reduced bullwhip effect (that causes supply chain participants far from the end consumers to experience greater demand variations than participants close to the end consumers) (Dejonckheere, Disney, Lambrecht & Towill, 2004). Knowledge sharing significantly impacts the planning and delivery of logistics services (Zhou & Benton Jr, 2007). Providers' IT capabilities and knowledge sharing affect the integration of logistics systems directly and the organizational performance indirectly (Li, Yang, Sun & Sohal, 2009). Therefore, information integration and logistic actions integration, specifically, the order management, transportation management and warehouse management are needed for clients and service providers. Integration of logistics systems is a crucial way to realize these benefits and gain competitive advantage. Prajogo and Olhager (2012) propose the integration, but their research does not present detailed guidelines on the integration of order, transportation and warehouse management. They do not propose the integration logistic management systems for service providers, either.

Logistics service providers most commonly offer warehousing, transportation, and customs brokerage services (Langley, Allen & Colombo, 2003). Traditional core competencies of logistics service providers also include (1) booking services for air and sea freight forwarding services and (2) preparation of tailored documentation (Liu, So, Choy, Lau & Kwok, 2008). In sum, the most important logistics practices include freight forwarding services, warehousing services, and transportation services. Usually, forwarding services are based on specific order management services. Therefore, order management is also important for the integrated logistics systems.

Most logistics service providers in China are small or medium-sized. They subcontract parts of their operations to other supply chain partners (Liu, So, Choy, Lau & Kwok, 2008). Furthermore, poor communication channels (e.g., fax, phone, and email) lead to service breakdowns (e.g., service delays, human errors) and high operation costs. Due to these limitations, Chinese logistics clients are often unable to obtain up-to-date status information from their logistics service providers in real time for making timely decisions (Lu & Käkölä, 2013).

There are few studies on small and medium-sized logistics providers in the Chinese logistics market (Gunasekaran & Ngai, 2003b; Liu, So, Choy, Lau & Kwok, 2008). However, these logistics service providers have become increasingly important, because there has been a trend for logistics service providers to participate in clients' supply chains in order to provide logistics management (Liu, So, Choy, Lau & Kwok, 2008). Therefore, it is necessary to improve communication and operations for small and medium logistics service providers, which help them to offer services more effectively.

An Information System Design Product Theory for the class of Order, Transportation and Warehouse Management Systems (OTWMS) has been designed to help the logistics service providers to manage the logistics life-cycle from the service initiation phase through ordering, transportation execution, and warehouse tracking and management to service closure (Lu & Käkölä, 2013). Each information system design product theory for the class of OTWMS instance leverages database to store automated, OTWMS-generated logistics orders; schedule, transportation, warehouse and inventory information, and other relevant information produced during the logistics life-cycle (Lu & Käkölä, 2013). Information system design product theory for the class of OTWMS instances use the database to reuse logistics plans from previous services and manage newly-created logistics plans. Instances of information system design product theory for the class of OTWMS also support logistics team members' practices and help the clients capture and track the logistics service information. Clients and logistics providers can obtain real time information from information system design product theory for the class of OTWMS instances, which helps them to make decisions during the logistics life-cycle. Article 4 has more information about this.

In order to achieve such integration, several factors need to be considered. Firstly, client requirements need to be transformed to a detailed and executable logistics plan. Specific transportation management and warehouse management services must meet client requirements and deal with unexpected service breakdowns, necessitating quick but effective creation and sharing of knowledge within and between clients and logistics service providers (Lu and Käkölä, 2013). Clients and logistics service providers may be globally distributed, therefore, it is difficult to conduct face-to-face meetings with clients.

Changing client requirements or service breakdowns may require revising contracts between clients and logistics service providers and raise service risks.

There is little theory-based guidance to help logistics service providers and clients design and leverage such integrated systems (Lu & Käkölä, 2013). The logistics research draws upon a case study to create an information system design product theory for the class of OTWMS. A complete information system design theory (ISDT) prescribes both the product and process aspects of a class of information systems, that is, the meta-requirements and the meta-design for all the products instantiating the class; the kernel theories from reference disciplines that are vital to determine what the products should do, and how the products should be built (Walls, Widmeyer & EI Sawy, 1992). ISDTs make the development of products more tractable for application developers by focusing their attention and restricting their options. They also help organizations to source products and components from commercial and open source markets.

3.2.1 An information system design product theory for the class of Order, Transportation, and Warehouse Management System

Information system design product theory for the class of OTWMS was created in Article 4, which is based on a literature review and the analysis of the design and use of the innovative Total Order Management (TOM) system. TOM is an instance of the information system design product theory for the class of OTWMS and it is developed by the case company.

"*OTWMS has a logical and clear logistics service processes description that describes drafting service plan based on clients' requirements, signing contract, delivering services, monitoring services and dealing with breakdowns, and evaluating and terminating service by clients. TOM as an instance of OWTMS has additional financial management function, which help us to manage and evaluate the service. For example, we can get the cost and profit information from financial management.*"-CIO

The information system design product theory for the class of OTWMS has been designed to be abstract and generic enough so that logistics providers can use it to improve their processes and information systems, regardless of their current practices and systems (Lu and Käkölä, 2013). It may be possible for logistics service providers to benefit from the theory for information system design product theory for the class of OTWMS even without replacing any existing systems. Providers can thus use even separate order management systems, transportation management systems, and warehouse management systems and use the theory to better integrate and organize these systems to enable the end-to-end life-cycle (Lu & Käkölä, 2013). For example, an instance of information system design product theory for the class of OTWMS can track an order execution process against the logistics plan and report execution results and/or breakdowns. It does not need to help execute specific logistics tasks, but it needs to track and report the results of the tasks. A specific task can be run by other logistics management tools. For example, the logistic case company sometimes uses bought warehouse management systems instead of the original warehouse management systems developed by themselves for delivery services. However, they still use TOM (an instance of information system design product theory for the class of OTWMS developed by the case company) to manage the entire logistics life-cycle. Therefore, the analysis of the practices and information systems of the case company has helped the logistics research to scope the design product theory for information system design product theory for the class of OTWMS appropriately.

Instances of information system design product theory for the class of OTWMS help logistics providers to prioritize and valuate logistics requirements. The order database stores the requirements and their interdependencies. The prioritization and valuation methods are beyond the scope of the logistics research.

Most logistics resources are allocated to deal with the highest priority requirements. Information system design product theory for the class of OTWMS instances make it easy to track orders and transportation services, because transportation and warehouse management artifacts are bidirectionally linked to orders (Lu & Käkölä, 2013). In the case company, all the stakeholders involved in the service engagement record their actions and relevant information to instance of information system design product theory for the class of OTWMS, helping the order management team to manage service provisioning and to effectively communicate with clients.

For each logistics service engagement between a client and a provider, the used information system design product theory for the class of OTWMS instance used indicates which orders provide the purpose for the transportation and which warehouses are leveraged. Moreover, all practices can be tracked to responsible team members, improving accountability and service quality. Version management helps to identify the stakeholders involved with different versions of various artifacts and the actions the stakeholders have taken (Lu & Käkölä, 2013). Therefore, order management teams can analyze and control the impacts of order changes and revise the schedules and logistic plans as necessary to meet the most important requirements and recover from service breakdowns. Article 4 provides more details about this.

3.3 The best practices for international eSourcing service providers

Based on the two case studies and literature reviews on eSourcing services (Rottman and Lacity, 2004; Käkölä, 2008; eSCM-SP, 2010; Plugge, Bouwman & Molina-Castillo, 2013) as well as the best practices for eSourcing clients (Käkölä, 2008), this research generalizes the best practices framework for international eSourcing service providers to help them establish standardization service delivery processes, enhance service capabilities, improve service effectiveness, and use ICT tools to effectively execute eSourcing services (Käkölä & Lu, 2015).

This thesis employs the three main activities (domain analysis, domain design and domain implementation) of domain engineering (Akoka, 2005, 461-463) and the three frameworks (the work system framework, the service value chain framework and the work system life cycle model) of service systems (Alter, 2006, 2008) to generalize the common and variable aspects of the eSourcing life-cycle among the two different eSourcing domains. Based on domain analysis research, the basic elements of the eSourcing domain such as the involved stakeholders (e.g., clients, service providers and sub-teams of the service providers), the relationships among these stakeholders (e.g., the relationships between clients and service providers and the relationships among sub-teams), client requirements (e.g., requests for proposal) and service delivery-related artifacts (e.g., service plan and solutions for service breakdowns) have been identified. Domain design and domain implementation research focuses on the sourcing life-cycle, on service delivery processes, on how service providers meet client requirements in the service delivery processes and on how they reuse artifacts to shorten service time and improve software quality. For example, both testing and logistics service providers can reuse artifacts from previous similar projects to draft service plans and both of them have to deal with service breakdowns and/or clients change requirements and create solutions for breakdowns.

Based on work system framework analysis, this study analyzes how testing and logistic service providers deliver services within and across organizations. Based on service value-chain framework research, the specific delivered services and the interaction between clients and service providers are analyzed in the testing life-cycle and logistics life-cycle. The work system life-cycle model research helps this doctoral thesis to generalize how testing and logistics service providers deal with service routine and service breakdowns in the testing life-cycle and logistics life-cycle, respectively. For example, both testing and logistics services providers need to send service-relevant information (e.g., about service progress or service breakdowns) to clients during the service. Clients can adjust requirements based on the information and service providers need to relocate resources based on the revised client requirements.

The best practices include seven phases in the entire eSourcing life-cycle from service providers’ perspectives. Each phase contains the main activities, ICT tools used to support the practices in each phase, preconditions and performance measures, and the expected outcomes. FIGURE 1 presents the best practices framework in detail.

FIGURE 1 A set of best practices for international eSourcing of software products and services from eSourcing service providers' perspective

Information and communication technology infrastructure: Email, Internet, Internet telephone, Intranet, Mobile communication, Social Media, Teleconferencing, Telephone, Videoconferencing

The eSourcing life-cycle	Setting objectives and evaluating capabilities	International market research and evaluation	Bidding on contracts	Contract negotiation
Supporting classes of information systems	Decision support systems; Process asset library; Project estimation and measurement systems	Banner advertising; Blogs; Customer relationship management systems; Country-related market database; Newsgroups; Search engines; Trade statistics and industry-specific databases; Trading web sites; Webcasting	Customer relationship management systems; Newsgroups; Search engines; Project estimation and measurement systems; Requirements, test, defect, and release management systems; Trade mailing lists	Customer relationship management systems; Contract database; Project estimation and measurement systems; Requirements, test, defect, and release management systems
Main activities of the life-cycle	Setting business objectives; Evaluating service capabilities; Forming an eSourcing team; Evaluating potential eSourcing benefits	Defining and approving critical assessment criteria; Analyzing political, social, competitive, and legal environments of the markets and/or countries; Identifying and selecting prime potential markets and countries; Generating an initial list of prospective clients	Selecting prospective clients; Analyzing the profitability of the request for proposal (RFP); Clarifying the advantages and disadvantages of bidding on a contract; Preparing a service plan to a bidders' conference	Clarifying legal and commercial terms, conditions, and property rights; Specifying the service delivery process and baseline for the scope of delivery; Clarifying all the requirements in RFP; Signing the contract(s)
Preconditions and performance measures	Clarity of methods for evaluating service capabilities; Ability to formulate business objectives, organize the eSourcing team, and estimate benefits from economic and capability development viewpoints; Adequacy and validity of market and internal	The review and approval of the selection criteria by critical stakeholders; The number of investigated markets and/or countries; The number of leveraged eSourcing service opportunity channels; The number of prospective clients	The documentation, review, and approval of the advantages and disadvantages of bidding on a contract by the stakeholders; Correctness and comprehensiveness of the service plan to meet the requirements specified in RFP; Diligence of activities to prove the service capabilities	Agreement on the RFP and client requirements and legal and commercial terms, conditions, and property rights of the contract; Ability to specify a service plan for the service delivery; Openness and collaborative effort during contract negotiations
Expected outcomes	A set of defined strategic objectives; Evaluated service capabilities; An experienced eSourcing team; Adequate knowledge of competitiveness and profitability	The assessment criteria have been established, reviewed, and approved by the stakeholders; An extensive overview of the prospective markets and/or countries; The initial list of prospective clients	Identified advantages and disadvantages of bidding concerning each RFP; RFP analyzed from the viewpoint of profitability; Extensively documented and justified service capabilities in the service plan	Mutual agreement on contract specifications (e.g., scope, price, breakdown recovery processes, schedule, resource requirements, security provisions, and intellectual property rights); A signed contract; A win-win situation for the negotiating parties; Close, long-term relationship

The eSourcing life-cycle	Service delivery	Managing the eSourcing relationship and services	Contract termination and evaluation
Supporting classes of information systems	Customer relationship management systems; Extranet; Project estimation and measurement systems; Training and simulation systems; Requirements, test, defect, and release management systems	Customer relationship management systems; Extranet; Process asset library; Project estimation and measurement systems; Requirements, test, defect, and release management systems	Customer relationship management systems; Extranet; Process asset library; Project estimation and measurement systems; Requirements, test, defect, and release management systems
Main activities of the life-cycle	Building a joint implementation team; Training the relevant employees of both parties; Refining the service plan; Setting up the infrastructure; Executing service delivery; Reviewing service milestones together; Reporting service delivery progress regularly	Ensuring open sharing of knowledge; Instituting performance metrics, monitoring mechanisms, and incentive, corrective action, and penalty systems; Monitoring performance; Revising the contract if necessary; Executing incentives and corrective actions	Transferring resources to clients or third parties; Refining assessment criteria; Evaluating services and the relationship; Receiving payments; Analyzing the service results with respect to the baseline set in the contract; Documenting the lessons learned
Preconditions and performance measures	Completion of training and service deliverables, including progress reports, in accordance with the established service plan and process objectives; Completion of milestone reviews in accordance with the planned schedule	Effectiveness of the created management structure, knowledge sharing scheme, and recovery from service breakdowns; Institutionalization of metrics, mechanisms, and incentive, corrective action, and penalty systems according to the contract and service plan; Compatibility of service processes and environments	The performance of the final deliverable and the delivery time and total cost of the project versus the contract; Client and user satisfaction; The extent of capability improvement; Relationship quality; Ability to assess alternative sourcing arrangements
Expected outcomes	Effective implementation team; Revised and detailed service plan; Infrastructure to implement the plan; Defined performance metrics; Skilled and motivated employees; Timely and technically high quality progress reports and deliverables; Ability to deal with breakdowns effectively	Effective management structure, sharing of knowledge, and recovery from breakdowns; Enacted performance metrics and monitoring mechanisms; Comprehensive incentive, corrective action, and penalty systems; Services meet the RFP on time and in budget	Fully satisfied clients; Services completed and relationships formed in line with the business objectives, the strategic plan and client requirements; Improved service provider capabilities; On-time delivery within the budget

4 THE CLASS OF INFORMATION SYSTEMS FOR ESOURCING SERVICE PROVIDERS

This chapter generalizes the most important class of information systems for eSourcing service providers. It is based on the two eSourcing sub-domains research and analyzes the common and variable aspects of the eSourcing life-cycle among each eSourcing domain, literature reviews and the best practices framework for eSourcing service providers. The thesis argues that the most important class of information systems for eSourcing service providers is the class of eSourcing requirements, delivery and completion management systems. Sections 4.1 and 4.2 present the meta-requirements and meta-design of the class of eSourcing requirements, delivery and completion management systems, section 4.3 is the validation for the class of eSourcing requirements, delivery and completion management systems. The last part is the comparison of the class of eSourcing requirements, delivery and completion management systems with kernel theory eSCM-SP and information systems design theory for Dual Information Systems (DIS).

4.1 Meta-requirements of the design theory for the class of eSourcing requirements, delivery and completion management systems

This section generalizes the meta-requirements of information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMA. Based on comparing the common and variable aspects of the two sets of meta-requirements, this research generalizes the most important business practices for the providers of eSourcing services and meta-requirements for the class of eSourcing requirements, delivery and completion management systems.

4.1.1 Meta-requirements of the design product theory for the class of eSourcing requirements and delivery management systems

The class of eSourcing requirements, delivery and completion management systems will offer three categories of services: (1) requirements management, (2) delivery management and (3) completion management (Table 6). Requirements management is responsible for, requirements such as prioritization and management. Prioritization relates to establishing priorities for requirements based on client requirements and business risks (e.g., lowest cost, most effective). Requirements management is responsible for a variety of issues such as transforming client requirements to specific executable requirements, arranging proper resources for realizing the requirements, and enabling clients to monitor and track delivery service progress. If clients change the requirements or there are unexpected breakdowns due to, for example, unavailability of critical service components, requirements management should adjust and re-prioritize the requirements as necessary. Delivery management is responsible for delivering the service according to the agreed-upon engagement, identifying and tracing breakdowns during the services, and reporting the breakdowns and their impacts. Completion management is responsible for transferring the resources to clients or third parties, and to record lessons learnt for improving future services.

TABLE 6 A framework for categorizing the services of the design product theory for the class of eSourcing requirements, delivery and completion management systems

Requirements management	Delivery management	Completion management
1 Based on business priorities and risks, prioritize a requirement 2 To reduce duplication, collect requirements from previous similar services 3 Manage interdependencies between requirements and align requirement, delivery, and related resources 4 Provide clients with requirements status	1 Monitor delivery progress against the service plan 2 Report execution results and delivery breakdowns 3 Collect metrics and identify and track breakdowns during the services 4 Generate a delivery results report	1 Transfer the resources to clients or third parties 2 Record lessons learnt for future services

In the initiation phase, both testing service providers and logistics service providers need to analyze the request for proposal (RFP) from clients and the documented requirements for the service. They need to analyze RFP and requirements to create a business case for estimating the profitability of the service. If the service is profitable, the providers will draft service plans to bid.

Clients will analyze the plans, choose the proper service providers, sign the contracts and transfer necessary resources to service providers. Service providers will arrange proper employees for the service and offer training to develop the necessary skills for them. Therefore, in the initiation phase, both testing service providers and logistics service providers need to analyze clients' requirements and transform clients' requirements to executable requirements. They prepare resources and arrange proper employees for the service.

Requirements analysis and management are the main activities through the eSourcing life-cycle. Both information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS support requirements prioritization and management during the sourcing life-cycle. The requirement database enables information system design product theory for the class of eSourcing requirements, delivery and completion management systems to manage prioritized requirements and provide real-time requirements information and associated delivery and breakdown information to evaluate the service quality and progress. Furthermore, bidirectional traceability is supported between requirements and delivery information artifacts across the eSourcing life-cycle.

In the delivery phase, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports track and record delivery information to the database. All the business practices are used to deliver services to meet client requirements and the request for proposal (RFP). For example, information system design product theory for the class of RTDMS instances support testing service providers to execute testing services and defect management, and information system design product theory for the class of OTWMS instances support logistics service providers to deliver cargo and offer warehouse management services. Therefore, delivery teams are responsible for executing specific delivery services in the eSourcing life-cycle. Clients and service providers can track the delivery process and obtain the relevant information in a timely manner through information system design product theory for the class of eSourcing requirements, delivery and completion management systems. The delivery processes include various stakeholders. The execution of the delivery services involves the coordination of the flow of information, services, knowledge transfer and related activities among the stakeholders. Therefore, it is important to manage the life-cycle effectively to meet the delivery performance expectations of the stakeholders.

Instances of information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS support effective communication and get timely information on the services for clients and service providers, which makes the service process transparent and seamless. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems provides support for clients and service providers to communicate effectively and seamlessly throughout the delivery phase. For example, the information about the requirements that have been met can be shared with stakeholders to adjust resource allocation. In addition, information system design product theory for the class of eSourcing requirements, delivery and completion management systems enables clients and service providers to deal with breakdowns quickly. For example, whenever a service is delayed, clients and service providers need to find out the reasons for the delay and inform each other to come up with possible solutions. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems records all the breakdowns and solutions to help both clients and service providers to improve their performance.

Whenever service breakdowns happen, information system design product theory for the class of eSourcing requirements, delivery and completion management systems will send relevant information to the stakeholders who need to communicate with clients and adjust the service plans as necessary. Clients will estimate the impacts of breakdowns and decide whether to change their requirements or service plan. If they change the requirements, they have to adjust the service plan and negotiate with service providers as necessary.

The completion phase starts once clients have received the services that they need. Service providers can then prepare for transferring the services to the clients or third parties. Clients need to check the services to determine whether they meet the service closure conditions. If the conditions are met, the engagement between the client and the service provider can be closed. The client needs to pay for the services according to the original agreement and the realized service quality (e.g., on time and within the budget). When the service provider's financial department receives the payment, the service provider can close the engagement, summarize the lessons learnt, and compare the performance during the engagement with earlier measurement results to improve their service capabilities. For example, information system design product theory for the class of eSourcing requirements, delivery and completion management systems should be able to benchmark the performance with domain-specific industry standards and previous performances and report the results to delivery teams and other stakeholders.

Both information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS store relevant artifacts to the database for future reuse in the completion phase. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems stores relevant information artifacts to the database for future reuse. The artifacts to be stored include requirements, schedules, service plans, and metrics, helping service providers to improve their delivery management and performance. The change management part is optional in both information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems has also the change management part to deal with clients' change requirements. Therefore, it summarizes and generalizes the common parts of information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS.

4.2 Meta-design of the design product theory for the class of eSourcing requirements, delivery and completion management systems

Requirements, Delivery and completion artifacts are managed by the information system design product theory for the class of eSourcing requirements, delivery and completion management systems. Requirements are associated with the Delivery and Completion management artifacts. The following parts will explain the relationships between these artifacts. Requirements are based on client requirements and RFPs. Each requirement needs at least one delivery service to complete it. Each delivery service links with at least one requirement.

Based on Lu and Käkölä (2014), this section introduces generic structures and attributes of the class of requirements management, delivery management and completion management artifacts presented above. According to the design product theory, the class of eSourcing requirements, delivery and completion management systems should include at least these structures and attributes to be effective.

4.2.1 Information model for the eSourcing Requirements, delivery and completion management process

In order to design an effective eSourcing requirements, delivery and completion management process, the thesis defines the information model for the process. Figure 2 presents a simple and generic model based on the literature review (Rottman and Lacity, 2004; Käkölä, 2008; eSCM-SP, 2010; Plugge, Bouwman & Molina-Castillo, 2013) and on a detailed examination of the information management process from the case companies.

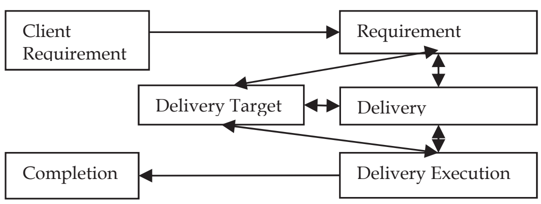

FIGURE 2 Information model of the meta-design of the design product theory for the class of eSourcing requirements, delivery and completion management systems

It consists of six classes of artifacts used in the process: Client Requirement, Requirement, Delivery, Delivery Target, Delivery Execution, and Completion. Each artifact is related to at least one artifact to support traceability and hierarchical management during the life-cycle (e.g., each artifact has a number of versions as the service progresses).

The Client Requirement artifacts are used for documenting specific requirements from clients and the external environment in general. Internal and external requirements are separated to meet the meta-requirements for the eSourcing services and information security. For example, Client Requirement maybe focuses on the strategic importance of business requirements for competitive advantage. One Client Requirement can be met by several Requirements. Therefore, Client Requirements are high-level and abstract requirements. Access rights for Client Requirements and Requirements should be clearly defined before the service initiation to deal with information security concerns.

The Requirement artifacts deal with the internal requirements developed by service providers to realize Client Requirements. Separating Client Requirements and Requirements can facilitate change management during the service process. Client Requirement artifacts can be changed when service has breakdowns. Clients need to negotiate with service providers before they change the service plans in ways that necessitate changes in Requirements. Service providers have to estimate the risks associated with the changes and analyze the impacted Requirement artifacts. If the impacted Requirement artifacts need to be changed and significant additional workload results, service providers need to renegotiate the contracts with clients.

Delivery artifacts respond to and meet the needs specified in one or more Requirement artifacts. For example, each Requirement artifact has a respective delivery plan and delivery results. Delivery plans and delivery results are associated with, respectively, specific delivery schedules and detailed delivery reports. Delivery artifacts are the detailed inputs for and outputs of Delivery Execution. Each delivery execution implements the associated delivery plan and is documented by a Delivery Execution artifact.

Delivery Target artifacts document the artifacts (e.g., system and software components) to be delivered in the services. As the service progresses, each version of the deliveries artifacts is recorded by Delivery Target artifacts. Each Delivery Target artifact is associated with Requirement, Delivery and Delivery Execution artifacts.

Delivery executions, recorded by Delivery Execution artifacts, validate whether the delivery target meets the associated requirements. They can be created and managed as hierarchical structures (e.g., component delivery, and system delivery). The delivery execution processes may reflect business processes or product architectures. Delivery Execution is the largest artifact in the information model containing implementation, workflow, validation and verification.

Delivery Execution artifacts are associated with Completion, Delivery Target and Delivery artifacts. For each newly-found service breakdown, service providers need to create a respective Completion artifact and document its associations with Delivery Execution artifacts for traceability. After clients and service providers have repaired the service breakdown, service providers start a new delivery cycle. They need to ensure the service breakdown won't reoccur.

Completion artifacts document the delivery results and record service breakdowns and solutions.

Client Requirements are associated with one or more Requirements, which, in turn, are linked to one or more Delivery artifacts. Abstract Client Requirements typically represent business problems. Refining Client Requirements to detailed enough Requirements facilitates process management and better estimation of project costs, schedules, and required resources. Delivery artifacts and Delivery Execution artifacts describe implementable solutions to the associated Requirement artifacts. Delivery Target artifacts describe the versions of executed Delivery targets. Completion artifacts are linked to Delivery Execution artifacts.

4.2.2 Requirements

This section describes the meta-requirements of the design theory for the class of eSourcing requirements, delivery and completion management systems. Figure 3 visualizes how the meta-design covers the eSourcing life-cycle. Table 7 presents the generic structure of requirements artifacts. In the following, each class within the structure is explained.

TABLE 7 Generic Structure of Requirements Artifact

Class	Questions	Attributes
Description	What is the requirement about?	Name, ID, Description, Required date and time of delivery, Rationale
Origin	Which client requirements does the requirement refer to?	Author, Source, Date of creation
Analysis	What are the implications of the requirement?	Status, Required effort, Priority, Scheduled date and time of delivery
Workflow	What should be done to this requirement next? By whom?	Assigned Delivery services, Responsible person, Realized requirement closure date and time
History	What has been done to the requirement artifact? When?	Information about all prior edits, editors, and changes

Description describes what a requirement is about, the purpose of the requirement, and the schedule for the requirement delivery. If there are service breakdowns that lead to change requirements, clients may send a new requirement and providers need to renew the service plans to execute the services. Name and ID are used for identification and traceability.

Origin describes the client requirements the Requirement is based on. One client requirement may be transformed to several executable Requirements.

Analysis is used to probe the implications of the Requirement. Priority is used to rank Requirements and arrange suitable resources and efforts. During the service delivery phase, status can be used to check the Requirements status (e.g., delivered or not).

Workflow describes what should be done next to this Requirement and by whom. Requirements management teams need to allocate each Requirement to one or more Delivery management services.

History is used to provide information about the responsible managers and all prior edits of Requirement attributes. As a result, the stakeholders can be held accountable for their actions and unexpected service breakdowns can be dealt with effectively. Changed Requirements may necessitate unexpected revisions of services plans and raise service risks. History information helps service providers to proactively eliminate many breakdowns and to recover from breakdowns.

4.2.3 Delivery

Table 8 presents the generic structure of delivery artifacts.

Description describes the purpose of an executed Delivery. Process indicates the processes needed for the Delivery service. These processes include the expected results and any service breakdowns.

Origin describes the Requirement(s) the Delivery service refers to. One Requirement may need more than one Delivery service.

Analysis is used to probe the implications of a Delivery service. If the Delivery has any breakdowns, it includes the reasons and impacts of the breakdowns and effects of solutions. This information can also be reused to help service providers and clients optimize service plans and improve service effectiveness in future. Priority describes the priority of the Delivery service, and status refers to the Delivery progress (e.g., routine, pause by breakdown or repairing breakdown). Required effort describes the Delivery costs, time, and resources, which can be used to calculate the total service cost of an engagement. This information can be reused to estimate the profitability and feasibility of future engagements.

Delivery methods provide traceability links to the tools and methods that are used in the Delivery services.

Workflow describes who is responsible for Delivery processes and the realized time of Delivery. If it has a breakdown, it includes the processes that should be taken to provide the Delivery and the responsible stakeholders. The delivery team needs to communicate with the requirement team and form a reasonable solution for breakdowns.

TABLE 8 Generic Structure of Delivery Artifact

Class	Questions	Attributes
Description	What is the delivery service about?	Name, ID, Description, Rationale, Process, Required date and time of delivery
Origin	Which requirement does the delivery service refer to?	Author, Source requirement, Date of execution
Analysis	What are the implications of the delivery service?	Status, Required effort, Risk, Priority, Scheduled date and time of delivery
Delivery methods	Which tools and delivery methods are involved in this delivery service?	IDs of tools to be used
Workflow	What should be done to this delivery service next? By whom?	Allocation to delivery team members, Responsible person, Realized date and time of delivery
History	What has been done to this delivery artifact? When?	Information about all prior edits, editors, and changes

4.2.4 Completion

Table 9 presents the generic structure of completion artifacts.

Description describes the purpose of the Completion. Process indicates the processes needed for the Completion service and includes transition resources to clients or third parties, summarizing the knowledge learned from the services.

Origin describes the Requirement(s) and the Delivery service refers to. One Completion may associate more than one Requirement and to a Delivery service.

Analysis is used to probe the implications of a Completion service. Service providers need to ensure all the Requirements have been met, and services have been delivered based on the contract. Priority describes the priority of the Completion service, and status refers to the Completion progress (e.g., transition resources, payments received). Required effort describes the transition resources to clients or third parties, summarizing the knowledge learned from the services. That knowledge can be used to improve service capabilities and the feasibility of future engagements.

Workflow describes who is responsible for Completion processes and the realized time of Completion.

History is used to provide information about the completion services. As a result, the stakeholders can check service quality, summarized knowledge, reused artifacts from the database, and information related to service quality. History information helps service providers to estimate service quality and to enhance service capabilities.

TABLE 9 Generic Structure of Completion Artifact

Class	Questions	Attributes
Description	What is the completion service about?	Name, ID, Description, Rationale, Process, Required date and time of completion
Origin	Which requirement and delivery does the completion service refer to?	Author, Source requirement, Date of execution
Analysis	What are the implications of the completion service?	Status, Required effort, Risk, Priority, Scheduled date and time of completion
Workflow	What should be done to this completion service next? By whom?	Allocation to delivery team members, Responsible person, Realized date and time of delivery
History	What has been done to this completion artifact? When?	Information about all prior edits, editors, and changes

4.3 Validating and scoping the design product theory for the class of eSourcing requirements, delivery and completion management systems

This section validates the design product theory for the class of eSourcing requirements, delivery and completion management systems and scope it.

4.3.1 Validating and scoping the design product theory for the class of eSourcing requirements, delivery and completion management systems

According to the representatives of the case companies, information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS incorporate the designs of the eSourcing life-cycle processes and information systems that have helped the companies to transcend the limitations of the mediated eSourcing business model in the respective domain (Lu & Käkölä, 2014). As a result, they have gained more profitable contracts and established direct communication with the end clients. The associated databases support each information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS instance, respectively, and accumulate the knowledge and lessons learnt from the services, helping the case companies enhance their service capabilities.

Specifically, an information system design product theory for the class of RTDMS instance aligns test requirements and related test cases and defects, which helps test teams to monitor service progress and to locate defects efficiently in the service. The information system design product theory for the class of RTDMS instance sends real time defects information to testers and clients, so that all the involved stakeholders can communicate with each other as necessary, thus making the testing service process and corporation transparent and seamless. Before Ltesting used an information system design product theory for the class of RTDMS instance, test teams needed to collect the defects and send them to clients in regular batches (e.g., each hour), creating unnecessary delays. Clients had to repair the defects without sufficient background information. For example, clients did not necessarily know the relationships of the defects to test requirements, the times when test teams had run particular test cases, and the order of test case execution. Therefore, the availability of defect information and aligned requirements and test cases reduce the costs incurred by service providers and clients during the eSourcing life-cycle.

The associated databases have offered the case company defect information from previous similar projects and helped clients to repair defects quickly, which has impelled clients to outsource larger and longer-term contracts to Ltesting. Ltesting gains more profitable contracts and establishes strategic relationships with clients. For example, in 2009, the case company gained some important clients in the Chinese market (e.g., CICC) and got a long term contract from a French client, and the tested systems for Irish customs and tax systems.

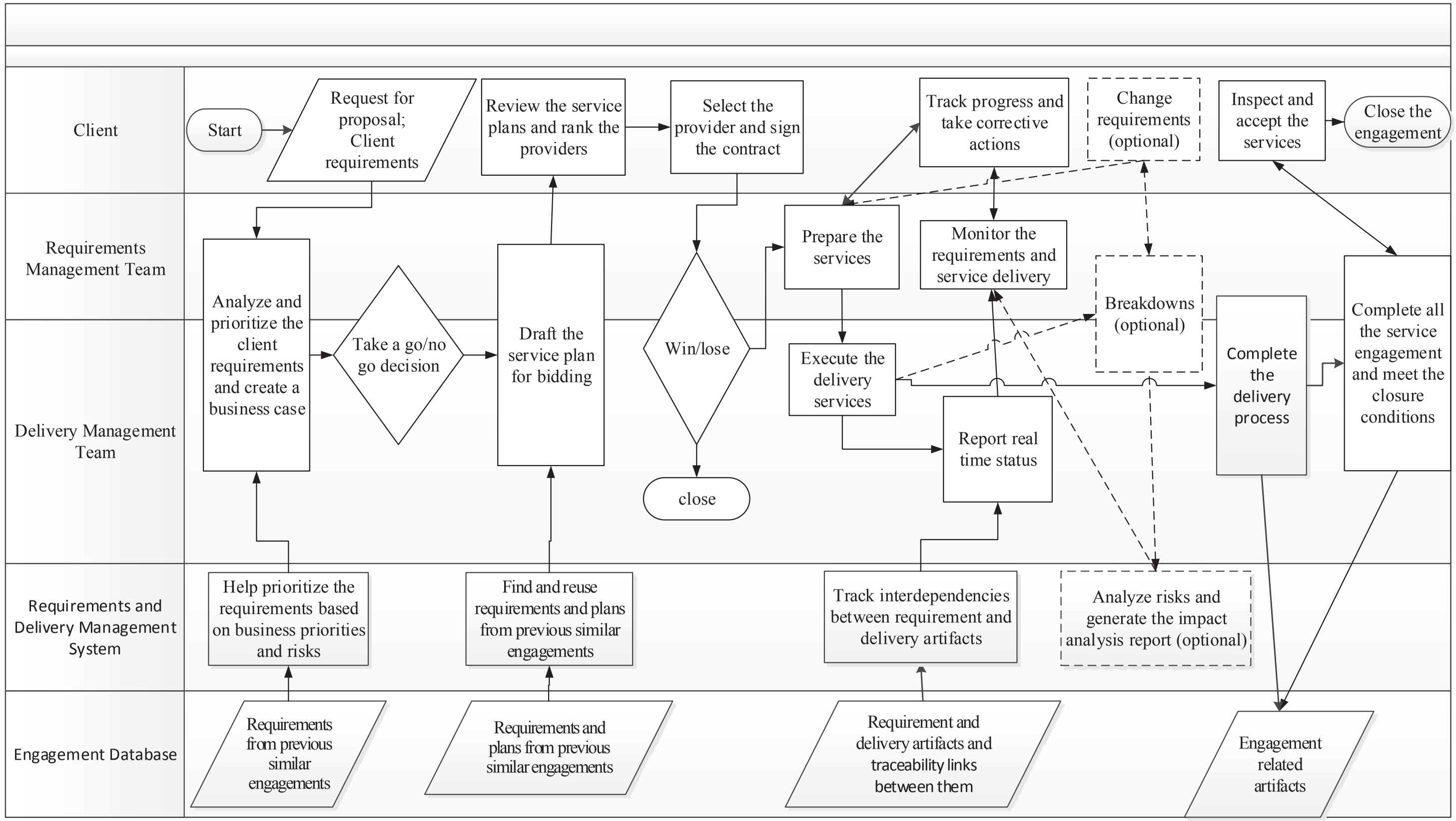

FIGURE 3 A meta-design for the class of eSourcing Requirements, Delivery and Completion Management Systems supporting the eSourcing life-cycle

The HP Quality Center (QC), a commercially available integrated requirements, test and defect management product, served as an instance of information system design product theory for the class of RTDMS in the case company. It required clear access rights for the different roles involved, causing some trouble for Ltesting in the early stages of its adoption of information system design product theory for the class of RTDMS. Depending on the project characteristics and the financial pressure, sometimes test analysts and test managers have to do testers' work. Information system design product theory for the class of RTDMS has impelled Ltesting to improve its organizational structure and, specifically, service team structure. They have established independent measure management teams, quality assurance teams and configuration management teams, facilitating the provisioning of more comprehensive and professional services.

"*All the test projects use QC or other similar test platforms, but the other platforms mainly focus on test management or defect management. QC is the best because it can also offer complete requirements management and defect management services. The specific test tools can run on this platform, but they can also run on other platforms, so based on QC we can better control the entire service life-cycle.* "-CEO

"*QC is a good instance of RTDMS. We rely on it to improve our performance, which helps us to improve communication and service effectiveness. Clients are satisfied with our service quality improvement and want to give better contracts to us, which means larger and more profitable projects.*" –CEO

The case company always tries to get permission from clients to use QC to manage the services, because it can cover the entire life-cycle and offer comprehensive test management, which is convenient for both service providers and clients. Isolated management tools (e.g., requirements management or defect management tools) cannot offer comprehensive testing environment and context information (e.g., related test cases and requirements) for clients and service providers. This would impact on service progress.

"For example, isolated defect management tools cannot offer comprehensive test environment and background information for clients, which is not helpful in repairing defects and improving service effectiveness. Sometimes clients have to ask this information to repair defects. If requirements management tools do not link with relevant test cases or defect, it is also difficult for us to estimate the service progress and service quality. When the information from hundreds of previous projects has been examined and compared after we have used QC in our services, the service quality has been seen to improve a lot. Most of other testing management tools focus on defect management, they do not have enough support for requirement management and test management." – CEO

PGL has developed an information system design product theory for the class of OTWMS instance for executing effective logistics services. When an order is generated, information system design product theory for the class of OTWMS instance provides the transportation team and the warehouse management team with the order information in real time, so they can prepare for service delivery proactively. Clients can get all the information they need whenever they need it, based on the order number. For example, they can easily track cargo information. Information system design product theory for the class of OTWMS has thus helped PGL to gain clients' trust and more contracts that are profitable.

"*Without the help of OTWMS, it would be impossible to deal with breakdowns in one hour. This system enables us to communicate with all stakeholders, including clients in real time, which is crucial to deal with breakdowns effectively.*" - An order manager

"*The timely information enhances our transportation and warehouse management effectiveness, helping us to have a better performance in the service life-cycle. OTWMS helps us to make the life-cycle transparent and seamless, so we can earn clients' trust and establish strategic relationships with them.*" –CIO

The information system design product theory for RTDMS and information system design product theory for the class of OTWMS contribute in their respective domains and thus help the case companies to transcend the limitations imposed by the mediated sourcing business model. The class of eSourcing requirements, delivery and completion management systems is based on and partly validated by information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS, so it can be expected to facilitate small and medium-sized eSourcing service providers in overcoming the limitations imposed by the mediated sourcing business model.

4.3.2 Evaluating the quality of the design product theory for the class of eSourcing requirements, delivery and completion management systems

There are multiple views of theory in information systems research. Gregor (2006) identifies five interrelated types of theory: (1) theories for analyzing, (2) theories for explaining, (3) theories for predicting, (4) theories for explaining and predicting, and (5) theories for design and action. Weber (2012) establishes strict criteria for theories and ascribes the term "theory" only to theories for explaining and predicting. The criteria to evaluate the quality of theories extend across the perspectives of importance, novelty, parsimony, level and falsifiability. The design product theory for the class of eSourcing requirements, delivery and completion management systems is a type 5 theory for design and action (Gregor, 2006), making the design and implementation process more tractable for developers and sub-domains (c.f., Markus, 2002). The following part (Table 10) evaluates the design product theory for the class of eSourcing requirements, delivery and completion management systems based on the criteria established by Weber (2012).

TABLE 10 Summary evaluation of Information system design product theory for the class of eSourcing requirements, delivery and completion management systems

Importance	The created information system design product theory for the class of eSourcing requirements, delivery and completion management systems that was created offers as generalizable scientific knowledge as possible about the most important business practices for eSourcing service providers from the viewpoint of service provisioning, breakdown recovery, and the redesign of the eSourcing life-cycle. eSourcing service providers can use it to establish domain-specific design product theories and to instantiate them into information systems that support the design, service provisioning, and breakdown recovery within the eSourcing life-cycle.
Novelty	The meta-requirements and the meta-design of the information system design product theory for the class of eSourcing requirements, delivery and completion management systems is well organized to meet the needs of service providers but also helpful for clients for implementing eSourcing life-cycle management and estimating service providers' capabilities. To our knowledge, the extanting literature provides little theoretical guidance for managing requirements, delivery, and completion knowledge during the life-cycle.
Parsimony	The design product theory for the class of eSourcing requirements, delivery and completion management systems has been designed to be abstract and generic enough to allow services providers use it to improve their processes and information systems regardless of their current practices and systems. It may be possible for the providers to benefit from the theory even without replacing any existing systems. Service providers can thus use even separate requirements management systems, delivery management systems, and completion management systems while using the theory to better integrate and organize these systems for enabling the end-to-end life-cycle. Therefore, it is parsimonious.
Level	The theory is framed at a broad level as a broad-range theory. The range of phenomena theory covers is the entire eSourcing life-cycle.
Falsifiability	The research validates information system design product theory for the class of eSourcing requirements, delivery and completion management systems through validating the information system design product theories for the class of RTDMS and the class of OTWMS, which specialize the abstract class for the subclasses in their specific domains. This research is limited by the single Chinese eSourcing context and the single case study in specific domains.

4.4 Comparison of the information system design product theory for the class of eSourcing requirements, delivery and completion management systems with the kernel theories

This section compares the information system design product theory for the class of eSourcing requirements, delivery and completion management systems with the kernel theories to clear the contributions of this doctoral thesis. eSourcing Capability Model for Service Providers (eSCM-SP) and Information Systems Design Theory (ISDT) for Dual Information Systems (DIS) are the kernel theories of this research. With the help of comparison, this section presents the contributions and limitation of this doctoral thesis.

4.4.1 Comparison of the information system design product theory for the class of eSourcing requirements, delivery and completion management systems with eSCM-SP

This research uses eSCM-SP V2, the latest version of the eSCM-SP model, as the reference model. The eSCM-SP life-cycle involves three phases. (1) Initiating an engagement: gathering and negotiating requirements with a client, contracting, and designing, resourcing, and deploying the service. (2) Service is delivered according to the commitments established for the engagement. (3) The engagement is completed primarily by transitioning the resources from the service provider to the client or to a third party (eSCM-SP, 2010). Ongoing practices are carried out throughout the eSourcing life-cycle to perform management functions. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, performance management), and there are specific practices in each phase. eSCM-SP does not include the information systems to support involved stakeholders practices during the eSourcing life-cycle, but information system design product for the class of eSourcing requirements, delivery and completion management systems offers support for the entire eSourcing life-cycle from initiation through delivery to completion phase, partly supporting ongoing practices.

In the initiation phase, a client provides a request for proposal (RFP) and documented requirements for the project. The service providers need to analyze the proposal and requirements to create a business case for estimating the profitability of the service. If the service is profitable, the service providers will draft service plans to bid. Clients will analyze the plans, choose the proper service providers, sign the contracts and transfer necessary resources to the service providers. The service providers will arrange proper employees for the service and offer training to develop the necessary skills.

Requirements analysis and management are the main activities through the eSourcing life-cycle. The requirement database enables information system design product theory for the class of eSourcing requirements, delivery and completion management systems to manage prioritized requirements and provide real-time requirements information and associated delivery and breakdown information to evaluate the service quality and progress (Lu & Käkölä, 2014). The information system design product theory for the class of eSourcing requirements, delivery and completion management systems collects previous relevant artifacts from each engagement between a service provider and a client. Service providers analyze the client requirements and search for relevant artifacts to reduce time and costs involved in creating a suitable service plan and to estimate the service price and the required time and resources based on previous experiences. After that, they draft the service plan and bid. Furthermore, bidirectional traceability is supported between requirements and delivery artifacts across the eSourcing life-cycle.

In the delivery phase, delivery teams are responsible for executing specific delivery services. Clients and service providers can track the delivery process and obtain the relevant information in a timely manner through information system design product theory for the class of eSourcing requirements, delivery and completion management systems (Lu & Käkölä, 2014). The process includes various stakeholders. The execution of the process involves the coordination of the flow of information, services, and related activities among the stakeholders. Therefore, it is important to manage the life-cycle effectively to meet the delivery performance expectations of the stakeholders.

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems is expected to support the entire eSouricng life-cycle for sharing information, thus, enabling clients and service providers to communicate effectively throughout the delivery phase. For example, the information about what requirements have been met can be shared with stakeholders to adjust resource allocation. In addition, information system design product theory for the class of eSourcing requirements, delivery and completion management systems enables clients and service providers to deal with breakdowns quickly. For example, whenever a service is delayed, clients and service providers need to find out the reason for the delay and inform each other for reaching possible solutions. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems records all the breakdowns to help both clients and service providers to improve their performance.

The completion phase starts when clients have received the services they need. Service providers can then prepare for transferring the services to the clients or third parties. Clients need to check the services to determine whether they meet the service closure conditions. If the conditions are met, the engagement between the client and the service provider can be closed. The client needs to pay for the services according to the original agreement and the realized service quality (e.g., on time and within the budget). Once the service provider’s financial department has received the payment, the service provider can close the engagement, summarize the lessons learnt, and compare the performance during the engagement with earlier measurement results to improve their service capabilities. For example, information system design product theory for the class of eSourcing requirements, delivery and completion management systems should be able to benchmark the performance with domain-specific industry standards and previous performances and report the results to delivery teams and other stakeholders (Lu & Käkölä, 2014).

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems stores relevant artifacts to the database for future reuse. The artifacts to be stored include requirements, schedules, service plans and metrics, helping service providers to improve their delivery management and performance.

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports the entire eSourcing life-cycle but it cannot support high levels of eSCM-SP practices, because most Chinese providers are in relatively early phases of eSourcing capability development (Oshri, Kotlarsky, Rottman, & Willcocks, 2009) and thus cannot use the most advanced practices of eSCM-SP. For example, the people management practice *ppl01,* Encourage Innovation, is used to encourage organization to establish and implement a policy to support innovation (eSCM-SP, 2010). Both clients and service providers can benefit from innovation because "process and service innovation can improve providers' performance and deliver the improved value to clients" (eSCM-SP, part 2, 2010, 50). Furthermore, the Encourage Innovation practice can create new business opportunities and improve client satisfaction. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems cannot offer support for this practice, but it supports change management and stores the responding solutions for the change management. The change management may be reasoned from clients change requirements, or both clients and service providers deal with service breakdowns. Therefore, the responding solutions for the change management can be used for redesigning the eSourcing life-cycle as necessary, proactive elimination of some breakdowns, and effective long-term enactment of routines (Käkölä and Taalas, 2008).

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems only supports part of the ongoing services of eSCM-SP. Ongoing practices include six capability areas: knowledge management, people management, performance management, relationship management, technology management, and threat management. Relationship management and technology management are missing from information system design product theory for the class of eSourcing requirements, delivery and completion management systems because it has been designed to be abstract and generic enough to allow eSourcing providers use it to design domain-specific information systems and improve their processes and information systems regardless of their current practices and systems. For example, information system design product theory for the class of eSourcing requirements, delivery and completion management systems can track the requirements execution process against the service plan and report execution results and breakdowns. It does not need to execute specific tasks, but it needs to track and report the results of the tasks. Specific tasks can be run by using other management tools. For example, the testing case company uses QC to manage the entire testing life-cycle, but sometimes they use RequisitePro as a separate requirements management tool in the services, because the functions of QC are not as good as the respective functions of the competing tools. Therefore, the analysis of the practices and information systems of the case companies has helped this research to scope the design product theory for information system design product theory for the class of eSourcing requirements, delivery and completion management systems appropriately.

4.4.2 Comparing information system design product theory for the class of eSourcing requirements, delivery and completion management systems with information systems design theory for Dual Information Systems

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems and Information Systems Design Theory (ISDT) for Dual Information Systems (DIS) are both ISDT for the abstract class of information system, because they are not supposed to have instances and build instances for practice.

The meta-design of information systems design theory for DIS has three layers: knowledge-base layer, business-system layer and project-system layer. The meta-design of information system design product theory for the class of eSourcing requirements, delivery and completion management systems covers entire eSourcing life-cycle and has corresponding practices in the three layers.

In the knowledge-base layer, information system design product theory for the class of eSourcing requirements, delivery and completion management systems stores current and historical artifacts to the database for future reuse. The artifacts to be stored include requirements, schedules, service plans, and metrics, which are used to help providers to improve their delivery management and performance in the services. The artifacts are stored in an abstracted form, because domain-independent artifacts are on the level of life-cycle and functional roles, not on the level of individual requirements or delivery. On the domain-specific level, for example, in information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS, the stored artifacts could enlarge to include specific and detailed information, such as specific service breakdowns, the reasons for breakdown and solutions (Lu & Käkölä, 2014). “The knowledge-base layer of information systems design theory for Dual Information Systems is a repository of explicit work and IS design knowledge in the knowledge-base layer of a hypertext organization." (Käkölä & Taalas, 2008, 4) Therefore, information systems design theory for DIS and information system design product theory for the class of eSourcing requirements, delivery and completion management systems have similar support for knowledge creation, storage, dissemination and reuse.

In the business-system layer, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports the end-to-end eSourcing services. Information related to client requirements is stored in the database. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports information sharing among stakeholders, therefore, clients and service providers and the internal service teams can communicate with each other (Lu & Käkölä, 2014). The resources are allocated based on the priority of requirements. Unexpected breakdowns are inevitably among routines for several reasons: (1) clients' change of requirements, (2) technological problem, and (3) poorly designed service plan and schedule. Breakdown management collects information on breakdowns and the reasons for them, and stores them in the database. Generally, the corresponding solutions for breakdowns should be approved by clients before using them in the services, and the solutions could also be reused in future services. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems stores the historical information on the services breakdowns and corresponding solutions in order to improve the service provider's performance and service effectiveness in the services. The service provider's performance is monitored during the services, and the collected metrics could help service providers to get close to the best practices and to improve the service quality (Lu & Käkölä, 2014). Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports flexible service processes during the life-cycle. For example, if clients change requirements and/or deal with serious breakdowns, the service providers need to redesign the service plan and adjust the responding resources allocation as necessary. In addition, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports relatively autonomous sub-teams using integrated information systems to coordinate and execute complex workflows and manage breakdowns in the workflows. "Actors on the business-system layer draw on services of the business-system layer of Dual Information Systems to learn, enact, and coordinate activities, to zoom in on the details of their work, and to deepen their understanding of the computerized aspects of work in order to handle unexpected (coordination) breakdowns." (Käkölä & Taalas, 2008, 4) Therefore, both information systems design theory for DIS and information system design product theory for the class of eSourcing requirements, delivery and completion management systems offer support in the business-system layer to execute routines and deal with breakdowns.

In the project-system layer, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports involved teams to deliver services, and partly supports service providers to access and redesign the eSourcing services plan with clients, because the redesign actions are initiated when serious breakdown happened and clients require a redesign of the delivery schedule or service plan. Therefore, information system design product theory for the class of eSourcing requirements, delivery and completion management systems could provide support to redesign life-cycle as the business-system layer supports as information systems design theory for DIS. Service providers' performance can be measured by industry metrics and, benchmarks, and their performance can be monitored during the sourcing life-cycle. The knowledge of the current and historic business practices should be available for performance and redesigning processes. This information is stored in the database. "Self-organizing project teams on the project-system layer use services of the project-system layer and the knowledge-base layer of information systems design theory for DIS to produce innovative work and IS (re)design that can be enacted on the business-system layer." (Käkölä & Taalas, 2008, 4) Information systems design theory for DIS supports redesign of work and information systems, and information system design product theory for the class of eSourcing requirements, delivery and completion management systems partly supports service providers and clients to redesign projects and replan the services when serious breakdowns have taked place or clients require a redesign of the delivery schedule or service plan. The solutions for breakdowns can be used for redesigning the eSourcing life-cycle as necessary, proactive elimination of some breakdowns, and effective long-term enactment of routines (Käkölä and Taalas, 2008).

Information systems design theory for DIS presents the layer of redesign, but it does not show how the involved roles offer support to each other to redesign the process. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems can support the involved teams' members to redesign the processes. eSourcing service providers perform requirement and delivery practices during the services. Benchmarking helps them to get the best practices and compare their performance with the metrics. Standardized reports and metrics could help them to measure their performance in the services. During the sourcing life-cycle, service teams' performance is monitored, and the metrics can be collected during the services. Knowledge-base layer stores the information about all the services-related details, and project-system layer could analyze the data before and during the services. Therefore, service providers could use the information referred to above and modeling services to improve their service process, and/or innovate new processes, and/or information system designs that leverage the lessons learnt from benchmarking and suit the new clients or new client requirements. If the new solution has been approved by clients, the generated process can be used in the services to improve service quality. If it is necessary to change the process in the business-system layer, the stakeholders should get the training materials and relevant documents. The materials can be constructed using the authoring service and the team members can only access the authored materials.

In the testing case research, the service providers did not have the support teams during the early period. The measurement management, configuration management and quality assurance management teams were missing from their early services, but their work was performed by the testing team. For example, testers were responsible for testing assignments and collecting metrics during the services. There are several potential risks if service providers lack measurement management, configuration management or quality assurance management in the services: (1) if the schedule is tight and time is limited in the services, testers might ignore the metric collection assignments, (2) as the service progresses, testers and test managers might not align configuration items as they should, and (3) testing team might not switch the roles between testing services and quality management in time to ensure testing service quality.

To improve their service quality and meet clients requirements, the case company used the reference models (e.g., CMMI, Rational Unified Process) to redesign their services team structure and added independent configuration, measurement and quality assurance management teams to the whole service teams. The information systems were also revised. For example, the authorities for different roles are clearer than before, and testers do not need to collect metrics information any more. Testing service providers are responsible for planning and delivering services, the sub-teams (e.g., measurement and quality assurance management teams) are able to use the integrated information systems to collect meaningful data about routines and breakdowns, data that is vital for improving and redesigning the routines. Therefore, the sub-teams could offer comprehensive and objective information to each other for redesigning the processes.

In the logistics case research, the case company redesigned the service processes from 2003 onwards because the original service couldn't meet client requirements (e.g., offer specific information on service delivery and service breakdowns). They developed TOM systems in order to offer specific information and meet clients' requirements. After that, all their departments' (e.g., warehouse management, transportation management) delivery services were based on the TOM systems. TOM is an instance of information system design product theory for the class of OTWMS, which has improved the service efficiency and reduced the service time.

Therefore, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports redesigning of information systems and work in the project-system layer. It helps stakeholders to attend the sourcing life-cycle in order to share knowledge of the service processes within and between service providers' sub-teams and clients, crystallize it into work systems and information systems (Alter 2006) that offer better services for clients, offer management for the whole life-cycle on a high level, create knowledge and share it through the knowledge-base and business-system layers to allow it to be applied effectively across organization boundaries. Furthermore, information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports redesign projects focusing on the periodical redesign of the service planning and execution processes within eSourcing, not on redesigning for the entire life-cycle.

4.4.3 Summary

This chapter generalizes the most important class of information systems for eSourcing service providers, which is based on two case studies and the best practices framework for eSourcing service providers. The chapter compares information system design product theory for the class of eSourcing requirements, delivery and completion management systems with eSCM-SP and information systems design theory for DIS to present the contributions of this doctoral thesis.

eSCM-SP does not include the information systems to support practices during the eSourcing life-cycle, but information system design product theory for the class of eSourcing requirements, delivery and completion management systems offers support for the entire eSourcing life-cycle. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports the entire eSourcing life-cycle for service providers, but it cannot support high levels practices of eSCM-SP because (1) most Chinese providers are in relatively early phases of eSourcing capability development and thus cannot use the most advanced practices of eSCM-SP, and (2) information system design product theory for the class of eSourcing requirements, delivery and completion management systems has been designed to be abstract and generic enough so that eSourcing providers can use it to design domain-specific information systems and improve their processes and information systems regardless of their current practices and systems.

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports breakdowns management, and it also partly supports redesign service process and information systems. Although it does not have the independent redesign project layer as information systems design theroy for DIS, it still is able to help service providers to redesign the service processes. Furthermore, information system design product theory for the class of eSourcing requirements, delivery and completion management systems could support information sharing among stakeholders, therefore, the stakeholders can communicate with each other and redesign the process and information systems as necessary.

5 SUMMARY

This section summarizes the dissertation and the research. The research contributions and limitations are described respectively. The last part is the future research.

5.1 Conclusions and contributions

China is in an important position in the global eSourcing market. This research focuses on the eSourcing service life-cycle, the supporting ICT tools in the life-cycle, and relevant business practices from service providers' perspective in the Chinese eSourcing service context. Most of China's eSourcing service providers are small or medium-sized and typically work for larger intermediaries instead of end-clients. This mediated eSourcing model restricts their business and capabilities development (Järvenpää & Mao, 2008). Järvenpää and Mao (2008) try to transcend the mediated eSourcing model limitation from the development of client-specific, process, and human resource capabilities, but their research does not cover the entire eSourcing life-cycle and cannot enhance providers' capabilities comprehensively. The existing literature does not extensively address this business model and ways to overcome its limitations.

To overcome this gap in the research, this dissertation presents the best practices for international eSourcing service providers and creates an information system design product theory for the class of eSourcing requirements, delivery and completion management systems to help eSourcing service providers to manage the entire eSourcing life-cycle and support their activities in the life-cycle. The best practices for eSourcing service providers describes each phase of the eSourcing life-cycle from the perspectives of main activities, supporting ICT tools, performance measures and expected outcomes. The literature provides little theory-based guidance to help companies to design and use information system design product theory for the class of eSourcing requirements, delivery and completion management systems to achieve the goals of cycle time reduction and improved product quality, service delivery, and overall effectiveness. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems works with the other supporting ICT tools to support eSourcing service providers in executing and delivering services in the eSourcing life-cycle.

This research studies the eSourcing in the ICT sourcing (ICTS) and business process sourcing (BPS) categories, respectively, based on specific domains, specifically, software testing services and logistics services. For the software testing services, the research probes reusable artifacts in Article I. A dynamic testing life-cycle model is created in Article II, and this article also probes the most important class of information systems for the services providers. Article III creates an Information System Design Product Theory for the class of Requirements, Test, and Defect Management Systems (RTDMS) to support providers reuse artifacts, manage the services and share information and knowledge with clients. Information system design product theory for the class of RTDMS helps clients and testing service providers to manage and control the testing process, enforce standardized service processes, and improve service effectiveness. Article III presents all the involved stakeholders and their performance in the testing life-cycle. The measurement management, configuration management and quality assurance management teams work with the test team to execute the testing services. For the logistics services, the research generalizes the most important class of information systems for logistics providers and creates an Information System Design Product Theory for the class of Order, Transportation and Warehouse Management Systems (OTWMS) in Article IV. Information system design product theory for the class of OTWMS helps clients and logistics service providers to manage the logistics process, execute standardized service processes, and improve service effectiveness. The cross-case research results can be found in Article V and Chapter 4. This research also creates an Information System Design Product Theory for the class of eSourcing requirements and delivery management systems (RDMS) in Article V. After that, it was found that the completion phase is also important in case of the information system design product theory for the class of eSourcing requirements and delivery management systems (RDMS). Therefore, RDMS name was changed to information system design product theory for the class of eSourcing requirements, delivery and completion management systems in this dissertation. eSourcing service providers can use it to establish domain-specific design product theories and to instantiate domain-specific design product theories into information systems that support the design of a solution that meets client needs, service delivery, and breakdown recovery within the eSourcing life-cycle. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems is used to create more detailed domain-specific design product theories. Indeed, information system design product theories for the class of OTWMS and the class of RTDMS are such domain-specific theories used to prescribe information system subclasses of the class prescribed by the theory for information system design product theory for the class of eSourcing requirements, delivery and completion management systems. It is expected that information system design product theory for the class of eSourcing requirements, delivery and completion management systems will help eSourcing service providers and commercial software vendors to design domain-specific integrated systems for service provisioning and breakdown recovery throughout the eSourcing life-cycle in a variety of ICT and business process sourcing domains, helping clients and service providers to manage and control the eSourcing life-cycle.

The eSourcing Capability Model for Service Providers (eSCM-SP) is a building block of this research because it is the most comprehensive eSourcing model available for service providers. eSCM-SP covers the entire eSourcing life-cycle from initiation and delivery to completion. Ongoing practices are run throughout the life-cycle to perform management functions. eSCM-SP is applicable to both ICT sourcing and business process sourcing and it can help service providers improve their capabilities related to both ongoing, phase-specific, and engagement-specific eSourcing practices throughout the eSourcing life-cycle. eSCM-SP helps the research to analyze eSourcing life-cycle from end-to-end. Information systems design theory for DIS is the other building block of this project. Information systems design theory for DIS has three layers: knowledge-base layer, business-system layer and project-system layer. Every layer has relevant practices. The theory helps the research to analyze eSourcing from different layer perspectives. Therefore, eSCM-SP and information systems design theory for DIS help in studying eSourcing services from multiple perspectives.

This research makes several contributions. The best practices for eSourcing service providers presents the main activities in each phase of the eSourcing life-cycle and help service providers to establish standardization service process and enhance service effectiveness. The information system design product theories for the class of RTDMS and the class of OTWMS contribute in their respective domains. The third main contribution is the information system design product theory for the class of eSourcing requirements, delivery and completion management systems. This theory is partly derived (1) deductively from two comprehensive kernel theories: eSCM-SP and information systems design theory for DIS and (2) inductively from the domain-specific information system design product theories for the class of RTDMS and the class of OTWMS. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems prescribes an abstract class of systems because instances of the class need not be built.

The research compares information system design product theory for the class of eSourcing requirements, delivery and completion management systems with eSCM-SP and information systems design theory for DIS in Chapter 4. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems covers the entire eSourcing life-cycle described in the eSCM-SP model, but it cannot support high levels practices in eSCM-SP because most Chinese providers are in relatively early phases of eSourcing capability development (Oshri, Kotlarsky, Rottman, & Willcocks, 2009) and thus cannot use the most advanced practices of eSCM-SP (e.g., the practices in level 4). Some eSCM-SP practices (e.g., encourage innovation, cultural fit) used for higher level of service capabilities cannot be supported by information system design product theory for the class of eSourcing requirements, delivery and completion management systems, because it is used to support eSourcing service providers who have mediated eSourcing model limitation and are in low levels of eSourcing service capability. In addition, information system design product theory for the class of eSourcing requirements, delivery and completion management system has been designed to be abstract and generic enough so that eSourcing providers can use it to design domain-specific information systems and improve their processes and information systems regardless of their current practices and systems. Thus Chinese providers can use eSCM-SP as a roadmap to improve their capabilities to higher levels. This comparison shows that information system design product theory for the class of eSourcing requirements, delivery and completion management systems could support entire eSourcing life-cycle, although it does not support the high level practices of eSCM-SP.

Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports relevant practices in the different layers described in information systems design theory for DIS. Information system design product theory for the class of eSourcing requirements, delivery and completion management systems supports breakdown management and stores the solutions for breakdowns, which helps in redesigning service process and information systems as necessary. Unlike information systems design theory for DIS, it does not have an independent redesign project layer, but it still can help providers to redesign the service processes. Furthermore, information system design product theory for the class of eSourcing requirements, delivery and completion management systems could support the team members' practices and help them to support each other in order to redesign the process and information systems as necessary.

The doctoral thesis validates information system design product theory for the class of eSourcing requirements, delivery and completion management systems through validating the information system design product theories for the class of RTDMS and the class of OTWMS that specialize the abstract class for the subclasses in their specific domains.

5.2 Limitation of the research

This research is limited by a single Chinese eSourcing context and a single case study in specific domains. The information system design product theory for the class of eSourcing requirements, delivery and completion management systems that was created is abstract and needs evaluation from more specific domains. However, Chinese eSourcing business is growing fast, and the chosen case companies are leading service providers in their service domain. Information system design product theory for the class of RTDMS and the dynamic testing life-cycle model have been shown and approved by the CEO and all the key managers in the testing case company, and all of them have more than ten years of testing experience. Information system design product theory for the class of OTWMS has been shown and approved by the CIO and key managers in the logistics case company, and they have more than ten years logistics service experience as well. Therefore, the conclusions are useful for other fast-growing eSourcing markets and eSourcing service providers who have similar limitations.

5.3 Future research

Future research needs to use the design product theory to build a set of other domain-specific design product theories and to track the theories back to the design product theory in order to further validate and revise it. Future research also needs to proactively redesign the business models and practices to ensure long-term effectiveness and reduce the likelihood that the encountered breakdowns would happen again. Process redesign and improvement are also important future research topics. Industries and service providers could create specific assessment models or service processes based on information system design product theory for the class of RTDMS and information system design product theory for the class of OTWMS. Even clients can reuse these two theories to assess the capabilities of service providers. For example, to access the quality of service providers' test artifacts in the database and estimate service providers' capabilities, clients might want to estimate how much test artifacts have been reused from the service providers' database.

This thesis uses domain engineering (Akoka, 2005, 461-463) and Alter's research on service system fundamentals (Alter, 2006, 2008) to generalize the common and variable aspects of the eSourcing life-cycle among the two different eSourcing domains. Change management is an optional part in information system design product for the class of RTDMS and information system design product for the class of OTWMS. But it is also an optional part in information system design product for the class of eSourcing requirements, delivery and completion management systems. Therefore, all the three design product theories include change management. The information system design product theory for the class of eSourcing requirements, delivery and completion management systems only generalizes the common parts of information system design product theories for the class of RTDMS and the class of OTWMS in this research. Future research need to generalize the variable parts to design domain-specific information systems and improve their processes and information systems.

Knowledge Process Sourcing (KPS) can be considered as “the outsourcing of firm activities that directly involve the production of knowledge and innovation, and that involve some degree of firm-specific capabilities.” (Mudambi & Tallman, 2010, 1436). Knowledge process sourcing (KPS) has higher requirements on inter-organizational information systems support to be successful. This is a new trend for eSourcing, and future research needs to validate the results of this dissertation in knowledge process sourcing contexts.

REFERENCES

Akoka, J. et al. (Eds.) 2005. Perspective in conceptual modeling, Tutorial 3: domain engineering-using domain concepts to guide software design. Springer-Verlag Berlin Heidelberg.

Alter, S. 2006. The work system method: connecting people, processes, and IT for business results. The work systems press, Larkspur, CA.

Alter, S. 2008. Service system fundamentals: work system, value chain, and life cycle. IBM Systems Journal, 47(1), 71-85.

Aron, R., Clemons, E. & Reddi, S. 2005. Just right outsourcing: understanding and managing risk. Journal of Management Information Systems, 22(2), 37-55.

Barthelemy, J. 2001. The Hidden Costs of IT Outsourcing. MIT Sloan Management Review, 42(3), 60-69.

Barthelemy, J. & Geyer, D. 2005 An empirical investigation of IT outsourcing versus quasi-outsourcing in France and Germany. Information & Management, 42(4), 533-542.

Bell, C.R. & Zemke, R.E. 1987. Service breakdown: the road to recovery. Management Review, 76(10), 32-35.

Borman, M. 2006. Applying multiple perspectives to the BPO decision: a case study of call centres in Australia. Journal of Information Technology, 21(2), 99-115.

Cachia, A. & Kruger, C.J. 2007. Guidelines for identifying risk vulnerabilities associated with ICT sourcing. S.Afr.J.Bus.Manage, 38(4), 33-52.

Cachon, G.P. & Fisher, M. 2000. Supply Chain Inventory Management and the Value of Shared Information. Management Science, 46(8), 1032-1048.

Carmel, E. & Agarwal, R. 2002. The maturation of offshore sourcing of information technology work. MIS Quarterly Executive, 1(2), 65-77.

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.

Carmel, E. & Nicholson, B. 2005. Small firms and offshore software outsourcing: high transaction costs and their mitigation. Journal of Global Information Management 13(3), 33-54.

Carmel, E. & Tjia, P. 2005. Offshoring Information Technology: Sourcing and outsourcing to a global workforce. Cambridge: Cambridge University Press.

Childerhouse, P. & Towill, D.R., 2003. Simplified Material Flow Holds the Key to Supply Chain Integration, Omega, 31(1), 17-27.

Chiu, H.N., 1995. The Integrated Logistics Management System: A Framework and Case Study, International Journal of Physical Distribution & Logistics Management, 25(6), 4-22.

Constantinescu, R., 2005. An Outsourcing Model of Software Development. In S. Morasca (Ed.), proceedings of the 11th IEEE International Software

Metrics Symposium (METRICS'05), Los Angeles, California, IEEE Computer Society, 3 pp. -47.

Cunningham, J.B. 1997. Case study principles for different types of cases. Quality & Quantity, 31(4), 401-423.

Dejonckheere, J., Disney, S.M., Lambrecht, M.R. & Towill, D.R. 2004. The Impact of Information Enrichment on the Bullwhip Effect in Supply Chains: A Control Engineering Perspective. European Journal of Operational Research, 153(3), 727-750.

Earl, M. 1996. The risks of outsourcing. MIT Sloan Management Review, 37(3), 26-32.

Eisenhardt, K.M. 1989. Building Theories from Case Study Research. The Academy of Management Review, 14(4), 532-550.

Elitzur, R. & Wensley, A. 1997. Game Theory as a Tool for Understanding Information Services Outsourcing. Journal of Information Technology, 12(1), 45-60.

Engardia, P. & Einhorn, B. 2005. Outsourcing innovation. Business Week, March 21, 85-94.

eSourcing Capability Model for Service providers (eSCM-SP) V2.01. 2010. http://www.itsqc.org/index.html, (2014-08-25).

Farrel, D., Laboissiere, M.A. & Rosenfeld, J. 2006. Sizing the emerging global labor market: rational behavior from both companies and countries can help it work more efficiently. Academy of Management Perspectives, 20(4), 23-34.

Fawcett, S.E., Osterhaus, P., Magnan, G.M., Brau, J.C. & McCarter, M.W. 2007. Information Sharing and Supply Chain Performance: The Role of Connectivity and Willingness. Supply Chain Management: An International Journal, 12(5), 358-368.

Feeny, D., Lacity, M. & Willcocks, L.P. 2005. Taking the measure of outsourcing providers. MIT Sloan Management Review, 46(3), 41-48.

Frohlich, M. & Westbrood, R. 2001. Arcs of integration: an international study of supply chain strategies. Journal of operations management, 19(2), 185-200.

Gorla, N. & Somers, T.M. 2014. The Impact of IT Outsourcing on Information Systems Success. Information & Management, 51(3), 320-335.

Gottschalk, P. & Solli-Sæther, H. 2006. Maturity model for IT outsourcing relationships. Industrial Management & Data Systems, 106(2), 200-212.

Gregor, S. 2006. The Nature of Theory in Information Systems. MIS Quarterly, 30(3), 611-642.

Gunasekaran, A. & Ngai, E.W.T. 2003a. Information Systems in Supply Chain Integration and Management. European Journal of Operational Research, 159(2), 269-295.

Gunasekaran, A. & Ngai, E.W.T. 2003b. The Successful Management of a Small Logistics Company. International Journal of Physical Distribution and Logistics Management, 33(9), 825-842.

Hart, P. & Saunders, C. 1997. Power and trust: critical factors in the adoption and use of electronic data interchange. Organization Science, 8(1), 23-42.

Heidegger, M. 1977. The question concerning technology. New York, Harper & Row, 3-35.

Herath, T. & Kishore, R. 2009. Offshore outsourcing: risks, challenges, and potential solutions. Information Systems Management, 26(4), 312-326.

Hevner, A.R., March, S.T., Park, J. & Ram, S. 2004. Design science in Information systems research. MIS Quarterly, 28(1), 75–105.

Hong, Y., Kim, M. & Lee, S.W. 2010. Requirements management tool with evolving traceability for heterogeneous artifacts in the entire life cycle. In J.E. Guerrero (Ed.), Software Engineering Research, Management and Applications (SERA), 2010 Eighth ACIS International Conference on, Montreal, QC, Canada, IEEE Computer Society, 248-255.

Hughes, J. & Kaplan, J. 2009. Where IT infrastructure and business strategy meet, CIO and CTO should take the lead in explaining how IT infrastructure creates business value-especially in challenging times. The McKinsey Quarterly, May 2009.

Information Technology Advisory Committee (ITAC) 2003. Information Technology Outsourcing, www.cica.ca/itac, (2014-07-15).

Järvenpää, S. & Mao, J. 2008. Operational Capabilities Development in Mediated Offshore Software Services Models. Journal of Information Technology, 23(1), 3-17.

Kaiser, K. & Hawk, S. 2004. Evolution of offshore software development: from outsourcing to cosourcing. MIS Quarterly Executive, 3(2), 69-81.

Käkölä, T. 2008. Best Practices for International eSourcing of Software Products and Services. In R.H. Sprague Jr. (Ed.), 41th Hawaii International Conference on System Sciences (HICSS), Waikoloa, Hawaii, IEEE Computer Society, 17.

Käkölä, T. & Taalas, A. 2008. Validating the information systems design theory for dual information systems, in proceeding of the 29th International Conference on Information Systems, Paris, Association for Information Systems, 119.

Käkölä, T., Koivulahti-Ojala, M. & Liimatainen, J. 2010. An Information Systems Design Product Theory for Integrated Requirements and Release Management Systems. Journal of Software Maintenance and Evolution: Research and Practice, 23(6), 443-463.

Käkölä, T. & Lu, Y. 2015. Best Practices for International eSourcing of Software Products and Services from the Service Provider's Perspective. In 48th Hawaii International Conference on System Sciences (HICSS), IEEE Computer Society.

Kern, T., Willcocks, L. & Van Heck, E. 2002. The winner's curse in IT outsourcing: strategies for avoiding relational trauma. California Management Review, 44(2), 47-69.

Kim, G. & Kim, S. 2008. Exploratory study on effective control structure in global business process sourcing. Information Resources Management, 21(3), 101-118.

Klein, R., Rai, A. & Straub, D.W. 2007. Competitive and Cooperative Positioning in Supply Chain Logistics Relationships. Decision Science, 38(4), 611-646.

Koveos, P. & Tang, L. 2004. Offshore outsourcing: Japan, Europe and the rest of the world. Indian Journal of Economics and Business, Special Issue, 3(3), 43-62.

Lacity, M. & Hirschheim, R. 1993. The information systems outsourcing bandwagon, MIT Sloan Management Review, 35(1), 73-86.

Lacity, M. & Willcocks, L.P. 2014. Nine Practices for Best-in-Class BPO Performance. MIS Quarterly Executive, 13(3), 131-146.

Lai, F., Zhao, X. & Wang, Q. 2006. The Impact of Information Technology on the Competitive Advantage of Logistics Firms in China. Industrial Management & Data Systems, 106(9), 1249-1271.

Langley, C.J., Allen, G.R. & Colombo, M.J. 2003. Third-party logistics: Results and findings of the 2003 eighth annual study. Georgia Institute of Technology.

Lazic, L. 2010. Software testing optimization by advanced quantitative defect management. Computer Science and Information Systems, 7(3), 459-487.

Lee, H.L., So, K.C. & Tang, C.S. 2000. The Value of Information Sharing in a Two-level Supply Chain. Management Science, 46(5), 626-643.

Lee, J.N., Miranda, S.M. & Kim, Y.M. 2004. IT outsourcing strategies: universalistic, contingency, and configurational explanations of success. Information Sysems Research, 15(2), 110-131.

Levina, N. & Ross, J.W. 2003. From the vendor's perspective: exploring the value proposition in information technology outsourcing. MIS Quarterly Executive, 27(3), 331-364.

Lewin, A.Y. & Peeters, C. 2006. Offshoring work: business hype or the onset of fundamental transformation? Long Range Planning, 39(3), 221-239.

Li, G., Yang, H., Sun, L., & Sohal, A.S. 2009. The Impact of IT Implementation on Supply Chain Integration and Performance. International Journal of Production Economics, 120(1), 125-128.

Li, S., Ragu-Nathan, B., Ragu-Nathan, T.S. & Rao, S.S. 2006. The Impact of Supply Chain Management Practices on Competitive Advantage and Organizational Performance. Omega, 34(2), 107-124.

Liu, J.J., So, S.C.K, Choy, K.L., Lau, H. & Kwok, S.K. 2008. Performance Improvement of Third-party Logistics Providers-An Integrated Approach with A Logistics Information System. International Journal of Technology Management, 42(3), 226-249.

Ltesting. 2014. http://en.ltesting.com.cn/(2014-11-27).

Lu, Y. & Käkölä, T. 2011. Which Test Artifacts Testing Service Providers should Reuse and How? Experiences from a Case Study in the Chinese ICT Sourcing Market. In R.H. Sprague Jr. (Ed.), proceeding of 44th Hawaii International Conference on System Sciences (HICSS). Kauai, Hawaii, IEEE Computer Society, 1-10.

Lu, Y. & Käkölä, T. 2013. An Information System Design Product Theory for Integrated Order, Transportation and Warehouse Management Systems.

In R.H. Sprague Jr. (Ed.), proceeding of 46th Hawaii International Conference on System Sciences (HICSS). Wailea, Hawaii, IEEE Computer Society, 3717-3726.

Lu, Y. & Käkölä, T. 2014. An Information System Design Product Theory for the Abstract Class of Integrated Requirements and Delivery Management Systems. In R.H. Sprague Jr. (Ed.), proceeding of 47th Hawaii International Conference on System Sciences (HICSS). Waikoloa, Hawaii, IEEE Computer Society, 3677-3686.

Ma, J.Q., Li, J.Y., Chen, W.B., Conradi, R., Ji, J.Z. & Liu, C.N. 2008. A state of the practice study on communication and coordination between Chinese software suppliers and their global outsourcers. Software Process Improvement and Practice, 13(3), 233-247.

Manning, S., Massini, S. & Lewin, A.Y. 2008. A dynamic perspective on next-generation offshoring: the global sourcing of science and engineering talent. Academy of Management, 22(3), 35-54.

Markus, M.L., Majchrzak, A. & Gasser, L. 2002. A design theory for systems that support emergent knowledge processes. MIS Quarterly, 26(3), 179–212.

McFarlan, F. & Nolan, R. 1994. How to manage an outsourcing alliance. MIT Sloan Management Review, 36(2), 9-23.

Momme, J. 2002. Framework for Outsourcing Manufacturing: Strategic and Operational Implications. Computers in Industry, 49(1), 59-75.

Mudambi, S.M. & Tallman, S. 2010. Make, buy or ally? Theoretical perspectives on knowledge process outsourcing through alliances. Journal of Management Studies, 47(8), 1434-1456.

Oshri, I., Kotlarsky, J., Rottman, J.W. & Willcocks, L.L. 2009. Global sourcing: recent trends and issues. Information Technology & People, 22(3), 192-200.

Oshri, I., Kotlarsky, J. & Willcocks L. 2007. Managing dispersed expertise in IT offshore outsourcing: lessons from TATA consultancy services. MIS Quarterly Executive, 6(2), 53-65.

Patel, P. & Vega, M. 1999. Patterns of internationalisation of corporate technology: location vs. home country advantages. Research Policy, 28(2-3), 145-155.

Plugge, A., Bouwman, H. & Molina-Castillo, F.J. 2013. Outsourcing capabilities, organizational structure and performance quality monitoring: Toward a fit model. Information & Management, 50(2013), 275-284.

Poston, R.S., Simon, J.C. & Jain, R. 2010. Client communication practices in managing relationships with offshore vendors of software testing services. Communications of the Association for Information Systems, 27(9), 129-148.

Prajogo, D. & Olhager, J. 2012. Supply Chain Integration and Performance: The Effects of Long-term Relationships, Information Technology and Sharing, and Logistics Integration. International Journal of Production Economics, 135 (1), 514-522.

Prasanna, M., Sivanandam, S.N., Venkatesan, R. & Sundarrajan, R. 2005. A survey on automatic test case generation. Academic Open Internet Journal, 15(6), 4-9.

Ring, P. & Van de Ven, A. 1994. Developmental processes of cooperative interorganizational relationships. Academy of Management Review, 19(1), 90-118.

Rottman, J. & Lacity, M. 2004. Twenty Practices for Offshore Outsourcing. MIS Quarterly Executive, 3(3), 117-131.

Runeson, P. & Höst, M. 2009. Guidelines for conducting and reporting case study research in software engineering. Empirical Software Engineering, 14(2), 131-164.

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

Sabherwal, R. 2003. The evolution of coordination in outsourced software development projects: a comparison of client and vendor perspectives. Information and Organization, 13(3), 153-202.

Sahay, S., Nicholson, B. & Krishna, S. 2003. Global IT outsourcing. Cambridge, United Kingdom. Cambridge University Press.

Salo, A. & Käkölä, T. 2005. Groupware support for requirements management in new product development. Journal of Organizational Computing and Electronic Commerce, 15(4), 253-284.

Schniederjans, M., Schniederjans, A. & Schniederjans, D. 2007. Outsourcing management information sysytems. Hershey, United States of America. IDEA Group Publishing.

Seidmann, A. & Sundararajan, A. 1997. The Effects of Task and Information Asymmetry on Business Process Redesign. International Journal of Production Economics, 50(2-3), 117-128.

Sharma, R., Apoorva, S.R., Madireddy, V. & Jain, V. 2008. Best Practices for Communication between Client and Vendor in IT Outsourcing Projects. Journal of Information, Information Technology, and Organizations, 3, 61-93.

Sheu, C., Yen, H.R., & Chae, B. 2006. Determinants of supplier-retalier collaboration: evidence from an international study. International Journal of operations & Production Management, 26(1), 24-49.

Shi, Z., Kunnathurb, A.S. & Ragu-Nathan, T.S. 2005. IS outsourcing management competence dimensions: instrument development and relationship exploration. Information & Management, 42(6), 901-919.

Stock, G.N., Greis, N.P. & Kasarda, J.D., 1998. Logistics Strategy and Structure: A Conceptual Framework. International Journal of Operations and Production Management, 18(1), 37-52.

Stock, G.N., Greis, N.P. & Kasard, J.D., 2000. Enterprise Logistics and Supply Chain Structure: The Role of Fit. Journal of Operations Management, 18(5), 531-547.

Subramaniam, M. & Venkatraman, N. 2001. Determinants of transnational new product development capability: testing the influence of transferring and deploying tacit overseas knowledge. Strategic Management Journal, 22(4), 359-378.

Swafford, P.M., Ghosh, S. & Murthy, N. 2008. Achieving Supply Chain Agility Through IT Integration and Flexibility. International Journal of Production Economics, 116(2), 288-297.

Tafti, M. 2005. Risks Factors Associated with Offshore IT Outsourcing. Industrial Management and Data Systems, 105(5), 549-560.

UN. 2003. E-Commerce and Development Report 2003. United Nations Conference on Trade and Development, New York and Geneva, United Nations.

Van Aken, J.E. 2004. Management research based on the paradigm of the design sciences: the quest for field-tested and grounded technological rules. Journal of Management Studies, 41(2), 219-246.

Van der Zee, H.T.M. & Ribbers, P.M.A. 2000. Scenarios for strategic sourcing of informarmation and communication technology. In R.H. Sprague Jr. (Ed.), proceedings of the 33rd Hawaii International Conference on System Sciences-2000, Washington, DC, USA, IEEE Computer Society, 7071.

Walls, J.G., Widmeyer, G.R. & EI Sawy, O.A. 1992. Building an information system design theory for vigilant EIS. Information Systems Research, 3(1), 36-59.

Walls, J.G., Widmeyer, G.R. & EI Sawy, O.A. 2004. Assessing information system design theory in perspective: how useful was our 1992 initial rendition?. Journal of Information Technology Theory and Application, 6(2), 43-58.

Wang, E.T.G., Barron, T. & Seidmann, A. 1997. Contracting structures for custom software development: the impacts of informational rents and uncertainty on internal development and outsourcing. Management Science, 43(12), 1726-1744.

Weber, R. 2012. Evaluating and Developing Theories in the Information Systems Discipline. Journal of the Association for Information Systems, 13(1), 1-30.

Willcocks, L., Hindle, J., Feeny, D. & Lacity, M. 2004. IT and business process outsourcing: the knowledge potential. Information systems management, 21(3), 7-15.

Xinhua News Agency (XNA). 2009. Top ten industries promotion planning, www.gov.cn (2014-11-10).

Yu, Z., Yan, H. & Cheng, T.C.E. 2001. Benefits of Information Sharing with Supply Chain Partnerships. Industrial Management and Data Systems, 101(3), 114-119.

Zhang, B. & Han, Y. 2010. How to Improve Chinese Logistic Enterprises Efficiency. In 2010 Organizing Committee EBM (Ed.), proceeding of International Conference on Engineering and Business Management (EBM 2010). Scientific Research Publishing, USA, 2939-2943.

Zhao, X., Xie, J. & Zhang, W.J. 2002. The Impact of Information Sharing and Ordering Co-ordination on Supply Chain Performance. Supply Chain Management: An International Journal, 7(1), 24-40.

Zhou, H. & Benton Jr, W.C. 2007. Supply Chain Practice and Information Sharing. Journal of Operation Management, 25, 1348-1365.

Ziqun, L., Xiaoming, Z. & Shanghai, J. 2007. 2007 Report on China'a outsourcing development. Shanghai. Shanghai Jiao Tong Univeristy Press.

ORIGINAL ARTICLES

I

Which Test Artifacts Testing Service Providers should Reuse and How? Experiences from a Case Study in the Chinese ICT Sourcing Market

by

Lu, Y. and Käkölä, T. 2011
In R.H. Sprague Jr. (Ed.), proceeding of 44th Hawaii International Conference on System Sciences (HICSS). Kauai, Hawaii, IEEE Computer Society, 1-10.

Abstract

Software testing service providers are facing new requirements to shorten service times, lower costs, and increase service customization and quality. Reuse of test artifacts is a possible solution that can help providers to meet the requirements because reuse can improve software quality and productivity. However, the extant literature does not explain in depth which test artifacts should be reused and how. This paper focuses on ICT-enabled sourcing of software testing services in the Chinese market to identify the most important reusable test artifacts. There are two reasons for this research. First, most Chinese service providers are small or medium-sized and have to overcome obstacles such as the lack of advanced core technologies before they can play important roles in the global sourcing market. Second, testing is one of the best ICT services small- and medium-sized providers can provide to develop domain and technological knowledge required to overcome most obstacles.

1. Introduction

Software has become a key asset for competitive high technology products. It has also become ubiquitous in society. High software quality and competitiveness in software development have thus become critical concerns for software businesses. Increasingly, complex high quality systems are produced with constant, or even diminishing, human resources [44].

Software testing is an empirical investigation conducted to provide stakeholders with information about the quality of the products and/or services under test with respect to the contexts in which the products and services are intended to operate [25; 5; 28]. It is divided in static and dynamic testing. Static testing is the process of reviewing documents and detecting errors without executing the software. Dynamic testing is the process of detecting faults in the software by executing the software with appropriate test materials such as automated scripts or test-specific software components [7]. Software testing is crucial for software quality assurance [17]. It is also one of the most time- and labor-intensive activities, consuming between 30% and 50% of the total development cost [4; 38; 27]. Companies are often faced with time and resource constrains, limiting their abilities to complete testing efforts effectively [28].

Software testing has proven challenging enough to warrant the establishment of an industry of specialized testing service providers. With more and more incumbents entering the markets, competition within this industry has become increasingly fierce and time-based. To meet the competitive pressures, the providers need to be capable of developing mass-customizable and cost effective services that address the needs of their clients rapidly and profitably. Therefore, testing service providers need effective ways for executing software testing.

Testing effectiveness can be radically increased by creating reusable test artifacts (hereafter, domain test artifacts) for particular software application domains. Yet, conventional software testing is application specific, providing few test artifacts that could be reused across applications. Software artifacts and test artifacts are typically developed by different teams and described in separate documents, making test artifact reuse difficult [6; 16; 45].

Software product line engineering is a new paradigm through which software-intensive products, sharing common and variable features, can be derived quickly by developing and reusing common and variable domain artifacts for the product line [35]. Common domain artifacts deal with the features that are always included in the applications, while the variable domain artifacts provide the functional and quality elements that may vary or be totally excluded from the applications. Software product line engineering has received a great deal of attention for its potential to foster reuse of software artifacts across development phases [11], improving productivity [29] and software quality, reducing development costs [37], and shortening the time to market [13]. To reap maximum benefits from software product line engineering, software providers should focus on specific application domains; create software product lines with a number of inter-related software products for the domains; establish requirements, reference architectures, and detailed designs for the domains; and radically improve testing effectiveness by creating domain test artifacts to test the requirements, reference architectures, detailed designs, and implementations of all applications [35].

Many methods and technologies have been developed to facilitate product line engineering, including design patterns, frameworks, and component implementation technologies (e.g., .NET and J2EE) [17]. However, reuse technologies have not been applied in testing service provisioning to a maximum possible extent. This is largely due to the fact that testing service providers serve multiple application domains lacking adequate commonality to warrant reuse investments.

Software product line engineering is knowledge intensive, requiring the learning, sharing, and accumulation of knowledge through knowledge repositories and social networks [15]. The creation and use of domain test artifacts benefit both testing service providers and clients. Effective use of knowledge gained from prior and current engagements enables the providers to reuse their best practices, to address problems that have occurred, and to improve overall performance and test service quality in current and future engagements [42]. Development budgets shrink because large numbers of high quality test artifacts can be built, used, and improved based on the experiences obtained when it is known that the artifacts will be used many times over long periods. These advantages, combined with the increased availability of domain artifacts, are the main reasons why reuse is becoming increasingly prominent [33].

The domain test artifacts must be stored in test artifact repositories accessible to all relevant stakeholders to enable artifact evolution and reuse. Application testers can then select, modify, and configure appropriate subsets of the artifacts to cover the features and non-functional characteristics of the applications being tested.

However, the extant research does not provide comprehensive guidelines for creating knowledge repositories for software artifacts [3; 8; 10]. Practical implementations of repositories and systematic research about the use of the repositories in real settings are also lacking. For example, Käkölä et al. [24] presents (1) detailed guidelines for designing repositories consisting of requirement, architecture, and detailed design artifacts for both domain and application levels, and their packaging into releasable entities, and (2) real life experiences from using such a repository. However, testing is out of the scope of Käkölä et al. [24]. In addition, much of the research is focused on large organizations in developed economies [12].

To facilitate the creation and use of domain test artifacts and associated repositories, this paper draws upon a literature review and a case study in the context of the Chinese ICT sourcing market. It probes the following research question:

- Which test artifacts the testing service providers should reuse during the testing life-cycle to shorten service times, reduce costs, and enhance service quality?

The chosen case organization, a medium-sized Chinese professional testing firm, is appropriate for this research for two reasons. First, the firm has developed its own test artifact repository, where all domain test artifacts are stored. Second, it has all the major characteristics of the Chinese ICT sourcing service providers. For example, it is growing fast and desiring entry into the international testing market. It can also offer various testing services. Research findings can thus be generalizable at least in the Chinese context.

The Chinese ICT sourcing market has been chosen as the context for this research because China has grown into one of the major sourcing service bases in the global ICT sourcing context [18] but the Chinese software industry is facing a great deal of challenges. Chinese providers need to overcome many obstacles before they can play important roles in the global ICT sourcing market. For example, they need to learn to scale up their production, develop competitive core technologies that can serve as domain artifacts, and develop and hire top-level executives and experts competent in international business [34]. Most Chinese providers are small or medium-sized. They typically leverage the mediated offshore sourcing model, delivering software services to larger foreign ICT clients that contract and interface with the actual end-clients onshore [23]. This business model usually restricts the providers to small, low-value projects and hampers the sharing of knowledge with end-clients, severely impeding the capability and business development of Chinese providers.

Software testing is one of the best ICT services small- and medium-sized providers can provide to develop domain and technological knowledge required to overcome most obstacles discussed above. For example, comprehensive testing services are practically impossible to develop without understanding the business domains of end-clients. Once such domain knowledge has been created, it can be deployed, for example, to broaden the scope of services from testing to software development or to create new products, thus transcending the restrictions of the mediated sourcing model.

This paper contributes to software product line research and software testing research by delineating and analyzing the most important reusable test artifacts based on a case study and by illustrating how testing service providers should manage testing services to improve service effectiveness. This research shows that test plans, test cases, test reports, and the lessons-learned documents are the main domain test artifacts. Test artifact repositories are also crucial artifacts for testing service providers.

The paper is organized as follows. Section 2 discusses the case organization and the research methodology. Section 3 presents the findings from the case study, including what to reuse, how to reuse, and who should be responsible for reuse. Section 4 discusses the characteristics of knowledge that can be reused during the testing life-cycle. Conclusions and suggestions for future research conclude the paper.

2. Description of the research methodology and the case organization

This research uses a single qualitative case study to collect data covering the complete sourcing life-cycle for testing service providers, including the most important testing practices, the artifacts reused in these practices during the various phases of the life-cycle, and the people responsible for reusing the artifacts. As a result, this research provides a holistic, systemic understanding of the phenomenon of test artifact reuse in the context of ICT-enabled sourcing of software testing services [14].

Ltesting is a medium-sized (less than 50 employees) professional software testing services provider founded in 2002 [30]. It has rapidly established a leading position in the Chinese testing service market based on its rich testing experiences and professional services. It has established strategic partnerships with HP, IBM, and some other multinational companies. It offers various types of testing-related services, such as software testing services, test training services for individuals and companies interested in offering testing services, test management services, and consulting services for constructing software quality systems. When clients source testing work to Ltesting, it expects the clients to be closely involved in the sourcing engagements in order to ensure the sourced projects meet clients' test requirements and help clients to obtain expected results. Ltesting implements most testing services offsite, being responsible solely for the test project. It also offers onsite testing services: its testers join clients' test teams and are managed by the clients. To identify the most important domain test artifacts for the service provider, this research focuses on the projects following the offsite model.

Test teams are responsible for testing. Usually the teams have four roles: test manager, test analyst, tester, and seller (Table 1). Sellers serve as boundary spanners between clients and providers. Sellers are especially important for solving communication challenges in international sourcing when clients and vendors use different languages, have different cultures [31; 36]. Test teams can be organized flexibly based on the project characteristics, personnel workloads, and client requirements. For example, when the projects are small, testers need not be involved in test teams because test managers and test analysts can do their work.

The investigation has proceeded in the following stages. First, a reference model was selected from literature to understand the international sourcing life-cycle holistically from both clients' and providers' viewpoints. The eSourcing Capability Model for Service Providers (eSCM-SP) was selected as the reference model because it has been demonstrated to help various types of sourcing service providers to improve their capabilities related to both ongoing, phase-specific, and engagement-specific sourcing practices throughout the sourcing life-cycle [43]. According to the eSCM-SP, the life-cycle involves three phases from the provider's viewpoint: initiation, delivery, and completion. This paper organizes findings related to the domain test artifacts based on the three phases to help providers reuse the artifacts during the life-cycle. Second, Chinese software industry and software testing services industry were studied [18; 31; 46;] and scientific literature was reviewed in an iterative fashion to identify the key characteristics of successful testing providers (e.g., international growth orientation, sophisticated web-based integrated information systems) and the most suitable candidates for an in-depth case study. Third, Ltesting was selected because it was successful and possessed the required characteristics. Fourth, the first author spent over three weeks observing life in the case organization, analyzing documents and memoranda, and interviewing key personnel. The in-depth interviews involved the CEO, all testing managers, and a number of test analysts to uncover the routine practices and information systems associated with testing work and major breakdowns disrupting work. Each interview was started by following a questionnaire and was concluded with an open discussion to address emerging issues. Interviews were summarized and sent to the interviewees, who verified them and provided feedback as necessary.

Title	Responsibility	Contribution to reuse
Test Manager	Test managers are responsible for test project planning, management, risk evaluation, and report review. During project completion, test managers summarize the projects and the lessons learnt. They have at least five years of relevant work experience.	Overall responsibility for management and maintenance of the test artifact repository. Identifying and selecting the test artifacts that will become domain test artifacts in the repository.
Test Analyst	Test analysts analyze test requirements; design test plans (together with the test manager); and design test cases. They have three to four years of relevant work experience.	Identify domain test artifacts that can be reused in the project. Support the conceptual integration of domain test artifacts in a specific project.
Tester	Testers conduct the specific test assignments. They have more than one year of work experience.	Submit change requests and defect reports to developers and maintainers of domain test artifacts.
Seller	Sellers communicate with clients, acting as bridges between clients and service providers. They need to have comprehensive testing knowledge, because they attend the testing service life-cycle from early bidding and negotiation through to service completion.	Receive feedback from clients concerning the test results achieved through reused test artifacts.

Table 1. Job Descriptions of Key Roles in Test Teams

Fifth, after the three-week visit, data about testing strategies, routine practices, organizational structure, and enabling classes of information systems, uncovered through observations, document analysis, and/or the interviews, were analyzed to create a preliminary software testing life-cycle model. Most significant breakdowns in routines were also analyzed because the model should help providers to improve their processes and competencies in order to eliminate most breakdowns in advance. The phases of the preliminary model were compared to the respective sourcing phases prescribed by the eSCM-SP. Testing related literature was also used. If the analysis indicated that major deviations existed or some information was missing, clarifications were requested from key informants through email. Data collection and analysis continued for several months using the internet to collaborate with the case organization.

3. Most important reusable test artifacts based on the case study

This section describes the most important domain test artifacts according to the software testing life-cycle model. Table 2 summarizes the test artifacts and life-cycle phases in which they are reused.

To facilitate reuse of test artifacts in later projects, Ltesting's test manager will document in the end of each project the experiences that the test teams have gained and the artifacts they have created. Each artifact in the test artifact repository is the result of a packaging document that prepares an existing test artifact for reuse [40]. Each artifact package includes the documentation of the requirements and design specifications, the implementation environment, usage examples, and change requests for the artifact. To guarantee the reliability and quality of the artifact, each artifact package is validated before it is admitted into the repository. A sample application demonstrates how service projects can use the domain test artifacts. Even novices can then benefit from the organizational knowledge [32].

Reusable test artifacts	Responsibility	Life-cycle phase
Test plan	Test manager and test analyst	Initiation
Test case	Test analyst	Delivery
Test procedure specifications and defect reports	Test analyst and Tester	Delivery
Summarized service reports	Test manager	Completion

Table 2. Reusable Test Artifacts

Test managers and test analysts work together to draft test plans. Test analysts design and review test cases. Test managers and test analysts must create a good conceptual understanding of the relevant domain test artifacts and artifact interfaces during service initiation to appropriately integrate the artifacts for effective service delivery. Test analysts then identify functional components of the application to be tested that are not unique to the application. It is probable that domain test artifacts for such components already exist [40]. Test analysts search for appropriate domain test artifacts from the repository. If the repository contains an artifact implementing (parts of) the required functionality, they validate that the artifact can meet the requirements and/or modify it to meet the requirements. If no domain test artifacts can be found for common application-independent components, it is probably a good idea to start creating them as soon as possible, as they will save time and money and improve service quality later.

Generic test templates for multiple domains are commonly accessible from Internet and other channels. For example, IBM Rational Unified Process [39] offers comprehensive test templates for testing services such as functional tests, performance tests, and user interface tests. However, software businesses vary in terms of factors such as technologies, markets, competitive environments, software processes, the numbers and capabilities of personnel, and corporate cultures [18]. That is why most software process improvement initiatives and quality management toolkits recommend the customization and calibration of tools and templates to meet organizational requirements. Ltesting has decided not to use generic templates. It has developed own templates because they are more detailed and specific to local needs than generic templates.

"We will not use any template directly, but select the helpful parts. The most important [selection] criterion is that the selected parts meet the test requirements."(CEO)

3.1. Reuse in test plan generation

In the initiation phase, the test manager and test analysts need to draft a preliminary test plan to decide whether it is feasible to bid and, if it is, to create a tender. The plan includes the estimated work effort (in person hours), the required time, and the price. If they win the contract, they need to draft a specific test plan to describe how they will meet client's test requirements. The specific test plan defines the scope (i.e., the features to be tested and the testing activities to be performed); the methodologies and technologies to be used; the test artifacts to be reused; the people responsible for the tasks, and other resources; and the schedule of the testing activities [20]. Using historical information about previously tested similar applications can reduce the costs and improve the accuracy and relevance of test planning. For example, information about the planned and actual schedules and work hours spent on different testing activities help to estimate the needed time and resources (e.g., testers and equipment) of the projects being planned.

"Our previous experiences improve estimation accuracy." (CEO)

When the historical data of the testing projects is stored in and accessible through appropriate repositories, it will be easier to decide which approaches should be used based on past experiences [22].

"We need to assess the test requirements and review relevant service experiences. Then we can identify the helpful artifacts that can be reused." (CEO)

Ltesting records the relevant information about each project in a test artifact repository. During both preliminary and specific test planning, test managers and test analysts draw upon the repository and their personal service experiences to choose the best practices for meeting clients' test requirements. During the research, all the interviewed managers emphasized the critical role of service experience during the service delivery.

"Rich service experience helps us understand quickly what the clients want and estimate the required work effort and time and appropriate human resources for the project, which is crucial to negotiate the right price and schedule for the project." (Test manager)

The knowledge reused in test plan generation and test case generation has been created by test managers and test analysts during the completion phase of earlier projects. The people in the roles of test managers and test analysts change over time but the knowledge they create is so role and task specific that new people in the same roles usually do not have substantial challenges in deciding what information is useful and in putting the knowledge to effective use, because they generally understand their predecessors' implicit knowledge and assumptions [1]. Therefore, they can more easily understand and deal with contextual information in the documentation that might be "incorrect, incomplete, or incoherent" and they can successfully reuse the raw, unprocessed records created as a by-product of knowledge work [32].

In the same way that design patterns are developed and stored for design reuse, test plan generation patterns can be created for test plan generation. Ltesting has developed test plan templates for different application domains, guiding test managers and test analysts to plan and do the necessary tasks without forgetting anything.

3.2. Reuse in test case generation

In the initiation phase, test analysts will draft and review test cases to meet all the test requirements. The generation cannot be fully automated but Ltesting uses different types of test case patterns to generate test cases for different types of functionality. For example, Ltesting uses a user interface test case pattern as a reference to draft test cases for user interface testing. As a result, test cases can be generated faster. Another benefit from establishing test case patterns is that service providers can continue to provide testing services even when key staff members have left.

"We require our test managers and test analysts to work together in order to draft test plans and test case patterns for every project. If someone leaves, the remaining staff or a newly hired test analyst can continue the work based on the existing plan and patterns. However, none of the test artifacts are reused as they are. We will compare relevant projects and test case execution environments to the project at hand. If the situation is the same, we only need to revise test parameters [e.g., inputs and expected outcomes of test cases]. If the situation is similar, we have to identify the differences and select the useful parts to reuse. Generally, test case patterns are used as references for test case design." (CEO)

Ltesting stores and structures test cases to several sub-repositories based on the types of test cases. For example, functional test cases, user interface test cases, and performance test cases are stored in respective repositories. Test cases are not structured into repositories according to business domains (e.g., banking, insurance) because there are functional and quality requirements such as user authentication, security, and access controls that require very similar test cases across business domains.

Test cases are produced after the test plan has been completed. The bodies of test cases should contain test sequences, which are valid for testing the target software. The behavior related to the test environment (e.g., the information about the software, hardware, and other factors enabling a stable state where the test cases for the tested piece of software can be run) should be recorded in preambles. The behavior of the test environment generally requires that the system has to be brought to a certain state before the actual testing can be initiated. Generally, testers select test cases in an iterative manner, starting with an initial test case set and selecting more test cases based on the experiences from executing the test cases [22]. This matured set of test cases should be stored in the test artifact repository for use by other stakeholders. For example, test managers and test analysts testing similar applications can get ideas from this set and design test cases productively.

To reuse and manage test artifacts and their interconnections to related artifacts, service providers need an integrated management system for requirement, feature, test, and release artifacts [24]. It should be used for all test projects and offer an easy and effortless connection to the test artifact repository, from which the test analysts are able to retrieve the necessary artifacts. Ltesting uses HP Quality Center Software (QC) [19] to manage the testing life-cycle in collaboration with its clients. QC offers a web-based globally accessible suite of applications, supporting all essential aspects of testing from requirements management through test execution to defect management. Ltesting can serve its end-clients directly based on QC. In addition, QC offers different roles for different stakeholders to ensure people can access only the information targeted to them. One of the most critical ongoing knowledge management activities on the organizational level is searching and identifying test cases that have been designed during specific engagements but are generalizable for reuse in other similar engagements. During the service life-cycle, test managers and test analysts can access the relevant test case patterns easily, helping them identify helpful artifacts, save time, and improve quality.

"During the service, we use QC to manage test cases. After the service delivery has been completed, test cases and other test artifacts are transferred to clients, but we will create test case patterns and store them in repositories for subsequent reuse." (CEO)

Regression testing is an example of the reuse of test cases in the project level. It is the selective retesting of a modified software system to ensure that the bugs have been fixed and the newly added features have not created problems with the previously implemented functionality. To apply domain level reuse, test artifacts should be developed for a representative application in the domain and reused for testing other applications within the domain.

Test cases can be modeled through Unified Modeling Language [21] but Ltesting mainly uses natural language to describe test cases. The lack of standardized test case modeling hampers reuse. Ltesting would greatly benefit from using UML to model test cases. Test managers and test analysts should use UML not only to facilitate reuse but also to test features through UML designs without executing the software. In addition, UML facilitates test automation. Ltesting has recently started to use UML to describe new test cases and rewrite the old ones.

3.3. Reusing the test environment configurations and test procedure specifications during test execution

During the test execution phase, testers use test cases, implement them, conduct the tests after constructing a specific test environment, analyze the results, and generate test reports. A test environment is composed of parts such as hardware and software, connections, environment data, and maintenance tools and processes [2]. The parts are organized into specific environment configurations enabling the tests. The configurations are deployed to simulate routines that the tested software would implement in practice. Ltesting divides test tasks into specific test assignments and presents the test requirements using a tree structure. The test execution sequence depends on the established tree structure. Newly added features are typically tested incrementally and finally the new combined system is tested. For example, testing can proceed through unit testing, integration testing, and system testing. Testers set up the test environment configurations and implement the process gradually by invoking test cases based on the hierarchy. The test environment configuration documents are developed by analyzing the domain. They can be reused.

Test results must be analyzed to facilitate reuse, because test plans may require modifications after the completion of each test cycle. The analysis provides the previous phases with feedback on test plan quality, test approach suitability, correctness of the test implementations, and the coverage of the test plan. Ltesting analyzes the results by collecting relevant data and documenting the results in test procedure specifications. These documents identify (1) special requirements for setting up the test environment and (2) the methods and formats for reporting the test results and measurements. These specifications are used to execute test cases and can be reused for testing applications belonging to the same domain.

Even if the artifacts were well organized and accessible through test artifact repositories, testers may have considerable difficulties finding, selecting, and applying the most appropriate artifacts, if they lack the domain knowledge to interpret the artifacts [32] or are newly hired and simply do not know enough about the repositories. In such cases, testers cannot execute test assignments as expected without help from test analysts. Testers of Ltesting sometimes had these problems, but Ltesting worked actively to reduce them by having regular interactions with the clients before and during the service life-cycles to improve the domain knowledge of test teams.

3.4. Reuse in test report generation

Ltesting uses two kinds of test reports. In the test execution phase, it uses defect reports to inform clients about the found defects. In the completion phase, it provides clients with summary reports about the services delivered.

During the test execution phase, service providers uncover software defects, locate the causes of defects, and correct them. It is possible to reuse the test defect reports generated for earlier application projects within a domain during the testing of new applications belonging to that domain. For example, user interface defect reports identify the causes of past user interface defects. The causes may be the same in new user interface development projects.

In the completion phase, the summary reports document the service process, found defects, defect analysis, the test execution environment, and used testing methodologies and technologies. However, service providers should go beyond simple documentation and create patterns and/or templates that they can actively reuse whenever they offer similar services for the same application domain in the future.

3.5. Reuse the summarized knowledge for improving future service deliveries

In the completion phase, Ltesting systematically reviews the service life-cycle, including the executed test cases, the test plan, and other work products produced during the previous phases, to determine the service quality and the lessons learned and to identify areas to perform better in the next projects. It conducts a formal review process, comparing the service deliverables to the test requirements specified in the test plan. If the review indicates that the conducted tests do not fully meet the test requirements, the test team will continue testing until all the requirements are fulfilled and performs a new review. When the review is positive, Ltesting will release the final defect report to the client. Without formal, repeatable evaluation processes, testing evaluations would be unnecessarily biased.

The review report resulting from the review process can be used to measure the quality of services and the performance of the provider and to improve the personnel skills. It can be reused with appropriate modifications in a particular domain to improve testing services for the same domain in the future. It can also be reused to better manage current projects. For example, when a new project is in the same domain as some previous projects, the test requirements and test sequences are similar across the projects. Project and employee performance can thus be reviewed phase by phase by comparing the project to the previous ones in order to identify problems in the project that have never occurred before and areas where the performance is below the expectations. Past experience from earlier projects may also show that some application areas in the project tend to have especially high defect densities and/or specific defects. Tests can then be targeted to those areas to uncover the defects.

Ltesting records the breakdowns in routines and their causes to the review report throughout the service life-cycle and takes corrective actions as necessary. For example, if a service delivery has been delayed by poor test artifacts, such as unreasonable test procedures or test plans, Ltesting refines or revises its service procedure and/or test artifacts. Additionally, the number and nature of errors found and a summary of actual metrics data (e.g., the realized effort) will be collected.

"We will summarize every project and compare it to previous similar projects that belong to the same domain, helping us to refine our test artifacts and improve our services."(CEO)

4. Discussion

One of the main barriers to overcome when initiating knowledge management processes within a software business is the structuring of a knowledge repository to disseminate and reuse knowledge across the organization [24; 40]. A test artifact repository has been structured in Ltesting for storing and managing domain test artifacts representing abstract design solutions for families of testing problems. The repository has facilitated the identification, selection, and reuse of test artifacts, saved time in many cases, and reduced fluctuations in service quality.

"The repository has helped us reuse the artifacts during the services, saving some time and guaranteeing the service quality." (Test manager)

Repositories are thus of critical importance to testing service providers.

Methods such as feature-oriented domain analysis [26] should be used to identify the components and configurations to be tested and then structure the tests in such a manner that the tests match the features of the product configurations. For example, Ltesting performs domain analysis for user interface testing. User interfaces are developed with different programming languages (e.g., C++, JAVA) based on different platforms (e.g., Windows, Linux) to support different screen resolutions. Domain analysis may reveal that the user interfaces must share a uniform style and a set of common features despite these differences. Common test principles can thus be used and domain test cases can be designed and reused to implement the tests for the features of these user interface variants.

Most mature reuse processes use the ideas of product-line architectures [9] and domain analysis [41] to discover families of products that share common features and qualities. In order to achieve the high levels of maturity in test artifact reuse, testing service providers need a strategic approach that focuses on a portfolio of related products in an application domain instead of unrelated projects. The products must possess enough common characteristics to make reuse investments viable. Service providers responsible for the testing phase cannot directly influence software product line engineering because their clients are responsible for developing the products. However, testing has a pervasive role in product line engineering because product line requirements, reference architectures, and reusable software components need to be tested in a coordinated, holistic fashion [35]. If service providers can develop competencies enabling them to deal with all these aspects of product line engineering, they can greatly help their clients in transitioning from project-based business models toward product-line oriented business models. In addition, if service providers can establish long-term partnerships with clients that are only interested in traditional software project business (i.e., the long-term development and maintenance of tailored client-specific systems), a large set of reusable test artifacts can be developed to test different releases of the tailored systems over time.

Based on the analysis above, many artifacts can be reused during the testing process. Two basic types of test knowledge are recognized: generic test knowledge, which applies to most projects (e.g. test plan templates), and project specific test knowledge that can be used in specific domain (e.g. test case patterns). Test artifact reuse differs from software reuse because the artifacts cannot be reused alone. They are always associated with the requirements, software components, interfaces, or features that are the subjects of testing. In order to produce reusable unit tests, software components have to be designed for reusability [27]. Analogously, component integration tests can be reused for subsystem testing if there is a reusable product line reference architecture shared across all applications to be tested [35].

5. Conclusions and further research

Software testing effectiveness should be increased by creating domain test artifacts. This research found based on a literature review and a case study that test plans, test cases, and test reports are the most important test artifacts to be reused during the testing life cycle. Lessons learned from the testing projects need to be accumulated in order to improve services in the future. Test environment and test procedure documents can be reused in specific domains. Test case patterns and test plan templates are high-level artifacts, which can save time and help novices to perform test services effectively. Significant reuse work is implemented in the initiation phase when the test managers and test analysts choose suitable reusable test artifacts from the test repositories and define the best practices to meet client requirements. Testers provide valuable feedback concerning the reused artifacts to improve the quality of domain test artifacts and the usefulness of test artifact repositories. Test tools can be reused too. For example, HP Quality Center offers test management services which can be reused across various types of projects.

This research is limited to a single case study in a Chinese medium-sized testing service provider. Further research needs to examine in other organizations and countries, which domain test artifacts are the most important ones to reuse in order to help testing service providers to locate and reuse the artifacts more easily. Future research also needs to investigate how testing service providers could become central integrators of system, subsystem, and component-specific knowledge during the systems development phases that precede domain and application testing. If they could become such integrators, they could also catalyze the transitioning of their clients from project-based businesses toward software product line companies that strategically build and reuse knowledge assets to accelerate their product development and improve their product quality and end-user experience while lowering costs. We will investigate these areas and also participate in the international standardization work on software product line testing to help testing service providers improve their service qualities and shorten delivery times.

6. References

[1] Ackerman, M.S. "Definitional and Contextual Issues in Organizational and Group Memories", Information Technology and People, 9(1), 1996, pp.10-24.

[2] Ammann, P. and J. Offutt, Introduction to Software Testing, Cambridge University Press, 2008.

[3] Aurum, A., R. Jeffery, C. Wohlin and M. Handzic (Eds), Managing Software Engineering Knowledge, Springer-Verlag, New York, 2003.

[4] Beizer, B., Software Testing Techniques (2nd edition), International Thomson Computer Press: New York NY, 1990.

[5] Bertolino, A. Software Testing Research: Achieve-ments, Challenges, Dreams, Future of Software Engineering, IEEE, 2007.

[6] Binder, R. V., "Design for Testability in Object-Oriented Systems", Communications of the ACM, 37(9), Sept., 1994, pp. 87-101.

[7] Binder, R.V., Testing Object-Oriented Systems: Models, Patterns, and Tools, Addison-Wesley, 2000.

[8] Bjornsson, F.O. and T. Dingsoyr, "Knowledge Management in Software Engineering: A Systematic Review of Studied Concepts and Research Methods Used", Information and Software Technology, 50(11), 2008, pp. 1055-1168.

[9] Bosch, J., Design and Use of Software Architectures: Adopting and Evolving a Product-Line Approach, Addison-Wesley, 2000.

[10] Chewar, C.M. and D.S. McCrickard, "Links for a Human-centered Science of Design: Integrated Design Knowledge Environments for a Software Development Process", Proceedings of the 38th Hawaii International Conference on System Sciences, Big Island, HI, United States, 2005, pp. 256-266.

[11] Cohen, M.B., M.B. Dwyer and J. Shi, "Coverage and adequacy in software product line testing", ROSATEA, ACM, July, 17, 2006.

[12] Davison, R.M. and C.X.J. Ou, "Knowledge Sharing Initiatives in a Chinese Professional Services Firm", Sixteenth Americas Conference on Information Systems, Lima, Peru, August 12-15, 2010.

[13] Due, R., "The Economics of Component Based Development", Information Systems Management, 17(1), 2000, pp. 92-95.

[14] Eisenhardt, K.M., "Building Theories from Case Study Research", The Academy of Management Review, 14(4), 1989, pp. 532-550.

[15] Feldmann, R.L. and K.D. Althoff, On the Status of Learning Software Organizations in the year 2001. In: Althoff, K. D., Feldmann, R.L., Mller, W. (Eds.) Learning Software Organizations Workshop, Lecture Notes in Computer Science, Vol. 2176, 2001, pp. 2-6, Springer, Kaiserslautern, Germany.

[16] Freeman, R.S., "Testability of Software Components", IEEE Transactions on Software Engineering, 17(6), June, 1991, pp. 553-564.

[17] Gupta, M. and M. Prakash, "Possibility of Reuse in Software Testing", 6th Annual International Software Testing Conference in India, 2006.

[18] He, M., M. Li, Q. Wang, Y. Yang and K. Ye, "An Investigation of Software Development Productivity in China", ICSP, Springer-Verlag, 2008, pp. 381-394.

[19] HP Quality Center Software Version (9.20) User's Guide (QC), Mercury Interactive Corporation, QCUG 9.2/01, 2007.

[20] IEEE 829 Standard for Software and System Test Documentation. IEEE, 2008.

[21] ISO. ISO/IEC DPAS 19505-2: Information technology -- OMG Unified Modeling Language (OMG UML) Version 2.1.2 -- Part 2: Superstructure, International Organization for Standardization, 2010.

[22] Jalote, P., An Integrated Approach to Software Engineering, Second Edition, Narosa Publishing House, 2000.

[23] Järvenpää, S. and J. Mao, "Operational Capabilities Development in Mediated Offshore Software Services Models", Journal of Information Technology (23), 2008, pp. 3-17.

[24] Käkölä, T., M. Koivulahti-Ojala and J. Liimatainen, "An Information Systems Design Product Theory for Integrated Requirements and Release Management Systems", Journal of Software Maintenance and Evolution: Research and Practice (in press), Wiley, 2010.

[25] Kaner, C., Exploratory Testing. Quality Assurance Institute Worldwide Annual Software Testing Conference, Orlando, FL, Florida Institute of Technology, 2006.

[26] Kang, K.C., S.G. Cohen, J.A. Hess, W.E. Novak and A.S. Peterson, Feature-Oriented Domain Analysis (FODA). Feasibility Study, Technical Report CMU/SEI-90-TR-21, Software Engineering Institute, USA, 1990.

[27] Karinsalo, M. and P. Abrahamsson, "Software Reuse and the Test Development Process: A Combined Approach", International Conference on Software Reuse (ICSR), LNCS 3107, 59-68, 2004.

[28] Li, Q., Y. Yang, M.S. Li, Q. Wang, B.W. Boehm, and C.Y. Hu, "Improving Software Testing Process: Feature Prioritization to Make Winners of Success-critical Stakeholders", Journal of Software Maintenance and Evolution: Research and Practice, 2010.

[29] Lim, W.C., "Strategy-driven Reuse: Bringing Reuse from the Engineering Department to the Executive Boardroom", Annals of Software Engineering, 5, 1998, pp. 85-103,

[30] Ltesting, http://www.testage.com.cn/, 2010.

[31] Ma, J. Q., J.Y. Li, W.B. Chen, R. Conradi, J.Z. Ji and C.N. Liu, "A State of the Practice Study on Communication and Coordination Between Chinese Software Suppliers and Their Global Outsourcers", Software Process Improvement and Practice, 13(3), 2008, pp. 233-247.

[32] Markus, L.M. "Toward a Theory of Knowledge Reuse: Types of Knowledge Reuse Situations and Factors in Reuse Success", Journal of Management Information Systems, 18(1), 2001, pp. 57-93.

[33] McCarey, F., M.O. Cinneide and N. Kushmerick, Knowledge Reuse for Software Reuse, Web Intelligence and Agent System, IOS press, 2008.

[34] Ministry of Information Industry of the People's Republic of China and China Software Industry Association, Annual Report of China Software Industry, MIIPRC and CSIA, Beijing, 2007.

[35] Pohl, K., G. Bockle and F. van der Linden, Software Product Line Engineering-Foundations, Principles, and Techniques, Springer, Berlin, Heidelberg, New York, 2005.

[36] Poston, R.S., J.C. Simon and R. Jain, "Client Communication Practices in Managing Relationships with Offshore Vendors of Software Testing Services", Communications of the Association for Information Systems, 27(9), 2010, pp. 129-148.

[37] Poulin, J., "The Business Case for Software Reuse: Reuse Metrics, Economic Models, Organizational Issues, and Case Studies", 9th International Conference on Software Reuse, Italy, Springer, 439, 2006.

[38] Ramler, R., S. Biffl and P. Grunbacher, "Value-based Management of Software Testing", Value-based Software Engineering Springer: Berlin, 2005, pp. 226-244.

[39] Rational Unified Process (RUP), http://www-01.ibm.com/software/awdtools/rup/, 2010.

[40] Shemesh, Y., E. Glasser and M. Levy, "Core Asset Repository Methodology (CARM) for Software Reuse", Sixteenth Americas Conference on Information Systems, Lima, Peru, August 12-15, 2010.

[41] Sodhi J. and P. Sodhi, Software Reuse, Domain Analysis and Design Process. McGraw-Hill, 1999.

[42] Sprott, D., "Enterprise Resource Planning: Componentizing the Enterprise Application Packages", Communications of the ACM, 43(4), 2000, pp. 63-69.

[43] The eSourcing Capability Model for Service Providers (eSCM-SP) V2.01, http://www.itsqc.org/index.html, 2010.

[44] Van der Linden, F., K. Schmid and E. Rommes, Software Product Lines in Action: The Best Industrial Practice in Product Line Engineering, Springer, 2007.

[45] Voas, J.M. and K.M. Miller, "Software Testability: the New Verification", IEEE Software, 12(3), May 1995, pp. 17-28.

[46] Zhang, S., L.H. Huang and D.H. Yu, "An Analysis of Information Systems Research in Chinese Mainland", Communications of the Association for Information Systems, 17(35), 2006, pp. 785-800.

II

A Dynamic Life-cycle Model for the Provisioning of Software Testing Services

by

Lu, Y. & Käkölä, T., 2014

Journal of Systems Science & Control Engineering, Vol 2(1), 549-561

Abstract

Information and Communications Technology (ICT)-enabled international sourcing of software-intensive systems and services (eSourcing) is a powerful strategy for managing businesses more effectively. China is becoming a superpower for eSourcing service provisioning, but most Chinese providers are small or medium-sized and leverage the mediated eSourcing model, delivering services to foreign ICT clients that interface with end-clients onshore. This model restricts the providers to low-value projects. This paper probes eSourcing of software testing services within the Chinese market because testing is one of the best ICT services small- and medium-sized providers can provide to develop domain and technological knowledge required to transcend the limitations of the mediated sourcing model. This paper draws upon a literature review and a case study to create and validate a life-cycle model for the provisioning of software testing services that helps providers to develop the capabilities for overcoming the limitations of mediated sourcing. This model helps clients and providers to dynamically manage the life-cycle, assigns specific responsibilities to the involved roles, and supports the tasks of the roles and the entire life-cycle with a class of integrated requirements, test, and defect management systems. This class is found to be the most important class of information systems for enabling the life-cycle.

Keywords: eSourcing, Global software development, Software testing outsourcing, Requirements engineering and management.

1 INTRODUCTION

Based on the globalization of software business and the advances in information and communication technology (ICT), ICT-enabled sourcing of software-intensive systems and services (eSourcing) continues to increase. eSourcing is used worldwide to gain comparative cost advantage and outside expertise, to improve services, and to gain access to technology (Barthelemy and Geyer, 2005). More than 50% of American Fortune 500 firms and a significant proportion of Western European and Japanese firms used offshore software sourcing already in the beginning of this millennium (Carmel and Agarwal, 2002).

Software development activities such as design, development, and testing as well as support and maintenance activities are eSourced extensively to offshore destinations such as India, China, and Russia (Adya, et al. 2008; Poston et al, 2010). Yet, eSourcing consumers need to deal with various challenges such as language, geographical, cultural, technical, and political boundaries (Orlikowski, 2002; Rottman and Lacity, 2004). Software developed by offshore providers does not always meet quality thresholds and/or functional requirements, projects are not delivered on time, too much management bandwidth is needed, and anticipated benefits such as cost savings are not received (Lacity and Rudramuniyaish, 2009; Lee, 2006; McCarthy, Martorelli, Moore, Agosta and Ross, 2004).

Many researchers have tried to find solutions to overcome the challenges and improve eSourcing performance. For example, Barthelemy (2003), Adya et al. (2008), and Dedrick et al. (2011) focused on the process analysis, covering the eSourcing life-cycle from making sourcing decisions to project closure. Lacity and Rudramuniyaiah (2009) and Barthelemy (2003) focused, respectively, on cultural issues and crucial factors (e.g., the appropriateness of the selected provider) during the eSourcing processes. However, most extant research is based on the clients' perspective and does not give enough attention to the providers' viewpoints (Gonzalez et al, 2006). The literature does acknowledge that both clients and providers need to diligently manage the details in the services to overcome cultural, communication, geographical, and other boundaries (Lacity, Willcocks and Rottman, 2008). Offshore services are critically dependent on a supply of providers capable of offering comparative cost advantage, satisfactory quality, and on time delivery despite the differences in distance, time zones, and culture (Carmel and Tjia, 2005).

China has grown into one of the major sourcing service bases in the global ICT sourcing context (He, Li, Wang, Yang and Ye, 2008; Lacity, Willcocks and Rottman, 2008). However, most Chinese providers are small or medium-sized. They typically leverage the mediated offshore sourcing model, delivering software services to larger foreign ICT clients that contract and interface with the actual end-clients onshore (Järvenpää and Mao, 2008). This business model usually restricts the providers to small, low-value projects and hampers the sharing of domain knowledge with end-clients, severely impeding the capability and business development of

Chinese providers. Järvenpää and Mao (2008) focus on the development of client-specific, process, and human resource capabilities, but their research does not cover the entire eSourcing life-cycle and cannot enhance providers' capabilities comprehensively. The extant literature does not extensively address this business model and ways to overcome its limitations. In addition, as the eSourcing of knowledge work has accelerated, theoretical models to explain the phenomenon have not kept up (Riungu, Taipale and Smolander, 2010). For example, these models do not consider dynamic changes over time (Dedrick et al., 2011).

To meet competitive pressures and to establish long-term relationships with clients, eSourcing service providers (hereafter, providers) thus need to improve the management of the sourcing life-cycle (hereafter, life-cycle) and provide services that address the needs of their clients rapidly and profitably (Webb and Laborde, 2005). Providers need comprehensive theoretical and practical guidance to overcome the restrictions of the mediated sourcing model. The research reported in this paper is part of a larger project to develop a generic, dynamic life-cycle model for the provisioning of ICT sourcing services that helps ICT service providers to develop dynamic capabilities for overcoming the limitations of the mediated sourcing model. The project will span two domains: ICT service sourcing (ICTS) and business process sourcing (BPS). Dynamic capabilities refer to the abilities of organizations to maintain their flexibility by creating competencies to address external pressures (Schwarz, Kalika, Kefi and Schwarz, 2010). They help clients and providers reconfigure human and other resources to address changing environments and requirements. Most importantly, they help providers to develop differentiated services and sometimes even high-tech products for international markets. We expect that providers and clients can draw upon the life-cycle model to establish dynamic capabilities, enabling them to interact transparently, monitor the life-cycle in real time, identify communication and coordination breakdowns, and flexibly reconfigure resources to recover from breakdowns and to eliminate similar breakdowns proactively in future (c.f., Käkölä and Taalas, 2008).

This paper focuses on the ICTS domain and, especially, on the ICT-enabled sourcing of software testing services in the context of the Chinese ICT sourcing market. There are two reasons for this focus. First, software testing is one of the best ICT services small- and medium-sized Chinese providers can provide to develop in-depth domain and technological knowledge and other dynamic capabilities. Comprehensive testing services are practically impossible to develop without understanding the business domains of end-clients (Riungu, Taipale and Smolander, 2010). Developing the domain knowledge takes time and effort but once providers have created it, they can deploy it, for example, to broaden the scope of services from testing to software product development, thus transcending the restrictions of the mediated sourcing model. Second, high software quality is paramount as software has become ubiquitous in society. Software testing is an empirical investigation conducted to provide stakeholders with information about the quality of the products and/or services under test, with respect to the context in which they are intended to operate, the purpose of testing should be measure the dependability of tested software (Hamlet, 1994). It involves not only technical tasks but also considerations of economics and human psychology. Complete testing of complex applications takes too long and requires too many human resources to be economically feasible (Myers, 2004).

Capability Maturity Model Integration (CMMI) (Chrissis, Konrad and Shrum, 2011; Paulk, 2009) provides a reference model for organizational process and product improvement widely used in the software industry, for example, to decrease costs and to improve on-time delivery. However, CMMI is often regarded as the industry standard for software process improvement, not for testing process improvement because it covers testing at a relatively abstract level. To supplement the CMMI, several testing process reference models are available (Sanz, Garcia, Saldana and Amescua, 2009). Test Maturity Model Integration (TMMi) (Van Veenendaal, 2012) and Test Process Improvement (TPI) (Koomen and Pol, 1999) are the most widely used. TMMi, a reference model for testing process improvement structured like CMMI, defines five levels to assess the maturity of the testing process in an organization. TMMi helps organizations to improve their test processes toward proactive defect prevention, statistical process planning and control, appropriate test automation, and effective reuse of test artifacts. To reach a specific maturity level higher than 1, an organization must satisfy all of the appropriate goals of all the process areas at the specific level and also those at earlier maturity levels. TMMi thereby requires significant and carefully coordinated investments and extensive top management commitment. It does not specify information systems required in various maturity levels to support test processes. TPI is a framework used to improve an existing testing process or to develop a new one. It involves 20 key areas. Each area views the testing process from a specific viewpoint and is classified into two, three, or four levels of maturity. The areas impact the overall effectiveness of the testing process differently. Organizations can prioritize the implementation of the key areas based on their own needs, enabling light-weight bottom-up implementation. The interdependencies of the key areas and their maturity levels are presented in a Test Maturity Matrix, because the ability of the organization to reach a specific maturity level in a key area (e.g., collecting statistical data in the Metrics area about the defects found) may depend on reaching a specific maturity level in another key area (e.g., Defect Management). One key area deals with information systems to plan, control, execute, and automate the testing process. To reach a high maturity level in this key area, a number of sophisticated information systems needs to be leveraged. However, TMMi and TPI do not address the roles and associated competences and responsibilities required in the testing process. A well-defined testing reference model should include the necessary processes and their practices, detailed tasks in every phase, roles, technical instructions, and enabling information systems. Sanz, Garcia, Saldana and Amescua (2009) create a process model and an organizational structure for a test factory that comprehensively cover testing practices, roles, responsibilities, and competencies. However, Sanz, Garcia, Saldana and Amescua (2009) do not investigate the information systems required in the testing services.

This paper draws upon a literature review on software testing as a service (Riungu, Taipale and Smolander, 2010; Floss and Tilley, 2013; Yu, Tsai, Chen, Liu, Zhao, Tang and Zhao, 2010), testing process models (Burnstein, Suwanassart and Carlson, 1996; Canfora and Di Penta, 2006; Wang, Qian and Zhu, 2012), eSourcing services (eSCM-SP, 2010) and a case study in the context of the Chinese ICT sourcing market to create a dynamic eSourcing life-cycle model for the provisioning of software testing services, which includes the most important testing practices, the people responsible for specific testing assignments, and the information systems supporting the life-cycle. Dynamic capabilities are important for providers, because clients typically change their test requirements based on the results of test executions, requiring providers to replan the testing projects, reconfigure resources, and adjust schedules. The investigated case organization has all the major characteristics of the Chinese ICT sourcing service providers. For example, it is medium-sized but it is growing fast and desires entry into the international sourcing market. It is thus an appropriate organization to start developing a generic life-cycle model for Chinese ICT sourcing service providers. The created life-cycle model has been reviewed by the CEO and all the test managers of the case company and revised based on their feedback to ensure practical relevance. The eSourcing Capability Model for Service Providers (eSCM-SP, 2010) is a cornerstone of the created model, but eSCM-SP does not offer guidance for designing and using information systems to support clients and providers. Riungu et al (2010) and Floss and Tilley (2013) focus on software testing as a service but their research does not cover the entire life-cycle. The research papers of Burnstein, Suwanassart and Carlson (1996) and Wang et al (2012) on testing process models do not offer guidance for designing information systems to support the stakeholders. The life-cycle model presented in this paper includes the most important information systems for providers to support the stakeholders' work throughout the life-cycle. It is a comprehensive testing sourcing model for providers. This model has been reviewed and approved by the CEO and all the test managers of the case organization. They have more than ten years of testing experience.

This paper focuses on the following research question: which software testing practices and classes of information systems are the highest priority ones for Chinese testing service providers (hereafter, providers) from the viewpoint of executing the eSourcing life-cycle, designing and delivering the services, recovering from coordination breakdowns, and proactively eliminating most breakdowns to improve service delivery and to ensure organizational long-term effectiveness?

The paper proceeds as follows. The next section describes the research methodology and the case organization. Section "A dynamic eSourcing life-cycle model for software testing" presents the life-cycle model and validates it through a case study and a literature review. The section "Discussion" summarizes the most important components of the life-cycle model. Conclusions and ideas for future research are presented in the last section.

2 DESCRIPTION OF THE RESEARCH METHODOLOGY AND THE CASE ORGANIZATION

The investigation has proceeded in the following stages. First, a reference model was selected from the literature to understand the international eSourcing life-cycle holistically from both clients' and providers' viewpoints. The eSourcing Capability Model for Service Providers (eSCM-SP) was chosen as the reference model because it has been demonstrated to help various types of providers to improve their capabilities related to both ongoing, phase-specific, and engagement-specific sourcing practices throughout the life-cycle (eSCM-SP, 2010). eSCM-SP covers both the ICTS and BPS domains of this research project. The model drafted in this paper has been aligned with eSCM-SP but it is more operational for testing service providers. The eSCM-SP life-cycle (Table 1) involves three phases from the provider's viewpoint: initiation, delivery, and completion. Ongoing practices are run throughout the life-cycle to perform management functions. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, performance management). The capability areas include 84 specific practices. eSCM-SP prescribes five capability levels. Certified assessors can use eSCM-SP to determine the capability levels of providers. Clients can use the certifications to find and select providers. Providers can use eSCM-SP as a roadmap to improve their capabilities to higher levels. This investigation collected data and compared the practices of the case organization to eSCM-SP based on the three phases and specific practices.

Second, scientific literature was reviewed in an iterative fashion to identify the key characteristics of successful testing service providers (e.g., international growth orientation, sophisticated web-based integrated information systems) (Lacity, Willcocks and Rottman, 2008; Canfora and Di Penta, 2006). Third, Chinese software industry and software testing services industry were studied (Ma et al, 2008; Zhang et al, 2006; He, Li, Wang, Yang and Ye, 2008) to identify the most suitable candidate for a case study. Fourth, a company called Ltesting was selected because it had the required characteristics.

Fifth, the first author spent over three weeks observing life in the case organization, analyzing documents and memoranda, and interviewing key personnel. In-depth interviews involved the CEO, all test managers, and a number of test analysts to uncover the routine practices and information systems associated with testing work and the major breakdowns disrupting work. Each interview was started by following a questionnaire and concluded with an open discussion to address emerging issues. Interviews were summarized and the summaries were sent to the interviewees, who verified them and provided feedback as necessary. Sixth, after the three-week visit, the data collected about testing strategies, routine practices, organizational structure, and enabling classes of information systems were analyzed to create the preliminary life-cycle model. Most significant breakdowns in routines were also analyzed because the adoption of the finalized model should help organizations proactively eliminate most breakdowns (Käkölä and Taalas, 2008). The phases of the preliminary model were compared to the respective phases prescribed by the eSCM-SP and to the relevant testing-related literature (Riungu, Taipale and Smolander, 2010; Floss and Tilley, 2013; Yu, Tsai, Chen, Liu, Zhao, Tang and Zhao, 2010; Burnstein, Suwanassart and Carlson, 1996; Wang, Qian and Zhu, 2012). If the analysis indicated that major deviations existed or information was missing, clarifications were requested from informants through email. Data collection and analysis continued for several months using the internet to collaborate with the case organization. A year after the first round of interviews, the first author performed a second round of interviews in the case company to collect supplementary data related to breakdowns and workarounds. This time, the quality assurance manager, the measurement process manager, and other people supporting the test teams were also interviewed.

<table>
<tr><td colspan="3">Ongoing practices represent management functions that need to be performed during the entire eSourcing life-cycle in order to meet the intent of these practices.</td></tr>
<tr><th>Initiation</th><th>Delivery</th><th>Completion</th></tr>
<tr><td>Practices focus on the capabilities needed to effectively prepare for service delivery. The practices are concerned with collecting and analyzing service requirements, negotiating, contracting, and designing and deploying the services, including the transfer of the necessary resources.</td><td>Practices focus on service delivery capabilities, including the ongoing management of service delivery, verification that commitments are being met, and the financial management associated with service provision.</td><td>Practices focus on the capabilities needed to effectively close an engagement with particular client(s) at the end of the eSourcing life-cycle. They include the capture of the lessons learned from the engagement and the transition of resources to the client, or to a third party, from the provider.</td></tr>
</table>

Table 1. The eSCM-SP V2.01

Ltesting is a medium-sized (less than 50 employees) professional software testing services provider founded in 2006 (Ltesting, 2014). Senior staff members (e.g., test managers and test analysts) have more than ten years of work experience in providing testing sourcing services. It has established a leading position in the Chinese testing service market and set up strategic partnerships with HP, IBM, and some other multinational companies. It offers software testing services, test training services for individuals and companies interested in offering testing services, test management services, and consulting services for constructing software quality systems. Therefore, they are able to deliver testing services based on the test plans of clients and to design complete testing life-cycle services for clients. Ltesting expects the clients to be closely involved in the sourcing engagements in order to ensure the sourced projects meet clients' test requirements and help clients to obtain expected results. Ltesting implements most testing services offsite, being responsible solely for the test projects. It also offers onsite testing services: its testers join clients' test teams and are managed by the clients. To best address the research question probed in this paper, this research focuses on the projects following the offsite model. The proposed model is generic enough to cover all offsite services Ltesting offers.

Test teams are responsible for testing. Specific roles with clearly defined responsibilities need to be designed and appropriately motivated and skilled team members need to be allocated to these roles, so the members can meet their obligations and the test teams can fulfil their responsibilities (Käkölä and Taalas, 2008; Zhu and Zhou, 2006a, 2006b). Usually the teams have four roles: test manager, test analyst, tester, and seller (Table 2). Sellers serve as boundary spanners between clients and providers. They are especially important for solving communication challenges in international sourcing when clients and providers use different languages and have different cultures (Poston et al, 2010; Ma et al, 2008). Test teams can be organized flexibly based on the project characteristics, personnel workloads, and client requirements. For example, when the projects are small and there are not many projects, testers need not be involved in test teams because test managers and test analysts can do their work.

Title	Responsibility
Test Manager	Test managers are responsible for test project planning, management, risk evaluation, and report review. During project completion, test managers summarize the projects and the lessons learned. They have at least five years of relevant work experience.
Test Analyst	Test analysts analyze test requirements; design test plans (together with the test manager); and design test cases. They have three to four years of relevant work experience.
Tester	Testers conduct the specific test assignments. They have more than one year of work experience.
Seller	Sellers communicate with clients, acting as bridges between clients and providers. They need to have comprehensive testing knowledge, because they attend the life-cycle from early bidding and negotiation through to service completion.

Table 2. Job Descriptions of Test Team Members in the Case Organization

To deliver comprehensive testing sourcing services, test teams need support from three other teams: measurement management teams, quality assurance management teams, and configuration management teams (Table 3). The three teams work together with test teams and clients throughout the life-cycle. They provide relevant information to help test teams and clients track the service processes and manage the service quality.

Testing service providers need to choose suitable test metrics to manage and evaluate the effectiveness of testing processes. Test metrics are important service effectiveness indicators of the software testing process, helping service providers and clients evaluate the service progress and quality (William, 2006). Measurement management teams are responsible for choosing the metrics and collecting metrics data. To choose the suitable test metrics for the projects, measurement management teams need to identify the key software testing processes that should be objectively measured based on the priorities of test requirements. Service providers can improve their capabilities by comparing their performance with similar previous projects and/or with industry benchmarks. The metrics data should be collected and analyzed using relevant collection processes and analysis methods and traced, calculated, and managed effectively by the measurement management teams. Finally, the measurement management teams need to send analysis reports to the stakeholders to help them evaluate, manage, and improve the services and the enabling processes (Naik and Tripathy, 2008).

Title	Responsibility
Test Team	Test teams are responsible for test assignments, which include drafting test plans, executing tests, finding and reporting defects, and co-operating with other teams to deliver test services meeting clients' requirements.
Measurement Management Team	Measurement management teams are responsible for collecting and analyzing metrics data to help test teams and clients to improve and manage the quality of test services and to monitor the progress of testing services delivery. Clients and test teams can adjust or even redesign service processes based on measurement results.
Quality Assu-rance Manage-ment Team	Quality assurance management teams are responsible for ensuring that testing services can meet clients' requirements and are run right. Appropriate quality criteria help to ensure that test teams focus on the most important tasks needing more resources.
Configuration Management Team	Configuration management teams are responsible for managing the configurations of different versions of test artifacts during the services, dealing with version conflicts, and managing changes of test artifacts.

Table 3 Job Descriptions of Teams Involved in Testing Sourcing Services

Quality assurance management teams have managers and staff members. They need to ensure the services are executed in accordance with the approved quality criteria, specifications and standards and by leveraging the methods, standards, tools, and skills that are commonly recognized as the suitable best practices for the context (Naik and Tripathy, 2008; Iqbal and Rizwan Jameel Qureshi, 2009). Based on the test results and other information, they need to investigate, rank, and report any problems and risks in the design, planning, and execution of service processes.

Validation and verification are important activities of quality assurance management teams. Validation is a process of finding out if the test requirements are right (Ammann and Offutt, 2008), that is, whether the service provider is meeting the expectations of the clients. Verification is a process to ensure that the test teams implement testing services in the right way and follow predefined rules. Methods and techniques used in verification and validation should be designed carefully. The planning of verification and validation processes must thus start right from the beginning of the testing process (Adrion, Branstad, and Cherniavsky, 1982). As the testing services are delivered through different phases, quality assurance management teams need to ensure that all required specifications are met. Each verification process has a corresponding validation process that starts after the verification process ends (Bertolino, 2007; Biffl, Aurum, Boehm, Erdogmus, and Gruenbacher, 2006).

Configuration management teams are responsible for managing versions of test artifacts that are used and/or automatically generated in the services. For example, the configurations of test scripts, tools and hardware should follow the configuration management rules (Conradi and Westfechtel, 1998). Therefore, service providers should have relevant configuration management contents in the test plan and ensure they follow the rules during the service processes. Configuration management is necessary for the testing sourcing services. For example, if clients and service providers have any conflicts about the accuracy of test results, it is possible to check whether the software and hardware they use are the same. Configuration management aims at providing stakeholders with the right test artifacts at the right time (Estublier, 2000). When clients are repairing the defects and preparing new versions for testing, configuration management teams need to provide them with knowledge of the test environment. When new versions are tested, the teams ensure that service providers and clients work on the same versions.

The case company can organize testing teams flexibly based on client requirements (e.g., financial pressure) or project charters (e.g., duration, size). For example, team members can execute tasks of more than one role if the project charter and client requirements permit or necessitate it. Testers can collect and analyze metrics during test execution, thus undertaking measurement management responsibilities.

> “We have all the required people, enabling us to organize testing teams easily. Based on the client requirements and the project charter, we can design the involved roles in the testing teams before initiating the services. Stable teams benefit both clients and providers.” (CEO)

3 A DYNAMIC ESOURCING LIFE-CYCLE MODEL FOR SOFTWARE TESTING

This section presents the dynamic eSourcing life-cycle model for software testing from the provider's viewpoint (Figure 1). The relationships between clients and providers in the model are bilateral and dynamic as clients can modify test requirements and test plans during the life-cycle. The model involves feedback loops, adjustments, and revisions over time, enabling providers and clients to communicate effectively, avoid misunderstandings, and quickly reconfigure resources (Beizer, 1990; Karinsalo and Abrahamsson, 2004; Ramler, Biffl and Grunbacher, 2005). To validate the model, this section investigates the case organization and analyzes the relevant testing practices to compare the provider's experiences to the reference model provided by eSCM-SP. The findings are organized based on the initiation, delivery, and completion phases of eSCM-SP to offer additional insights about the practices of the provider.

Ltesting uses HP Quality Center Software (QC, 2009) to manage the life-cycle in collaboration with its clients. QC offers a web-based globally accessible suite of applications, supporting all essential aspects of testing from requirements management through test execution to defect management. Ltesting can serve its end-clients directly based on QC. There are many other similar suites in the market such as open source tools Bugfree and Testlink. This paper will use QC as an example to present how the requirements, test, and defect management tools in general support clients and providers during software testing projects.

3.1 Initiation

The initiation phase starts when a request for tender is received. The provider needs to assess the request and decide whether to create a tender or not. To make this decision, Ltesting's test manager and test analyst will usually work together to draft a preliminary test plan. If the plan shows that the tender represents a profitable opportunity, the tender is created and submitted to the client. If the contract is won, the client is responsible for providing the provider with the necessary resources and most importantly, the test requirements and access to people who created the requirements (e.g., requirements engineers, architects, and/or component developers). The provider develops a detailed test plan (together with the client) and the test cases and establishes the test environment.

3.1.1 Verify test requirements based on the request for tender

Clients present requests for tender to attract tenders and to evaluate and select the best providers. Requests for tender should include specific testing service requirements. Based on the request for tender, the provider analyzes test requirements and verifies the requirements. Test requirements are executable client requirements, which should fully cover client requirements, so no requirements are missed or overlooked. Sometimes, it is impossible to form an executable test plan because the test requirements are unclear or incomplete. If the applicable laws for tendering allow it, providers can request the client to elaborate on some requirements to ensure they understand what the client wants. Elaborations must be made available to all providers to afford fair bidding.

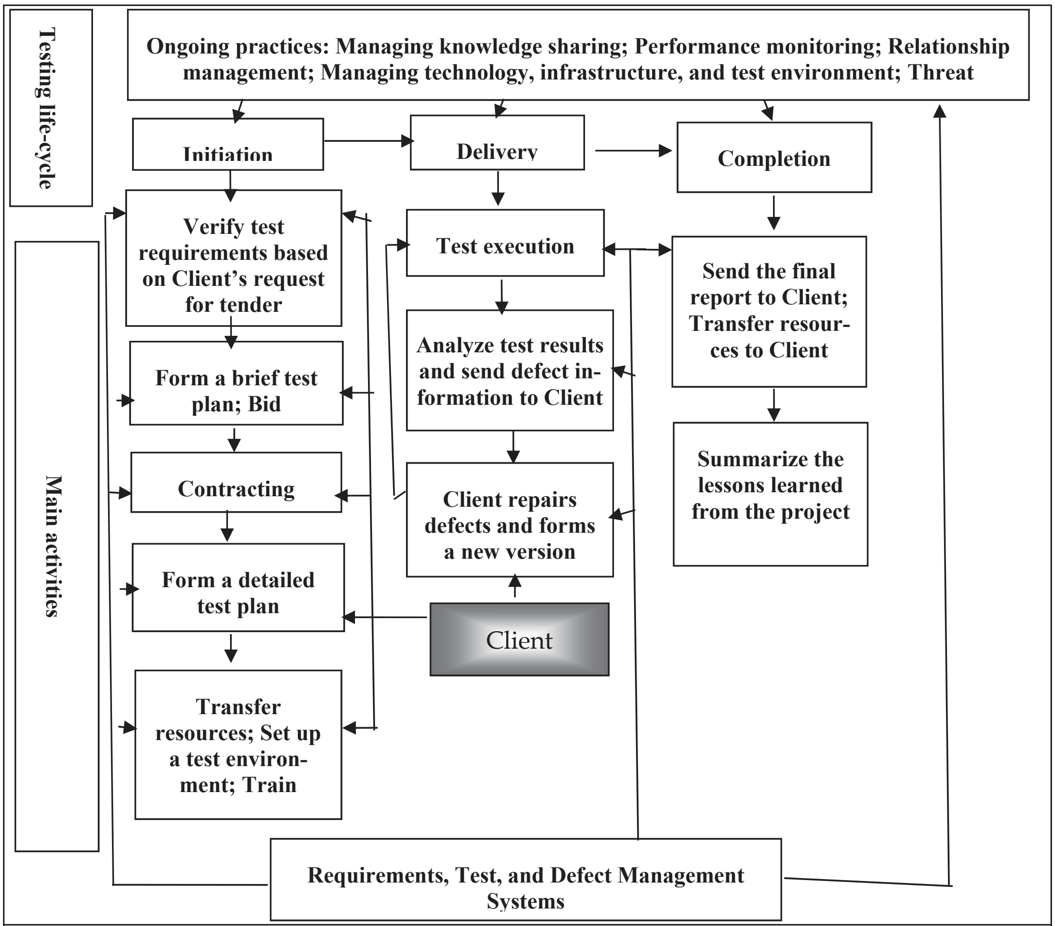

Figure 1. A Generic, Dynamic eSourcing Life-cycle Model for the Provisioning of Testing Services

According to the case study and the literature review (Boehm, 2001; Li, 2012), the biggest risk of testing eSourcing is service delay. Delays result if providers cannot verify the service scope and test requirements before test execution. Scoping is especially challenging when client requirements change often. Therefore, requirements management is crucial for clients and providers throughout the life-cycle.

3.1.2 Form a brief test plan and bid

Providers draft preliminary test plans to assess the profitability of the requests for tenders and to show clients, how and why they can meet the test requirements. If the providers decide to bid, tests plans will be attached to the bids. Clients review and compare the bids and test plans with respect to their requests for tender, select the best providers, and sign contracts with the winning bidders. Ltesting's test plans include the estimated work effort (in person hours), time, and price.

> "We need to assess the required work effort and bid for the project. Client will choose the most suitable provider based on their requirements. Price is only one of the considerations." (CEO)

To estimate the required work effort and form the preliminary test plan, Ltesting uses the Work Breakdown Structure (WBS) method. WBS is a tool used to define and group a project's discrete work elements (or tasks) in a way that helps define the total scope of the project (Brotheron, Fried and Norman, 2008). A complex project is broken down into individual components in a hierarchical structure, which can be outlined as a test task tree. Work elements can be completed independently of other elements, facilitating resource allocation, the assignment of responsibilities, and the monitoring of progress.

> "Our previous experiences make estimates more accurate." (CEO)

3.1.3 Prepare and sign the contract

The contract should define responsibilities and commitments for the client and the provider (Gopal, et al. 2003) such as the test scope and duration; the resources to be transferred or shared between the parties; security provisions; mechanisms for solving conflicts and dealing with service delays and requirements changes; rights for developed software assets and intellectual property; performance measures; quality assurance through process assessment methods such as ISO 15504 (SPICE) or CMMI; pricing; and milestones and deliverables. Contracts help manage the relationships and control risks during the life-cycle. If a client causes a service delay and the predefined work effort in hours is exceeded by more than 20 per cent, Ltesting will renegotiate the contract.

3.1.4 Form a detailed test plan

Ltesting drafts a detailed test plan and designs test cases after winning the contract. The plan includes the schedule and the methodologies and technologies to be used. Based on the Work Breakdown Structure analysis, test managers define the test tasks. To outline a strategy to achieve the test requirements, they also define testing techniques (e.g., stress test, performance test), mechanisms to handle defects (e.g., severity classification, authorization to open and close defects), required resources (e.g., hardware, personnel), and milestones. Before Ltesting drafts a detailed test plan, it stores all the requirements documents received from clients (e.g., requirements specification and design specification) to a repository and lists them as a requirements tree in order to manage the test requirements during services.

The client reviews the detailed test plan to determine how well it meets the goals defined in the beginning of the life-cycle. When the client has approved the plan and associated test cases, the testing project will move to the delivery phase.

Ltesting can use QC to detail test plans based on Microsoft Word documents. QC can read the plans from Microsoft Word documents and create the requirements trees and test plan trees accordingly. QC's requirements management module is an integrated solution for capturing, managing, and tracking requirements throughout the application development life-cycle. Usually, Ltesting begins to use this module after it has detailed test plans because of the rigidity in customizing QC. The extent to which Ltesting will use the module to capture requirements for the systems to be tested depends on the quality of clients' requirements specifications and management processes. In the beginning of service engagements, it is impossible to know the maturity levels of the clients' requirements management processes. When maturity levels prove poor, Ltesting needs to communicate and negotiate with clients to elicit, clarify, and prioritize requirements and detail the scope of each engagement until both parties approve the scope.

> "It depends on whether the client's requirements management and requirements specifications are poor or not. We seldom use QC to elicit requirements but we use QC to manage requirements." (CEO)

When the service scope and the test requirements have been validated, the test manager and test analyst(s) will add traceability links between appropriate requirements. When clients change requirements during the delivery phase, traceability links show the other requirements the changes may impact.

3.1.5 Transfer resources, set up a test environment, and arrange training

Resources to be transferred or shared are elaborated in the detailed test plan. They include test equipment, infrastructure, the software system to be tested, and the definition of the work context where the system is to be used. Before starting the service delivery, the provider needs to set up a test environment and arrange training for the test team and the client representatives involved. During training, stakeholders share relevant business (domain), process, architectural, and organizational knowledge. Whenever clients provide specific software and hardware platforms such as high-performance servers, Ltesting needs to familiarize itself with the test environment and the platforms.

In the eSCM-SP model, three capability areas are used in the initiation phase: contracting, service design and development, and service transfer. Ltesting uses all practices of these areas relevant to the initiation phase.

3.2 Delivery

This section describes based on the case study and the literature review (Ammann and Offutt, 2008; QC, 2009) how providers usually deliver services based on test plans and how Ltesting implements requirements, test, and defect management using QC.

QC's release management module helps both clients and providers manage application releases and development cycles efficiently. The provider can track the progress of application development to determine whether the release will take place as planned and to make informed budgetary and staffing decisions accordingly. QC can be used to define different roles for clients and providers (e.g., developer, project manager, and tester). Each role has different access rights and authorities.

3.2.1 Test execution

Based on the detailed test plan and test requirements, the tester typically runs both automated and manual tests to find defects. Before test execution, Ltesting's test manager will allocate test tasks to testers and ensure all the test requirements are covered and traceable to tests. QC supports functional, regression, load, unit, integration, system and other types of testing. Each type of testing has its own set of requirements, schedules, and procedures. QC helps providers and clients to monitor and control the execution process.

> "Of course clients have the right to choose the management tools used in the testing services, but we recommend QC. Our accumulated knowledge and test artifacts are stored in QC, helping us to reuse them easily and to deliver more efficient services." (CEO)

3.2.2 Analyze the test results and send defect information to stakeholders

Locating application defects efficiently is the main purpose in the delivery phase. Following a test run, the provider analyzes the test results to identify which tests failed and which steps caused the failure. The analysis also needs to determine whether a defect has been detected in the application. If no defect caused the test failure, the expected results of the test may need to be updated.

When Ltesting's testers find defects in an application, they submit the defects to the respective QC project. The project stores defect information for retrieval by authorized users such as the members of the development, quality assurance, and support teams. To help clients repair the defects, the defect reports include detailed defect information such as the related requirements, run steps, and defects.

3.2.3 Client repairs defects and forms a new test version

Clients and providers need to work together effectively to manage the entire defect life-cycle (Riungu, Taipale and Smolander, 2010) from initial problem detection through fixing the defect to verifying and validating the fix. The provider sends information about the newly found defects to the client. The client's development team repairs the defects, submits a new release, and requests the provider to execute it and analyze the results. If some defects occur again or severe new defects are found, both parties need to continue for another round of the loop.

When requirements change, a change impact report details the affected requirements, often enabling the provider to avoid a full regression test after each change. Regression testing can be performed selectively or for the complete product. Normally, full regression testing is executed during the end of the testing cycle and partial regression testing is run between the test cycles (TestingGeek, 2010). QC notifies dispersed teams of any requirements changes possibly affecting the tasks they are working on.

To support asset sharing and reuse, QC provides version control for requirements, tests, test scripts, and business components. Versioning enables dispersed testing teams to manage multiple versions of test artifacts in parallel, while providing an audit trail of changes throughout the life-cycle of each engagement. Version control thus helps clients and providers manage and track changes (Koivulahti-Ojala and Käkölä, 2010). The attributes of all the stored documents include name, status, version number, and author to help clients and providers to avoid parallel, conflicting changes of the shared files in a multiuser environment.

According to eSCM-SP, the delivery phase is associated with the service delivery capability area including eight practices: plan service delivery, train clients, deliver service, verify service commitments, correct problems, prevent known problems, service modifications, and financial management. Ltesting uses all of them except for the practice "train clients" that is executed in the initiation phase.

3.3 Completion

During the completion phase, the provider prepares the final report, transfers resources to the client (and, possibly, to a third party), and summarizes the lessons learned from the project. Clients need to ensure that the results of the engagements meet predefined acceptance conditions (e.g., the defect curve is in the convergent state and all the requirements have been met).

3.3.1 Send the final test report

In the end of the life-cycle, the provider sends the final report to the client and transfers the resources agreed upon in the contract to the client or third parties. The final report should include the test results, the recorded defects, defect analyses, test logs, and other test documents. In addition to technology, infrastructure, and knowledge resources, the test cases are transferred to the client because, based on the industry convention, clients have the copyrights of test cases.

3.3.2 Summarize the lessons learned

Summarizing and documenting the lessons learned from the engagements is important for providers from the viewpoint of continuous improvement of service capabilities and quality. For example, Ltesting compares the actual service duration to the duration estimated in the test plan, the number of actual working hours to the estimated working hours, the testers' actual performances to the expected ones, and the actual costs of resources to the estimated costs in the test plan. This information helps Ltesting to plan future projects more accurately and improve capabilities dynamically.

In the completion phase, Ltesting uses all practices of the service transfer capability area relevant to the phase: service continuity and resources, personnel, and knowledge transferred out.

3.4 Ongoing practices

Ongoing practices represent management functions that need to be performed throughout the eSourcing life-cycle to meet the intent of these practices (eSCM-SP, 2010). The ongoing practices of eSCM-SP involve six capability areas: knowledge management, people management, performance management, relationship management, technology management, and threat management. The practices of Ltesting include most of the practices of the six areas. However, Ltesting does not have appropriate practices to support innovation and continuously improve their service capabilities.

Most ongoing practices are enabled by requirements, test, and defect management systems during the life-cycle. For example, the performance management capability area focuses on managing organizational performance so that the client requirements are met and the organization keeps learning and improving its performance. The area is enabled by requirements, test, and defect management systems (c.f., Käkölä, Koivulahti-Ojala, and Liimatainen, 2011). QC implements most common requirements of the class of requirements, test, and defect management systems but it does not qualify as an instance of the class because other software products and manual routines are necessary to enable the life-cycle.

Threat management and relationship management capability areas deal with the project and relationship risks. Poor project performance typically leads to relationship risks. Poor requirements management usually causes project risks such as service delays and breakdowns. Service breakdowns can happen at anytime and anyplace due to, for example, the changing client requirements. Ltesting has set up appropriate ongoing practices and supporting information systems (including QC) to proactively eliminate some breakdowns before they occur and to deal with the emerging breakdowns. When breakdowns occur, Ltesting typically creates new knowledge together with clients to resolve the situations and get routines back on track. Any changes in requirements will lead to the re-evaluation of the detailed test plan. Ltesting assesses the impacts of new, changed, and deleted requirements on the other requirements and the required work efforts mostly based on the traceability links between the requirements and between requirements and other test artifacts. Knowledge management capability area plays a crucial role in both sharing and securing critical knowledge assets and building trusted relationships. Ltesting creates generic test artifacts based on the test artifacts created in earlier engagements and adapts and reuses them in subsequent engagements, helping Ltesting to shorten the development time, to improve the quality of test artifacts (e.g., test plans and test cases), and to achieve higher client satisfaction.

People management capability area refers to managing and motivating personnel to deliver services effectively. Based on the investigation, Ltesting has to improve its abilities in this capability area because existing competencies are not reviewed and developed systematically and career paths are not planned. Technology management capability area also needs to be improved. During each engagement, Ltesting arranges a person to manage the technology infrastructure with the client. However, no individual is specifically responsible for new technology initiatives such as researching and experimenting with innovations for automated software testing and test artifact reuse. Additionally, clients often require Ltesting to deploy mature but costly technologies and methods for testing, imposing restrictions for Ltesting to innovate and improve its service abilities. In future, Ltesting needs to allocate more resources to carry out such initiatives on an ongoing basis to ensure it will remain a forerunner in its field.

4 DISCUSSION

Clients' involvement and commitment to overcome the geographical, technological, cultural, and other sourcing barriers are critical to achieve successful software testing (Riungu, Taipale and Smolander, 2010). While some software defects are caused by coding errors, the most expensive defects are caused by requirement gaps (e.g., unrecognized or misunderstood client requirements) (Kolawa and Huizinga, 2007). All the interviewees agreed that initiation is the most important phase in the life-cycle and affects the other phases of the life-cycle. Requirements analysis and test planning are conducted in this phase. If these activities fail, risks will materialize through breakdowns and service failures will result. In conclusion, initiation is the most important phase and requirements analysis and test planning are the most important practices for providers to control risks.

The interviewees indicated that information systems should enable a seamless and transparent life-cycle from requirements elicitation, analysis, and prioritization through test planning, test case design, and execution to managing, repairing, and verifying defects. Therefore, the class of integrated requirements, test, and defect management systems is the most important class of information systems for testing. Instances of this class enable and are enabled by the effective execution of the eSourcing life-cycle model for testing service providers, helping the providers to meet changing requirements quickly and improve service quality. The QC platform is an example of a commercial system supporting the most common requirements for such a class. However, the experiences from the case study, the literature review (Floss and Tilley 2013; Riungu, Taipale and Smolander, 2010), and our earlier research (Käkölä et al., 2011) indicate that commercial and open source instances of the class for successfully enabling the entire life-cycle are scarcely available. QC is a comprehensive test management tool, but it is far from perfect. For example, the QC platform does not support comprehensive configuration management for all the used and/or created test artifacts. If testers delete a test case by mistake, they need to recreate the test case rather than reuse it from an early version. Configuration management is necessary for all the involved test artifacts during the life-cycle. Some functions of QC are not as good as the respective functions of competing tools.

> "For example, the requirements management tool is not as good as RequisitePro. The version control feature is quite basic compared to CVS or SVN. Therefore, sometimes we need to use more than one tool in the services, including specific tools for specific tasks (e.g., test execution or version management). Then we will transfer the results to QC for comprehensive test management." (A test manager)

The sourcing life-cycle model for testing services focuses on software testing but covers the relevant capability areas and practices of eSCM-SP. It is fully in line with the practices of Ltesting. Ltesting has been able to (1) deliver services directly to end clients through its transparent life-cycle model, (2) accumulate domain knowledge, and (3) communicate with all stakeholders effectively. As a result, it has successfully extended its service scope from the testing of banking software to financial and insurance services. Ltesting thus provides evidence that small and medium-sized providers following the life-cycle model can overcome the limitations of the mediated sourcing model, for example, by extending the scope of their services to relevant domains and by proactively communicating with their clients to deal with defects or changing requirements.

5 CONCLUSIONS AND FUTURE RESEARCH

This research focused on the most important business practices and information systems for providers of software testing services to help providers tap the potential of global testing service provisioning markets. The paper created a comprehensive eSourcing life-cycle model for testing services, enabling providers and clients to manage the life-cycle effectively. The paper validated the model (1) by reviewing the literature on testing as a service (Floss, and Tilley, 2013; Riungu, Taipale and Smolander, 2010; Yu, Tsai, Chen, Liu, Zhao, Tang and Zhao, 2010), the testing process model (Burnstein, Suwanassart and Carlson, 1996; Wang, Qian and Zhu, 2012; Canfora and Di Penta, 2006), and eSourcing services (eSCM-SP, 2010) and (2) by applying the model to evaluate the practices of the case organization. The extant literature does not present similar models for testing eSourcing. The initiation phase proved most important in the life-cycle. It was found that requirements analysis and test planning primarily conducted in the initiation phase are the most important testing practices. The most important class of information systems for testing service providers is the class of requirements, test, and defect management systems.

The generalizability of this research was limited by the deployment of the single case study methodology in the Chinese context. The Quality Center suite used in the case organization may also have biased this research. However, the literature review and interviews indicated that the other commercially available tools are similar to QC. Future research has to investigate the practices and information systems of testing service providers and their clients also in other countries. Providers need not use Quality Center but they may (and are likely to) use competing requirements, test, and/or defect management products. Future research also needs to validate the eSourcing life-cycle model in the context of cloud-based applications because QC is complex for small and medium-sized companies. The companies need to regularly maintain some servers (e.g., database server, web server and application server) locally. The model is partially based on the experiences of the case organization that transferred the applications to be tested and related resources to its own test environment and stored them in local databases. Testing in the cloud environment does not require such transfers to local databases. The model may thus benefit from minor enhancements to make it fully generalizable to the cloud.

References

Adrion, W. R., Branstad, M. A. and Cherniavsky, J. C., (1982). Validation, Verification, and Testing of Computer Software, ACM Computing Surveys (CSUR), 14(2), 159-192.

Adya, M., Nath, D., Sridhar, V. and Malik, A. (2008). Bringing Global Sourcing into the Classroom: Lessons from an Experiential Software Development project, Communications of the Association for Information Systems (22), 33-48.

Ammann, P. and Offutt, J. (2008). Introduction to Software Testing, Cambridge University Press.
Barthelemy, J. and Geyer, D. (2005). An empirical investigation of IT outsourcing versus quasi-outsourcing in France and Germany, Information & Management, 42(4), 533-542.
Barthelemy, J. (2003). The Seven Deadly Sins of Outsourcing, Academy of Management Executive, 17(2), 87-98.
Beizer, B. (1990). Software Testing Techniques (2nd edition), International Thomson Computer Press: New York NY.
Bertolino, A. (2007). Software Testing Research: Achievements, Challenges, Dreams, Proceedings of Future of Software Engineering, 85-103. IEEE.
Biffl S., Aurum A., Boehm B., Erdogmus H., Grünbacher P. (eds.) (2006). Value-Based Software Engineering, Springer-Verlag, Heidelberg, Germany.
Boehm, B. and Basili, V.R. (2001). Software Defect Reduction Top 10 List, IEEE Computer, 34(1), 135-137.
Brotheron, S.A., Fried, R.T. and Norman, E.S. (2008). Applying the Work Breakdown Structure to the Project Management Lifecycle, PMI Global Congress Proceedings.
Burnstein, I., Suwanassart, T. and Carlson, R. (1996). Developing a Testing Maturity Model for Software Test Process Evaluation and Improvement, Proceedings of International Test Conference, 1996, Washington, DC, 581 - 589. IEEE.
Canfora, G. and Di Penta, M. (2006). Testing services and service-centric systems: challenges and opportunities, IT Professional, 8(2), 10-17.
Carmel, E. and Agarwal, R. (2002). The maturation of offshore sourcing of IT work, MISQ-Executive, 1(2), 65-78.
Carmel, E. and Tjia, P. (2005). Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce, Cambridge University Press, Cambridge.
Chrissis, M.B., Konrad, M. and Shrum, S. (2011). CMMI for Development: Guidelines for Process Integration and Product Improvement, Third Edition. Addison-Wesley.
Conradi, R. and Westfechtel, B. (1998). Version Models for Software Configuration Management, ACM Computing Surveys, 30(2), 232-282.
Dedrick, J., Carmel, E. and Kraemer, K.L. (2011). A Dynamic Model of Offshore Software Development, Journal of Information Technology, 26, 1-15.
Estublier, J. (2000). Software Configuration Management: A Road Map, Proceedings of the Conference on the Future of Software, 279 - 289. ACM.
Floss, B. and Tilley, S. (2013). Software Testing as a Service: An Academic Research Perspective, Proceedings of 7th International Symposium on Service Oriented System Engineering (SOSE), Redwood City, 421-424. IEEE.
Gopal, K., Sivaramakrishnan, K., Krishnan, M.S., Mukhopadhyay, T. (2003). Contracts in Offshore Software Development: An Empirical Analysis, Management Science, 49(12), 1671-1683.
Gonzalez, R., Gasco, J. and Llopis, J. (2006). Information Systems Outsourcing: A Literature Analysis, Information & Management, 43, 821-834.
Hamlet, D. (1994). Foundations of software testing: dependability theory, Proceedings of the second ACM SIGSOFT symposium on Foundations of software engineering, 128-139
He, M., Li, M. S., Wang, Q., Yang, Y. and Ye, K. (2008). An Investigation of Software Development Productivity in China, Making globally distributed software development a success story, Proceedings of the international conference on software process, ICSP 2008, 381-394. Springer.
Iqbal, N. and Qureshi, M. R. J. (2009). Improvement of Key Problems of Software Testing in Quality Assurance, Science International, 21(1), 25-28, Lahore, Pakistan.
Järvenpää, S. and Mao, J. (2008). Operational Capabilities Development in Mediated Offshore Software Services Models, Journal of Information Technology (23), 3-17.
Käkölä, T., Koivulahti-Ojala, M. and Liimatainen, J. (2011). An Information Systems Design Product Theory for Integrated Requirements and Release Management Systems, Journal of Software Maintenance and Evolution: Research and Practice, 23(6), 443-463.

Käkölä, T. and Taalas, A. (2008). Validating the Information Systems Design Theory for Dual Information Systems, Proceedings of the 29th International Conference on Information Systems, Paris, Association for Information Systems.

Karinsalo, M. and Abrahamsson, P. (2004). Software Reuse and the Test Development Process: A Combined Approach, International Conference on Software Reuse (ICSR), LNCS 3107, 59-68.

Koivulahti-Ojala, M. and Käkölä, T. (2010). Framework for Evaluating the Version Management Capabilities of a Class of UML Modeling Tools from the Viewpoint of Multi-site, Multi-partner Product Line Organizations, Proceedings of the 43rd Hawaii International Conference on Systems Sciences, IEEE Computer Society, Hawaii, USA.

Kolawa, A. and Huizinga, D. (2007). Automated Defect Prevention: Best Practices in Software Management, Wiley-IEEE Computer Society Press, 41-43, 86.

Koomen, T. and Pol, M. (1999). Test Process Improvement: A Practical Step-by-step Guide to Structured Testing, Addison-Wesley, Great Britain.

Lacity, M. and Rudramuniyaiah, P.S. (2009). Funny Business: Public Opinion of Outsourcing and Offshoring as Reflected in U.S. and Indian Political Cartoons, Communications of the Association for Information Systems , 24(13), 199-224.

Lacity, M., Willcocks, L. and Rottman, J. (2008). Global Outsourcing of Back Office Services: Lessons, Trends, and Enduring Challenges, Strategic Outsourcing: An International Journal, 1(1), 13-34.

Lee, J.N. (2006). Outsourcing Alignment with Business Strategy and Firm Performance, Communications of the Association for Information Systems, 17(49), 1124-1146.

Li, Q., Yang, Y., Li, M.S., Wang, Q., Boehm, B. W. and Hu, C.Y. (2012). Improving Software Testing Process: Feature Prioritization to Make Winners of Success-critical Stakeholders, Journal of Software Maintenance and Evolution: Research and Practice, 24(7), 783-801.

Ltesting (2014). International Leading Software Testing Science and Technology Co., Ltd. http://www.ltesting.com.cn/.

Ma, J. Q., Li, J.Y., Chen, W.B., Conradi, R., Ji, J.Z. and Liu, C.N. (2008). A State of the Practice Study on Communication and Coordination Between Chinese Software Suppliers and Their Global Outsourcers, Software Process Improvement and Practice, 13(3), 233-247.

McCarthy, J.C., Martorelli, B., Moore, S., Agosta, L., and Ross, C.F. (2004). Offshore Outsourcing: The Complete Guide, Forrester Collection, September 7.

Myers, G. J. (2004). The Art of Software Testing (second edition), John Wiley & Sons.

Naik, K. and Tripathy. P. (2008). Software Testing and Quality Assurance: Theory and Practice, John Wiley & Sons, Inc.

Orlikowski, W. J. (2002). Knowing in Practice: Enacting a Collective Capability in Distributed Organizing, Organization Science, 13(3), 249-273.

Paulk, M.C. (2009). A History of the Capability Maturity Model for Software, The Software Quality Professional, 12(1), 5-19. American Society for Quality.

Poston, R.S., Simon, J.C. and Jain, R. (2010). Client Communication Practices in Managing Relationships with Offshore Vendors of Software Testing Services, Communications of the Association for Information Systems, 27, 9, 129-148.

QC (2009). HP Quality Center Software Version (10.00) User's Guide. Hewlett-Packard Development Company.

Ramler, R., Biffl, S. and Grunbacher, P. (2005). Value-based Management of Software Testing, Value-based Software Engineering Springer: Berlin, 226-244.

Riungu, L. M., Taipale, O. and Smolander, K. (2010). Software Testing as an Online Service: Observations from Practice, Software Testing, Verification, and Validation Workshops (ICSTW), 2010 Third international conference, Paris.

Rottman, J., and Lacity, M. (2004). Twenty Practices for Offshore Sourcing, MIS Quarterly Executive, 3(3), 117-131.

Sanz, A., Garcia, J., Saldana, J. and Amescua, A. (2009). A Proposal of a Process Model to Create a Test Factory, Proceeding WOSQ 09 Proceedings of the Seventh ICSE conference on Software Quality IEEE, 65-70.

Schwarz, A., Kalika, M., Kefi, H. and Schwarz, C. (2010) A Dynamic Capabilities Approach to Understanding the Impact of IT-Enabled Businesses Processes and IT-Business Alignment on the Strategic and Operational Performance of the Firm, Communications of the Association for Information Systems, 26(4), 57-84.

TestingGeek, Regression Testing, (2010). http://www.testinggeek.com/index.php/testing-types/testing-purpose/104-regression-testing.

The eSourcing Capability Model for Service Providers (eSCM-SP) V2.01 (2010). http://www.itsqc.org/index.html.

Van Veenendaal, E. (2012). Test Maturity Model integration (TMMi) Release 1.0, TMMi Foundation, Ireland.

Wang, P., Qian X. and Zhu, J. (2012). Testing Process Model and Classification of Test Methods for Embedded Software of Electrical Products, Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on, Beijing.

Webb, L. and Laborde, J. (2005). Crafting a Successful Outsourcing Vendor/Client Relationship, Business Process Management Journal, 11(5), 437-443.

William, P. (2006). Effective methods for software testing, third edition, John Wiley & Sons, Inc. New York, NY, USA.

Yu, L., Tsai, W., Chen, X., Liu, L., Zhao, Y., Tang, L. and Zhao, W. (2010). Testing as a Service over Cloud, Proceedings of 5th International Symposium on Service Oriented System Engineering (SOSE), Nanjing, China. IEEE.

Zhang, S., Huang, L.H. and Yu, D.H. (2006). An Analysis of Information Systems Research in Chinese Mainland, Communications of the Association for Information Systems, 17(35), 785-800.

Zhu, H. and Zhou, M. (2006a). Role-Based Collaboration and Its Kernel Mechanisms, IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews, 36(4), 578-589.

Zhu, H. and Zhou, M. (2006b). Supporting Software Development with Roles, IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 36(6), 1110-1123.

III

An Information System Design Product Theory for the Class of Requirements, Test and Defect Management Systems

by

Lu, Y. & Käkölä, T.,

Journal of Information Systems and e-Business Management. (will submit it soon)

IV

An Information System Design Product Theory for Integrated Order, Transportation and Warehouse Management Systems

by

Lu, Y. & Käkölä, T., 2013

In R.H. Sprague Jr. (Ed.), proceedings of the 46th Hawaii International Conference on System Sciences (HICSS). Wailea, Hawaii, IEEE Computer Society, 3717-3726.

Abstract

Globalization, lead time reduction and cost pressures, and other factors make it nontrivial for most companies to devise and manage effective logistics services to meet client requirements. They are better off using the services of specialized third party logistics service providers. These providers compete fiercely and need to develop and improve their services continuously to gain competitive advantage. Integrating the logistics process for improving communication and coordination is the most feasible way for third party logistics providers to deal with these pressures. Yet, the extant literature provides little theoretical guidance for integrating the logistics life-cycle, including ordering, transportation, and warehousing. This paper develops an Information System Design Product Theory for the class of integrated Order, Transportation and Warehouse Management Systems based on a case study and a literature review. The theory helps clients and service providers to acquire and design information systems for designing, managing and controlling transparent and seamless logistics processes that improve service effectiveness.

1. Introduction

Logistics provides organizations with time and space utilities such as the necessary quantity of right cargo in the right place at the right time [2, 11, 18]. As part of the supply chain services, logistics has to be fulfilled efficiently and effectively to meet client requirements [3, 16] such as globalization, lead time reduction, and client orientation [8, 23]. Meeting these requirements with own logistics services is nontrivial for most organizations. They should focus on core business and source logistics processes to specialized logistics service providers to improve their competiveness. Third-party logistics (3PL) providers are companies managing, coordinating, and delivering logistics services for the clients' value chains [8, 16]. The most important benefits clients gain from using 3PL services are the improved expertise in and efficiency of logistics [9, 19].

3PL providers need to establish close relationships with clients and offer comprehensive logistics services to enhance their competitiveness [17]. For example, providing clients with more value-added services (such as sharing the logistics information in a timely manner among business partners for better planning, reducing inventory, and shortening the delivery time) can improve competitiveness. These services and associated benefits can be achieved by designing and deploying integrated logistics information systems [8, 22].

Logistics information systems facilitate logistics management and allow clients to acquire logistics services and conduct business transactions via the internet [16]. Some of the main purposes of logistics information systems integration are to achieve real-time capturing and sharing of key information along the logistics service processes, process unexpected coordination breakdowns quickly, and make logistics decisions [18].

Investments in IT may fail to produce expected benefits unless clients and 3PL providers are willing to share logistics information [6]. Effective logistics can typically be achieved only when companies exchange not only transactional data (e.g., material or product orders) but also strategic logistics information helping stakeholders to make important decisions in their operations [14]. For example, the sales history information helps providers to forecast demand, improving service quality and efficiency. Real-time inventory information helps providers to plan their replenishment and delivery schedules, improving service quality and reducing inventory costs [13, 20]. Effective knowledge sharing necessitates frequent and intense communication between clients and providers and contributes to establishing cooperative relationships between partners. High degrees of symmetrical flows of strategic information between partners are likely to result [10].

A number of studies have demonstrated various benefits from information sharing with logistics partners in terms of inventory management [1, 13, 26, 28], improved agility and flexibility [24], and the reduced bullwhip effect (that causes supply chain participants far from the end consumers to experience greater demand variations than participants close to the end consumers) [4]. Knowledge sharing significantly impacts the planning and delivery of logistics services [29]. Providers' IT capabilities and knowledge sharing affect the integration of logistics systems directly and the organizational performance indirectly [15]. The integration of logistics systems is a crucial way to realize these benefits and gain competitive advantage. Information should flow seamlessly and the 3PL life-cycle should be transparent for clients and providers.

3PL providers most commonly offer warehousing, transportation, and customs brokerage services [12]. Traditional core competencies of 3PL providers also include (1) booking services for air and sea freight forwarding services and (2) preparing tailored documentation [16]. In sum, the most important logistics practices include freight forwarding services, warehousing services, and transportation services. Usually, forwarding services are based on specific order management services. Therefore, order management is also important for the integrated logistics systems.

Most 3PL providers in China are small or medium-sized. They subcontract parts of their operations to other supply chain partners [16]. Furthermore, poor communication channels (e.g., fax, phone, and email) lead to service breakdowns (e.g., service delays, human errors) and high operating costs. Due to these limitations, Chinese 3PL providers are often unable to obtain up-to-date status information from their supply chain partners in real time for making timely decisions.

There are few studies on small and medium-sized 3PL providers in the Chinese logistics market [7, 16]. However, these 3PL providers have become increasingly important, because there has been a trend for 3PL providers to participate in clients' supply chains for providing logistics management [16]. Therefore, it is necessary for small and medium 3PL providers to improve their communication and operations to offer services more effectively.

This paper proposes that integrated Order, Transportation and Warehouse Management Systems (hereafter, OTWMS) should be designed to help the 3PL providers to manage the 3PL life-cycle from the service initiation phase through ordering, transportation execution, and warehouse tracking and management to service closure. Each OTWMS instance leverages repositories to store automated, OTWMS-generated logistics orders; schedule, transportation, warehouse and inventory information; and other relevant information produced during the 3PL life-cycle. OTWMS instances use the repositories to help reuse logistics plans from previous services and manage newly created logistics plans. OTWMS instances also support logistics team members' work and help the clients capture and track the logistics service information. Clients and 3PL providers can obtain real time information from OTWMS instances, making the 3PL life-cycle transparent and seamless.

To achieve such integration, there are several factors to consider. Client requirements need to be transformed to a detailed and executable logistics plan. Specific transportation management and warehouse management services must meet client requirements and deal with unexpected service breakdowns, necessitating quick but effective creation and sharing of knowledge within and between clients and 3PL providers. Clients and 3PL providers may be globally distributed, making it difficult to conduct face to face meetings with clients. Changing client requirements or service breakdowns may require revising contracts between clients and 3PL providers and raise service risks.

There is little theory-based guidance to help 3PL providers and clients design and leverage such integrated systems. This paper draws upon a case study to create an information system design product theory (hereafter, design product theory) for the class of OTWMS. A complete information system design theory (ISDT) prescribes both the product and process aspects of a class of information systems, that is, *what* are the meta-requirements and the meta-design for all the products instantiating the class; which kernel theories from reference disciplines are vital to determine what the products should do, and *how* the products should be built [27]. ISDTs make the development of products more tractable for application developers by focusing their attention and restricting their options and help organizations to source products and components from commercial and open source markets.

This paper focuses on prescribing the product aspects for the class of OTWMS because the existing literature does not provide such a theory. Moreover, OTWMS instances can be built in many ways and it is thus not as fruitful to prescribe the process aspects as the product aspects. The design product theory addresses the following research question: What are the meta-requirements and the meta-design of the design product theory for the class of OTWMS in order to enable comprehensive 3PL life-cycle management?

This paper is organized as follows. Section 2 presents the research methodology and the case company. Section 3 presents the meta-requirements for OTWMS, that is, the specific practices and involved stakeholders and tasks in each phase of the 3PL life-cycle that must be supported by the OTWMS instances. Section 4 describes the meta-design for the class of OTWMS. The last section concludes the paper and suggests topics for future research.

2. Research methodology, kernel theory, and the case company

This research was conducted in the context of the Chinese 3PL logistics services market offering services for international and domestic clients that leverage information and communication technology enabled sourcing (eSourcing) of logistics services. The research is part of a larger project that investigates various classes of information systems facilitating the end-to-end eSourcing service provisioning life-cycle for markets such as software products and services eSourcing and business process eSourcing.

The eSourcing Capability Model for Service Providers (eSCM-SP) has been chosen as the reference theory of the larger research project to understand the eSourcing life-cycle holistically from providers' viewpoint and to ensure the comparability of various subprojects. eSCM-SP has been demonstrated to help various types of providers in different industries to improve their capabilities related to both ongoing, phase-specific, and engagement-specific sourcing practices throughout the eSourcing service provisioning life-cycle [25]. The life-cycle involves three phases from the provider's viewpoint: initiation, delivery, and completion. Ongoing practices are run throughout the life-cycle to perform management functions. The three phases and the ongoing practices cover ten capability areas (e.g., knowledge management, threat management, and performance management).

3PL services are widely delivered by Chinese providers. This research investigates the end-to-end eSourcing service provisioning life-cycle (hereafter "life-cycle") using a single qualitative case study to provide a holistic, systemic understanding of the phenomenon [5]. The case company was selected because it is a leader in 3PL business in China and its practices are likely to create an adequate baseline for design product theory creation. In 2012, it had more than 2200 employees and offices in more than 80 cities. The collected data covers the 3PL life-cycle, including the practices, the cooperation between logistics team members involved in these practices, the people responsible for logistics engagements as a whole, and the design and use of information systems that structure and are structured by these practices. Practices of the case company were compared to eSCM-SP based on the three phases and specific practices.

The first author conducted two rounds of investigation in the case company. A subsidiary of the corporate entity was investigated on-site in Beijing. It delivers logistics services based on the Total Order Management (TOM) information systems instance developed by the case company. The instance includes three main subsystems: Order Management System (OMS), Warehouse Management System (SMS), and Transportation Management System (TMS), enabling the offering of an integrated solution for clients' supply chain management. They support service delivery and proactively help to avoid service breakdowns in order to meet client requirements. SMS offers basic warehouse management functionality. To meet complex requirements, the case company bought another Warehouse Management System (WMS).

"SMS just has the basic functions of warehouse management. We have developed it ourselves to meet the general warehousing requirements. However, our business is growing and large international enterprises demand more complex warehousing services, so we bought the WMS. We will choose either SMS or WMS based on the client's warehousing requirements. SMS and WMS are also able to share information during the logistics services." - A warehouse manager

The investigation developed an understanding about (1) ways the workers in the company use the subsystems for providing logistics services and (2) the types of breakdowns occurring. The second round of data collection was conducted in December 2011 and January 2012 by visiting the headquarters of the company on-site in Guangzhou to probe how TOM's subsystems supported each service phase of eSCM-SP. For example, TOM can combine and optimize orders from clients. The introduction of TOM in 2002 enabled the company to reduce its logistics costs by 20% and the logistics costs of its clients by 2-4% and to improve service effectiveness by raising order processing speed from 50 orders per hour to 150 orders per hour.

Logistics teams are responsible for specific logistics services. Usually the most important services are delivered by three sub-teams: order management team, transportation management team, and warehouse management team (Table 1). These teams need to work together seamlessly to process orders, analyze client requirements, and draft logistics plans. In order to trace the service engagements initiated by orders, the 3PL life-cycle should be transparent for clients.

The design product theory for OTWMS has been created based on a literature review and the analysis of the design and use of the innovative TOM system of the case company. eSCM-SP has been chosen as the kernel theory of the design product theory because it takes the life-cycle view and the OTWMS instances must support the 3PL life-cycle.

The design product theory for OTWMS has been designed to be abstract and generic enough so 3PL providers can use it to improve their processes and information systems regardless of their current practices and systems. It may be possible for 3PL providers to benefit from the theory for OTWMS even without replacing any existing systems. Providers can thus use even separate order management systems, transportation management systems, and warehouse management systems and use the theory to better integrate and organize these systems for enabling the end-to-end life-cycle. For example, an OTWMS instance can track the order execution process against the logistics plan and report execution results and breakdowns. It does not need to help execute specific logistics tasks but it needs to trace and report the results of the tasks. A specific task can be run by using other logistics management tools. Therefore, the analysis of the practices and information systems of the case company has helped us to scope the design product theory for OTWMS appropriately.

Title	Responsibility
Order management team	Team is responsible for order validation, entry, and processing. It makes the logistics plan for the order together with a transportation team and a warehouse management team. During the delivery phase, it also coordinates with the transportation and warehouse management teams. It is solely responsible for communicating with clients. During order completion, it documents the services and the lessons learnt.
Transportation management team	Team defines the most efficient transportation schemes according to client requirements and drafts the logistics plan with the order management team. It executes the delivery service and reports any transportation breakdowns (e.g., delays, accidents, and non-forecasted stops).
Warehouse management team	Team uses SMS or WMS to manage the cargo and support the transportation team (e.g., ensuring the cargo is ready for delivery when the transportation team arrives).

Table 1. Responsibilities of Key Logistics Teams

3. Meta-requirements of the design theory for OTWMS

This section describes the meta-requirements for the design product theory of OTWMS, that is, what services integrated order, transportation, and warehouse management systems must provide to enable stakeholders to streamline the end-to-end 3PL life-cycle (Figure 1). OTWMS shall offer three categories of services: (1) order management, (2) transportation management, and (3) warehouse management (Table 2). Order management deals with, for example, order prioritization and management. Prioritization refers to establishing priorities for orders according to client requirements (e.g., lowest transportation costs, fewest possible stops to improve quality and to shorten lead-times). Order management is responsible for a variety of issues such as arranging interrelated orders (e.g., having the same final destinations and priorities) to optimize the resource usage of transportation services, informing clients on product availability and order status, and enabling clients to trace cargo information. For example, whenever breakdowns occur during warehouse management or transportation execution in the case company, the responsible teams must inform the order management team that will then communicate with the client(s) as necessary. OTWMS instances record breakdowns and send information to stakeholders. Warehouse management is responsible for the goods in warehouses, for optimizing the usage of the warehouse space, and for processing transactions such as receiving and picking goods. Transportation management is responsible for delivering cargo based on orders, client requirements, and the locations of ordered items in available warehouses. The most efficient transportation schemes are determined based on the orders' priorities and the availability information provided by warehouse management.

Order management	Transportation management	Warehouse management
1 Prioritize an order based on business priorities and risks 2 Collect orders from previous similar services to reduce duplication 3 Manage interdependencies between orders and align order, transportation, and warehouse teams. 4 Provide clients with order status.	1 Monitor the transportation progress against the logistics plan 2 Report execution results and transportation breakdowns (e.g., delays, accidents and unexpected stops) 3 Generate a transportation results report	Identify storage locations of cargo, track cargo from initial phase to completion phase, and report real time information about the status of cargo

Table 2. A framework for categorizing the services of the design product theory for OTWMS

3.1. Initiation

In the initiation phase, a client provides a request for proposal (RFP) and documented logistics requirements, including specific transportation requirements, warehousing requirements, expected arrival time, expected number (quantity), types, volumes (to determine the physical spaces needed during transportation and warehousing), and destinations of the cargo to be delivered. The 3PL providers need to analyse the proposal and requirements to create a business case for estimating the profitability of the service. If the service is profitable, the providers will draft logistic plans to bid. Clients will analyse the plans, choose the providers, sign the contracts, and send formal orders to selected providers. Providers will establish unique identifiers for the orders to trace and manage the orders during the end-to-end life-cycle. Whenever clients have special requirements, their orders will be marked as special orders. For example, in the case company, the orders that must be delivered to clients after normal working hours are considered special orders.

Order analysis and management are the main activities throughout the 3PL life-cycle. The order database enables an OTWMS instance to manage multiple order types and provides real-time visibility of order status and associated transportation and warehouse information to evaluate service quality and business risks. Furthermore, bidirectional traceability is supported between order, transportation, and warehouse information artifacts across the 3PL life-cycle.

3.1.1. Prioritizing the orders based on business priorities

Providers will analyze the RFP and client requirements. The involved teams need to work together to analyze the priorities of logistics requirements, check the availability of appropriate warehouse space (if necessary), and define the most efficient transportation schemes. The OTWMS instance helps them prioritize orders based on business priorities and risks.

3.1.2. Collect and analyze orders from previous service engagements to reduce duplication

The OTWMS instance collects order, logistics plan, transportation, warehouse, and other relevant artifacts from each engagement between a provider and a client. Providers analyze the client requirements and search for relevant artifacts to reduce the time and costs involved in creating a suitable logistics plan and to estimate the service price and the needed time and resources based on previous experiences. After that, they draft the logistics plan and bid.

Clients review the bids, including logistics plans, from several providers and select the proper providers. After contracting has been completed, clients will send formal orders to the providers who will schedule, resource, and execute the transportation and warehousing services.

3.1.3. Assign order to the suitable transportation team and prepare the delivery services

After the order management team receives the formal order, it needs to validate the order to ensure all the required information is included, schedule the service, send the order to the proper transportation team to execute the services, and inform the warehouse management team to prepare for offering storage or picking services. After that, the transportation team knows the destination of the cargo, checks the availability of necessary vehicle(s) and other resources, and prepares to deliver services. In the case company, OTWMS instances are used to assign people for the specific service delivery, making it easy for relevant stakeholders to hold these people accountable and to communicate with them in real time. When transportation teams need to pick cargo from clients, the order management team communicates with clients first to make appointments and then informs transportation teams the exact times and places for picking the cargo. Order management teams also communicate with clients whenever breakdowns occur.

3.2. Delivery

During the delivery phase, transportation teams are responsible for preparing and executing specific transportation services. Clients and providers can trace the transportation process and get the cargo status information timely through OTWMS. The process includes various stakeholders. The execution of the process involves coordination of the flow of information, services and finances among these stakeholders [21]. Therefore, it is important to manage the life-cycle effectively to meet the delivery performance expectations of the stakeholders.

OTWMS enables clients and providers to communicate effectively and seamlessly throughout the delivery phase. For example, logistics information such as order status and inventory reports can be shared with clients and providers to reduce the inventory and speed up the overall fulfillment process. In addition, OTWMS enables clients and providers to deal with breakdowns quickly. For example, when a transportation team of the case company arrives on time to pick cargo from a client, but the client is not ready, the team must inform the order management team responsible for dealing with the breakdown together with the client. Whenever transportation teams notice discrepancies between the cargos and the original orders, they must also inform the order management team about such breakdowns. OTWMS instances record all the breakdowns to help both clients and providers to improve their performance in future.

3.2.1. Transportation execution

OTWMS must support all physical and administrative operations regarding transportation. For example, it can trace each delivery event by event (shipping from A, arrival at B, customs clearance, etc.) and send transportation breakdown (e.g., delay or accident) reports and real time transportation status information to order management teams. Whenever clients refuse to receive the cargo due to breakdowns, the respective orders may have to be refunded. When the providers are not responsible for the breakdowns, transportation teams need to inform order management teams responsible for communicating with clients. OTWMS must also collect metrics to measure the transportation performance such as transport cost per distance (e.g., per mile or kilometer) and carrier rate acceptance.

3.2.2. Warehouse management

Warehouse management primarily aims at controlling the movement and storage of materials within a warehouse and processing the related transactions, including shipping, receiving, storing and picking. Warehouse management monitors the flow of cargo and optimizes stock based on real time information provided by technologies such as barcode scanners, mobile computers, and radio frequency identification. Once data has been collected, it is typically transmitted to a central database through either asynchronous batch processing or real time wireless transmission. OTWMS uses the database to provide useful reports about the status of goods in the warehouse for order management teams and clients.

Warehouse management teams prepare for service after providers receive the formal orders. They check numbers, types, required spaces, and arrival times of cargos from orders. In the case company, they also check the arrived cargos and the associated information together with the transportation teams. The validated information is stored in OTWMS.

3.2.3. Service breakdowns

Whenever service breakdowns happen, OTWMS will send relevant information to order management teams that need to communicate with clients and adjust transportation schemes as necessary. Clients will estimate the influence of breakdowns and decide whether to change their requirements. If they change requirements, they typically have to form and send new orders to providers. Service breakdowns can be caused by clients or providers. For example, if the client is not ready to receive cargo when the transportation team of the case company arrives, the transportation team reports the breakdown to the order management team that will communicate with clients and arrange redelivery or other solution. In the case company, stakeholders can generally obtain clear instructions for proceeding with the delivery within an hour after reporting breakdowns to the order management team.

"Without the help of OTWMS, it would be impossible to deal with breakdowns in one hour. This system helps us communicate with all stakeholders including clients in real time, which is crucial to deal with breakdowns effectively." - An order manager

3.3. Completion

The completion phase starts when providers have transported cargo to final destinations. Clients need to check the services and cargos to determine whether they meet the service closure conditions. If the conditions are met, the logistics engagement between the client and the provider can be closed. The client needs to pay for the services according to the original agreement and realized service quality (e.g., taking into account integrity and possible delays). When the provider's financial department will receive the payment, the provider can close the order, summarize the services, and compare the performance during the engagement with earlier measurements to improve their service capabilities. For example, OTWMS should be able to benchmark the transportation performance with industry standards and previous performances and report the results to transportation management teams and other stakeholders.

OTWMS stores relevant information and artifacts to the order database for further reuse. The artifacts to be stored include logistics requirements and optimized logistics plans and schedules, which can help the provider to improve their transportation and warehouse management in future.

1. A meta-design of the design product theory for OTWMS
2.

This section outlines a generic meta-design for OTWMS based on the analyses of interview transcripts, the 3PL life-cycle, the literature review, and the OTWMS instance developed by the case company. The meta-design covers the entire 3PL life-cycle outlined in Section 3 and visualized in Figure 1. The section concludes by explicating the linkages between order management, transportation management, and warehouse management subsystems to validate the meta-design and to justify its scope.

Order, transportation, and warehouse artifacts are managed by OTWMS instances. Orders are associated with the transportation and warehouse management artifacts. The relationships between these artifacts are explained next. Orders are based on client requirements and RFPs; each order needs at least one transportation service and zero or more warehousing services to complete it; each warehouse links with at least one order; and each transportation service delivers at least one order.

This section introduces generic structures and attributes of the three classes of artifacts presented above. According to the design product theory, OTWMS instances should include at least these structures and attributes to be effective.

4.1. Order

Table 3 presents the generic structure of order artifacts. In the following, each class within the structure is explained.

Description describes what an order is about, the purpose of the order, and the deadline for its delivery. If there are service breakdowns that lead to change requirements, clients may form a new order to execute the services. *Name* and *ID* are used for identification and traceability.

Origin describes the client requirements the order is based on. One order should cover all the client requirements.

Class	Questions	Attributes
Description	What is the order about?	Name, ID, Description, Required date and time of delivery, Rationale
Origin	Which client requirements does the order refer to?	Author, Source, Date of creation
Analysis	What are the implications of the order?	Status, Required effort, Priority, Scheduled date and time of delivery
Workflow	What should be done to this order next? By whom?	Assigned Transportation services, Assigned Warehousing services, Responsible person, Realized order closure date and time
History	What has been done to the order artifact? When?	Information about all prior edits, editors, and changes

Table 3. Generic Structure of Order Artifact

Analysis is used to probe the implications of the order. *Priority* is used to rank orders and arrange suitable resources and efforts. During the service delivery phase, *status* can be used to check the order status (e.g., shipping or waiting for picking).

Workflow describes what should be done next to this order and by whom. Order management teams need to allocate each order to one or more transportation and warehouse management services.

History is used to provide information about the responsible managers and all prior edits of various order attributes. As a result, the stakeholders can be held accountable for their actions and unexpected service breakdowns can be dealt with effectively. Changed requirements may necessitate unexpected revisions of logistics plans and raise service risks. History information helps logistics teams to proactively eliminate many breakdowns and to recover from breakdowns to continue service execution.

4.2. Transportation

Table 4 presents the generic structure of transportation artifacts.

Description describes the destination of transported goods. *Route* indicates the stops needed for the transportation service, including the final destination.

Origin describes the order(s) the transportation service refers to. One order may need more than one transportation service.

Analysis is used to probe the implications of a transportation service. *Priority* describes the priority of the transportation service and *status* refers to the transportation progress. Required effort describes the transportation costs, time, and resources, which can be used to calculate the total service cost of an engagement. This information can be reused to estimate the profitability and feasibility of future engagements.

Class	Questions	Attributes
Description	What is the transportation service about?	Name, ID, Description, Rationale, Route, Required date and time of delivery
Origin	Which order does the transportation service refer to?	Author, Source order, Date of execution
Analysis	What are the implications of the transportation service?	Status, Required effort, Priority, Scheduled date and time of delivery
Related transportation vehicles	Which vehicles and transportation methods are involved in this transportation service?	IDs of vehicles to be used
Workflow	What should be done to this transportation service next? By whom?	Allocation to transportation team members, Responsible person, Realized date and time of delivery
History	What has been done to this transportation artifact? When?	Information about all prior edits, editors, and changes

Table 4. Generic Structure of Transportation Artifact

Related vehicles provides traceability links to the vehicles (e.g., cars, ships) that are involved in the transportation services.

Workflow describes who is responsible for transportation processes and the realized time of delivery.

4.3. Warehouse

Table 5 presents the generic structure of warehouse artifacts. *Description* explains which logistics requirements are met by the warehouse.

Origin describes which orders are linked with the warehouse, when, and using which transportation services.

Analysis probes the availability and convenience (e.g., distance from the final destination) of the warehouse and the profitability of using it in order to prioritize and schedule the transportation service. This information can also be reused in future to help providers and clients optimize schedules and improve service effectiveness.

Workflow describes the processes that should be taken to use the warehouse and the responsible stakeholders. The transportation team needs to communicate with the warehouse team and form a reasonable store and pick schedule.

Class	Questions	Attributes
Description	Which logistics requirements does the warehouse meet?	Name, ID, Description, Rationale
Origin	Which orders are linked with the warehouse? Which transportation artifacts are linked with the warehouse?	Author, Source order, Source transportation, Scheduled dates and times of transportation
Analysis	What are the availability and convenience of the warehouse?	Status, Priority, Risk, Required warehouse space
Workflow	What should be done to use this warehouse next? By whom?	Allocation to transportation team and warehouse management team, Information about responsible person
History	What has been done to this warehouse artifact? When?	Information about all prior edits, editors, and changes

Table 5. Generic Structure of Warehouse Artifact

4.4. Validating and scoping the design product theory for OTWMS

OTWMS instances help 3PL providers to prioritize and valuate logistics requirements. The requirements and their interdependencies can be stored in the order database. The prioritization and valuation methods are beyond the scope of this paper.

Most logistics resources are allocated to deal with the highest priority requirements. OTWMS instances make it easy to trace orders and transportation services, because transportation and warehouse management artifacts are bidirectionally linked to orders. In the case company, all the stakeholders involved in the service engagement record their actions and relevant information to OTWMS, helping the order management team to manage service provisioning and to effectively communicate with clients.

For each logistics service engagement between a client and a provider, the used OTWMS instance indicates which orders provide the purpose for the transportation and which warehouses are leveraged. Moreover, all actions can be traced to responsible staff members, improving accountability and service quality. Version management helps to identify the stakeholders involved with different versions of various artifacts and the actions the stakeholders have taken. Therefore, order management teams can analyze and control the impacts of order changes and revise the schedules and logistic plans as necessary to meet the most important requirements and recover from service breakdowns.

3. Conclusions and future research
4.

This research established the meta-requirements and the meta-design of the design product theory for OTWMS. The theory helps 3PL providers design, acquire, and use OTWMS instances for providing logistics services. The validity of the theory was enhanced by a case study, which involved a leading Chinese 3PL provider, and the analysis of an OTWMS instance, which has been developed and used by the case company for about ten years.

OTWMS instances are expected to enforce standardized processes for 3PL providers and the implementation of best practices across services. Order management, transportation management, and warehouse management teams can share and reuse artifacts from the order database to raise productivity and quality. Clients can share logistics knowledge across service engagements to increase efficiency and reduce risks. 3PL providers and clients can aggregate quality metrics across service engagements.

This research is limited to one case study. The case company is not only a leader in its field in China but also the first Chinese company to use modern logistics concepts to provide clients with integrated logistics services. Its key staff members interviewed for this study have more than ten years of work experience in logistics services. The generalizability of the design product theory is thus expected to be significant. Future research needs to validate the theory using more case studies and help providers to better reuse logistics artifacts. New case studies and action research are thus necessary to make the theory even more credible for 3PL providers, information systems designers, logistics professionals, and researchers.

6. References

[1] Cachon, G.P., Fisher, M., 2000, “Supply Chain Inventory Management and the Value of Shared Information”, Management Science, 46(8), pp. 1032-1048.

[2] Caputo, M., Mininno, V., 1998, “Configurations for Logistics Co-ordination: A Survey of Italian Grocery Firms”, International Journal of Physical Distribution and Logistics Management, 28(5), pp. 349-376.

[3] Council of Logistics Management, 2003, “Supply Chain Visions: Logistics Terms and Glossary”, Council of Logistics Management, pp. 74.

[4] Dejonckheere, J., Disney, S.M., Lambrecht, M.R., Towill, D.R., 2004, "The Impact of Information Enrichment on the Bullwhip Effect in Supply Chains: A Control Engineering Perspective", European Journal of Operational Research, 153(3), pp. 727-750.

[5] Eisenhardt, K.M., 1989, "Building Theories from Case Study Research", The Academy of Management Review, 14(4), pp. 532-550.

[6] Fawcett, S.E., Osterhaus, P., Magnan, G.M., Brau, J.C., McCarter, M.W., 2007, "Information Sharing and Supply Chain Performance: The Role of Connectivity and Willingness", Supply Chain Management: An International Journal, 12(5), pp. 358-368.

[7] Gunasekaran, A., and Ngai, E.W.T., 2003, "The Successful Management of a Small Logistics Company", International Journal of Physical Distribution and Logistics Management, 33(9), pp. 825-842.

[8] Hertz, S. and Alfredsson, M., 2003, "Strategic Development of Third-party Logistics Providers", Industrial Marketing Management, 32, pp. 139-149.

[9] Jharkharia, S. and Shankar, R., 2005, "Selection of Logistics Service Provider: An Analytic Network Process (ANP) Approach", The International Journal of Management Science, 35(3), pp. 274-289.

[10] Klein, R., Rai, A., Straub, D.W., 2007, "Competitive and Cooperative Positioning in Supply Chain Logistics Relationships", Decision Science, 38(4), pp. 611-646.

[11] La Londe, B.J., 1983, "A Reconfiguration of Logistics Systems in the 80s: Strategies and Challenges", Journal of Business Logistics, 4, pp.1-11.

[12] Langley, C.J., Allen, G.R., and Colombo, M.J., 2003, Third-party logistics: Results and findings of the 2003 eighth annual study. Georgia Institute of Technology.

[13] Lee, H.L., So, K.C., Tang,C.S., 2000, "The Value of Information Sharing in a Two-level Supply Chain", Management Science, 46(5), pp. 626-643.

[14] Li, S., Ragu-Nathan, B., Ragu-Nathan, T.S., and Rao, S.S., 2006, "The Impact of Supply Chain Management Practices on Competitive Advantage and Organizational Performance", Omega, 34(2), pp. 107-124.

[15] Li, G., Yang, H., Sun, L., and Sohal, A.S., 2009, "The Impact of IT Implementation on Supply Chain Integration and Performance", International Journal of Production Economics, 120(1), pp. 125-128.

[16] Liu, J.J., So, S.C.K, Choy, K.L., Lau, H., and Kwok, S.K., 2008, "Performance Improvement of Third-party Logistics Providers-An Integrated Approach with A Logistics Information System", International Journal of Technology Management, 42(3), pp. 226-249.

[17] Matopoulos, A., Vlachopoulou, M., Manthou, V., Manos, B., 2007, "A Conceptual Framework for Supply Chain Collaboration: Empirical Evidence from the Agri-food Industry", Supply Chain Management: An International Journal, 12(3), pp. 177-186.

[18] Prajogo, D. and Olhager, J., 2012, "Supply Chain Integration and Performance: The Effects of Long-term Relationships, Information Technology and Sharing, and Logistics Integration", International Journal of Production Economics, 135 (1), pp. 514-522.

[19] Razzaque, M.A., Sheng, C.C., 1998, "Outsourcing of Logistics Functions: A Literature Survey", International Journal of Physical Distribution and Logistics Management, 28(2), pp.89-107.

[20] Seidmann, A., Sundararajan, A., 1997, "The Effects of Task and Information Asymmetry on Business Process Redesign", International Journal of Production Economics, 50, pp. 117-128.

[21] Simchi-levi, D., Kaminsky, D., and Simchi-levi, E., 2000, "Designing and Managing the Supply Chain-Concepts, Strategies and Case Studies", McGraw-Hill, pp. 8-11.

[22] Singh, P.J., Power, D., 2009, "The Nature and Effectiveness of Collaboration between Firms, Their Customers and Suppliers: A Supply Chain Perspective", Supply Chain Management: An International Journal, 14(3), pp. 189-200.

[23] Stock, G.N., Greis, N.P., Kasarda, J.D., 2000, "Enterprise Logistics and Supply Chain Structure: The Role of Fit", Journal of Operations Management, 18(5), pp. 531-547.

[24] Swafford, P.M., Ghosh, S., Murthy, N., 2008, "Achieving Supply Chain Agility Through IT Integration and Flexibility", International Journal of Production Economics, 116(2), pp. 288-297.

[25] The eSourcing Capability Model for Service Providers (eSCM-SP) V2.01, 2010, http://www.itsqc.org/index.html.

[26] Yu, Z., Yan, H., Cheng, T.C.E., 2001, "Benefits of Information Sharing with Supply Chain Partnerships", Industrial Management and Data Systems, 101(3), pp. 114-119.

[27] Walls, J.G., Widmeyer, G.R. and EI Sawy, O.A., 1992, "Building an Information System Design Theory for Vigilant EIS", Information Systems Research, 3(1), pp. 36-59.

[28] Zhao, X., Xie, J., Zhang, W.J., 2002, "The Impact of Information Sharing and Ordering Co-ordination on Supply Chain Performance", Supply Chain Management: An International Journal, 7(1), pp. 24-40.

[29] Zhou, H., Benton Jr, W.C., 2007, "Supply Chain Practice and Information Sharing", Journal of Operation Management, 25, pp. 1348-1365.

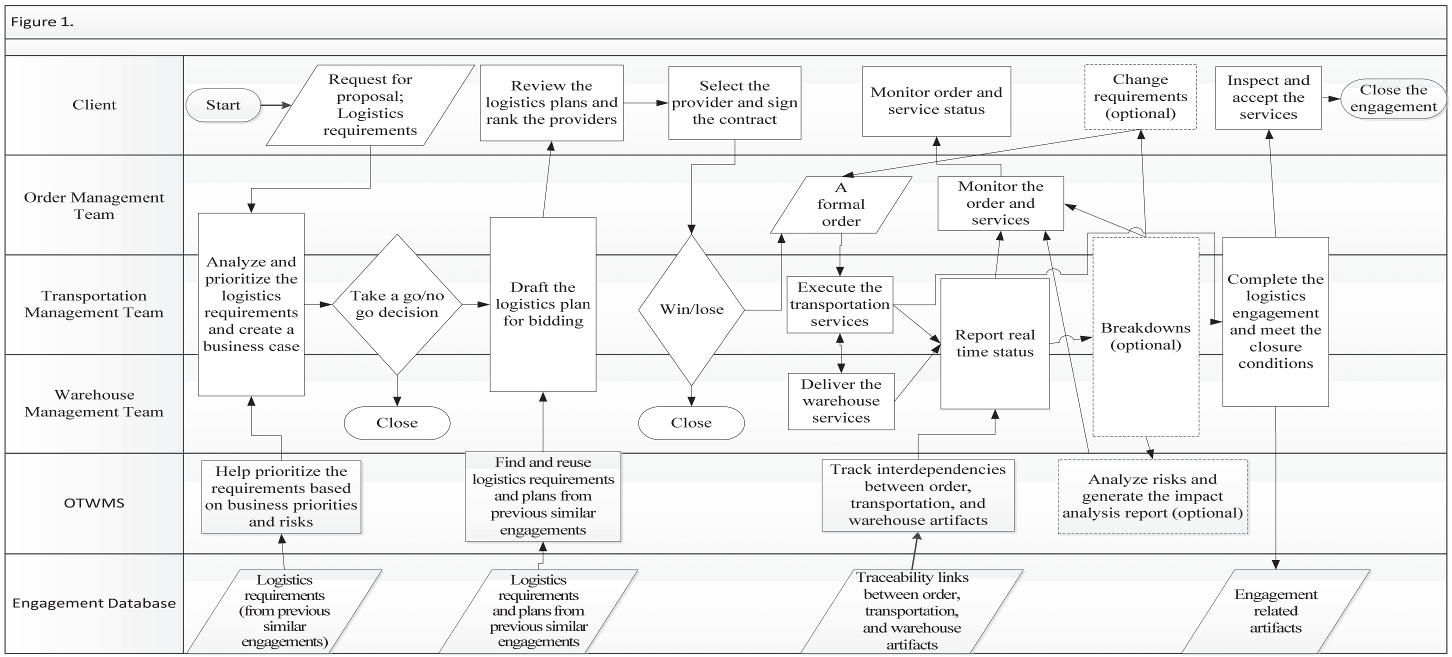

Figure 1. A Meta-Design for Integrated Order, Transportation, and Warehouse Management Systems enabling the Third-party Logistics Services Life-Cycle

V

An Information System Design Product Theory for the Abstract Class of Integrated Requirements and Delivery Management Systems

by

Lu, Y. & Käkölä, T., 2014

In R.H. Sprague Jr. (Ed.), proceedings of the 47th Hawaii International Conference on System Sciences (HICSS). Waikoloa, Hawaii, IEEE Computer Society, 3677-3686.

Abstract

Information and Communications Technology-enabled international sourcing of software-intensive systems and services (eSourcing) is increasingly used as a means of adding value, reducing costs, sharing risks, and achieving strategic aims. To maximally reap the benefits from eSourcing and to mitigate the risks, providers and clients have to be aware of and build capabilities for the eSourcing life-cycle. China is in a position to become a superpower for eSourcing service provisioning, but most Chinese eSourcing service providers are small or medium-sized and typically work for larger intermediaries instead of end-clients, limiting their business and capabilities development. The extant literature does not offer a comprehensive enough guidance for eSourcing life-cycle management to overcome this limitation. This paper presents an information system design product theory for the abstract class of Requirements and Delivery Management Systems. eSourcing service providers can use it to establish domain-specific design product theories and to instantiate them into information systems that support the design, service provisioning, and breakdown recovery within the eSourcing life-cycle.

1. Introduction

International sourcing is used worldwide to gain comparative cost advantage and outside expertise, to improve services, and to gain access to technology [1]. More than 50% of American Fortune 500 firms and an increasing proportion of Western European and Japanese firms are users of offshore software sourcing [4]. India, Russia, Philippines and China are the important nations for service provisioning [5,6]. The use of Information and Communications Technology (ICT) is crucial in international sourcing.

The offshoring of services is critically dependent on a supply of providers that have operational and strategic capabilities to offer comparative cost advantage, satisfactory quality, and on time delivery despite the differences in time zones and culture [3]. There is already a myriad of Chinese eSourcing service providers in the global market and their role is increasing quickly. Yet, there is relatively little research focusing on the Chinese eSourcing service provisioning industry.

The purpose of this paper is to report the status of the core deliverables of the doctoral dissertation project (hereafter, "project") of the first author and to invite criticism and feedback to improve the quality of the resulting dissertation. The project focuses on Chinese providers, but it aims at reaching results generalizable to providers in other nations with powerful eSourcing industries.

Language and time zone issues do not hinder Chinese providers from entering the western eSourcing markets [18]. Chinese providers need to improve mainly their business development and process capabilities and enabling information systems. Most importantly, Chinese client organizations must further develop their eSourcing cultures so Chinese service providers can have large and demanding domestic markets that stimulate and provide financing for the design and productization of innovative and competitive services and products [21]. Therefore, comprehensive advice is needed to help service providers to focus on the most value adding business strategies, eSourcing life-cycle phases, activities, and enabling classes of information systems that best improve their capabilities for service design and provisioning.

The extant literature does not offer a comprehensive enough guidance for eSourcing management in the Chinese context. It focuses on clients from the US and Europe [13]. Providers' perspective has not been studied sufficiently [2,7]. American and European enterprises are familiar with the large Indian companies and their sourcing services. The global sourcing research is also limited to the service model Indian providers use mainly with their American clients [11,20]. The lessons learnt are not necessarily applicable to Chinese providers. Most Chinese providers leverage the mediated offshore outsourcing business model, whereby a small or a medium-sized Chinese provider delivers offshore software services to a larger foreign ICT client that contracts and interfaces with the actual end-clients onshore [10]. This business model usually restricts the providers to small, low-value projects and hampers the sharing of knowledge with end-clients, severely impeding the capability and business development of Chinese providers. The extant literature does not extensively address this business model and ways to overcome its limitations. Järvenpää and Mao [10] focus on the development of client-specific, process, and human resource capabilities, but their research does not cover the entire eSourcing life-cycle and cannot enhance providers' capabilities comprehensively.

eSourcing can be divided into two categories: ICT services sourcing (ICTS) and business process sourcing (BPS). ICT sourcing occurs when an organization contracts one or more providers to perform an ICT function instead of performing the function itself. The provider can be a third party, another division, or a subsidiary of a single corporate entity [9]. BPS involves the sourcing of noncore ICT-enabled business processes to internal or external providers. It enables clients to focus on their primary business operations and to achieve lower costs, improved productivity, and more flexible staffing options [19].

This project studies one service provider in each of the ICTS and BPS categories using the case study research methodology. It aims at providing as generalizable scientific knowledge as possible concerning the most important business practices, activities, and classes of information systems for eSourcing (1) ICT services and (2) business processes. The project investigates the two contexts, respectively, through the following two eSourcing project domains in the Chinese eSourcing market: (1) software testing services and (2) third-party logistics (3PL) services. The two research cases are Ltesting (http://www.osourcing.net/) and PG Logistics (PGL, http://www.pgl-world.cn/).

Ltesting is a medium-sized (less than 50 employees) professional software testing services provider [17]. It has been chosen for this project because it has established a leading position in the Chinese testing service market based on its testing experiences from multiple domains (e.g., banking, insurance and telecommunications) and professional services. There are two reasons for selecting PG Logistics (PGL) as the second research case. First, PGL is the most influential third-party logistics enterprise in the Chinese market and the first Chinese company to use modern logistics concepts to provide integrated logistics services. Second, PGL has developed its own flexible and scalable third-party logistics information integration platform.

Extensive communication, coordination, and collaboration are required between the stakeholders involved in the eSourcing life-cycle. Numerous information systems are typically needed. Without adequate integration of these information systems, information quality will deteriorate, leading to potentially expensive breakdowns disrupting services and reducing service effectiveness. This project creates information system design *product* theories (hereafter, "design product theory") that help providers to design and integrate the most crucial classes of systems that support the end-to-end eSourcing life-cycle. A complete information system design theory (ISDT) prescribes both the product and process aspects of a class of information systems, that is, *what* are the meta-requirements and the meta-design for all the products instantiating the class; which kernel theories from reference disciplines are vital to determine what all the products should do, and *how* the products should be built [23]. ISDTs make the development of products more tractable for application developers by focusing their attention and restricting their options and help organizations to source products from commercial and open source markets. The project focuses on information system design product theories prescribing the product aspects for the classes of systems because system instances can often be built in many ways and it is thus not as fruitful to prescribe the process aspects as the product aspects.

In the context of software testing services, most commercially available information systems support the eSourcing life-cycle only from the providers' viewpoint and provide limited support for clients. They are used separately for requirements management, test execution management, or defect management. Standardized data transfer between the different information systems supporting specific life-cycle phases is difficult, reducing service effectiveness and raising the risks of failure. This research project has developed an information system design product theory for the class of Requirements, Test and Defect Management Systems (RTDMS) to support knowledge management throughout the eSourcing life-cycle for testing services [15].

In the context of third-party logistics services, there are few studies on small and medium-sized 3PL providers in the Chinese logistics market [8,14]. 3PL providers have become increasingly important because they participate in clients' supply chains for providing logistics management [14]. Therefore, it is necessary for small- and medium-sized 3PL providers to improve their communication and operations to offer services more effectively. This research project has developed an information system design product theory for the class of Order, Transportation and Warehouse Management Systems (OTWMS) to help clients and service providers to design, execute, manage, and control transparent and seamless logistics processes [16].

The project draws upon the two design product theories and the cross-case analysis to create an information system design product theory for the class of Requirements and Delivery Management Systems (RDMS). It helps both ICT services sourcing and business process sourcing service providers to design information systems for managing the end-to-end eSourcing life-cycle.

It is important to establish and execute efficient business models and processes throughout the eSourcing life-cycle and to recover from unanticipated coordination breakdowns quickly and effectively [12]. By analyzing breakdowns and their underlying causes, researchers and actors in the workplace can identify the problems that are not easily visible in normal routines and create new knowledge to solve such problems. Redesigning the eSourcing life-cycle when necessary ensures organizational survival, proactive elimination of some breakdowns, and effective long-term enactment of routines [12].

The project addresses the research question: which eSourcing practices, associated activities, and enabling classes of information systems are the highest priority ones for service providers from the viewpoint of executing the eSourcing life-cycle, recovering from coordination breakdowns during execution, and redesigning the life-cycle practices, activities, and systems to ensure organizational long-term effectiveness [12]? To answer this question, eSourcing practices, associated activities, and enabling classes of information systems are analyzed holistically as work systems.

This paper is organized as follows. Section 2 presents the research methodology and the kernel theory. Section 3 presents the meta-requirements for RDMS, that is, the practices and involved stakeholders in each phase of the eSourcing life-cycle that must be supported by the RDMS instances. Section 4 describes the meta-design of the design product theory for RDMS and validates it based on RTDMS and OTWMS theories. The last section concludes the paper and suggests topics for future research.

2. Research methodology and kernel theory

This project classifies eSourcing practices into ICTS and BPS categories and studies each category through a case study. After that, in the cross-case study, this project summarizes the common and variable aspects among these categories and drafts the design product theory for the class of RDMS.

The eSourcing Capability Model for Service Providers (eSCM-SP) is a kernel theory of this research project because it is the most comprehensive eSourcing model available for service providers. According to eSCM-SP, the life-cycle involves three phases. (1) Initiating an engagement involves gathering and negotiating requirements with a client, contracting, and designing, resourcing, and deploying the service. (2) Service is delivered according to the commitments established for the engagement. (3) The engagement is completed primarily by transitioning the resources from the provider to the client or to a third party [22]. Specific practices are enacted in each phase. eSCM-SP is applicable to both ICT and business process eSourcing and can help service providers improve their capabilities related to both ongoing, phase specific, and engagement specific eSourcing practices throughout the eSourcing life-cycle [22]. Yet, eSCM-SP has not been used and studied extensively in China. We expect only a relatively small subset of the best practices envisioned in eSCM-SP to be relevant for Chinese service providers, mainly because most providers are in relatively early phases of eSourcing capability development and thus cannot use the most advanced practices of eSCM-SP.

During both case studies and the cross-case study, the authors have analyzed the data by iterating between two phases. First, the data about the routines and the information systems they use, the most significant breakdowns in routines, and the processes and information systems used for recovering from breakdowns have been compared to the eSourcing phases and practices prescribed by the eSCM-SP. The project is especially interested in breakdowns that are caused by poorly designed, poorly used, and/or entirely missing computer-based information systems. Interactions between the eSourcing strategy, activities, processes, organizational structures, and information systems have been analyzed to define the most important information systems for the eSourcing life-cycle and its phases. Second, the results have been shown to the managers and the staff of the case companies to collect feedback, revised as necessary, and summarized. In the cross-case analysis, the common and variable parts of RTDMS and OTWMS have been compared and analyzed with respect to each phase of the eSourcing life-cycle. For example, through comparing the service providers' practices and the support offered by RTDMS and OTWMS in the initiation phase, the common and variable parts have been analyzed to draft the requirements management service of the design product theory for the abstract class of Requirements and Delivery Management Systems.

Generally, eSourcing service providers have teams to execute the specific services. Based on the case studies and eSCM-SP, this project focuses on the requirements management and delivery management teams (Table 1). These teams need to work together, for example, to process and analyze client requirements, to draft eSourcing service plans, to deal with service breakdowns, and to change service plans. In order to track the service engagements and to monitor the performance of providers, the eSourcing life-cycle should be transparent to clients.

Table 1. Responsibilities of key eSourcing service provisioning teams

Team	Responsibility
Requirements management team	Team is responsible for transforming clients requirements to executable requirements, requirements prioritization, and management. It makes the eSourcing service plan together with the delivery management team. During the delivery phase, it coordinates with the delivery management team and deals with service breakdowns. It is responsible for communicating with clients.
Delivery management team	Team executes the service delivery according to client requirements, drafts the eSourcing service plan together with the requirements management team, and reports service breakdowns (e.g., delays, accidents, and out-of-budget events). It collects metrics and deals with breakdowns during the service. During the service completion phase, it documents the services and the lessons learnt and transfers resources to clients or third parties.

The design product theory for the class of RDMS has been created based on a literature review and the analysis of the RTDMS and OTWMS. RDMS has been designed to be abstract and generic enough so eSourcing providers can use it to design domain-specific information systems and improve their processes and information systems regardless of their current practices and systems. Providers can thus use even separate requirements management and delivery management systems and use the theory to better integrate and organize these systems for enabling the end-to-end eSourcing life-cycle. For example, an RDMS instance can track the requirements execution process against the service plan and report execution results and breakdowns. It does not need to help execute specific tasks but it needs to track and report the results of the tasks. Specific tasks can be run by using other management tools. Therefore, the analysis of the practices and information systems of the case companies has helped us to scope the design product theory for the class of RDMS appropriately.

3. Meta-requirements of the design product theory for the class of RDMS

This section describes the meta-requirements of the design product theory for the class of RDMS, that is, what services integrated requirements and delivery management systems must provide to enable stakeholders to streamline the end-to-end eSourcing life-cycle (Figure 1). RDMS shall offer two categories of services: (1) requirements management and (2) delivery management (Table 2). Requirements management deals with, for example, requirements prioritization and management. Prioritization refers to establishing priorities for requirements based on client requirements and business risks (e.g., lowest cost, most effective). Requirements management is responsible for a variety of issues such as transforming client requirements to specific executable requirements, arranging proper resources for realizing the requirements, and enabling clients to track service progress. If clients change the requirements or there are unexpected breakdowns due to, for example, the unavailability of critical service components, requirements management should adjust and re-prioritize requirements as necessary**.** Delivery management is responsible for delivering the service according to the agreed-upon engagement, identifying and tracing breakdowns and their causes during the service, and reporting the breakdowns and their impacts. In the completion phase, delivery management needs to transfer the resources to clients or third parties and to record the lessons learnt for improving future services.

3.1. Initiation

In the initiation phase, a client provides a request for proposal (RFP) and documented requirements for the project. The service providers need to analyze the proposal and requirements to create a business case for estimating the profitability of the service. If the service is profitable, the providers will draft service plans to bid. Clients will analyze the plans, choose the proper providers, sign the contracts and transfer resources to the chosen providers. Providers will arrange proper staff for the services and offer training to develop the necessary skills.

Table 2. A framework for categorizing the services of the design product theory for RDMS

Requirements management	Delivery management
1 Prioritize requirements based on client needs and business and technology risks	1 Monitor the delivery progress against the service plan
2 Collect requirements from previous similar services to reduce duplication	2 Report execution results and delivery breakdowns
3 Manage interdependencies between requirements and align requirements, service delivery, and related resources.	3 Collect metrics and identify and trace breakdowns during the services
	4 Generate a delivery results report
4 Provide clients with requirements status.	5 Record lessons learnt for further services

Requirements analysis and management are the main activities throughout the eSourcing life-cycle. The requirement database enables a RDMS instance to manage prioritized requirements and provide real-time requirements information and associated delivery and breakdown information to evaluate the service quality and progress. Furthermore, bidirectional traceability is supported between requirements and delivery information artifacts across the eSourcing life-cycle.

3.1.1. Prioritizing the requirements based on business priorities and risks

Providers will analyze the RFP and client requirements. The stakeholders from clients and providers need to work together to analyze the priorities of requirements. Requirements and delivery management systems help them to prioritize requirements based on business priorities and risks.

3.1.2. Collect and analyze requirements from previous service engagements to reduce duplication

Requirements and delivery management systems collect relevant artifacts from each engagement between a provider and a client. Providers analyze the client requirements and search for relevant artifacts from previous engagements to reduce time and costs involved in creating suitable service plans, to estimate the needed efforts and resources based on previous experiences, and to price the services. After that, they draft the service plans and bid. Clients review the bids and service plans, and then select the proper service providers. After the contracting has been completed, clients will send relevant resources to the selected providers who will schedule and execute the delivery services.

3.1.3. Assign requirements to suitable teams and prepare the delivery services

After the requirements management team receives the client requirements, it needs to validate that the client requirements are executable, ensure all the required information is included, schedule the services, and send the requirement information to the stakeholders for preparation. After that, the delivery management team knows the relevant information and prepares for service delivery.

3.2. Delivery

In this phase, delivery management teams are responsible for executing specific delivery services. Clients and service providers can track the delivery processes and obtain the relevant information in a timely manner through RDMS. The delivery process includes various stakeholders. The execution of the process involves the coordination of the flow of information, services, and related activities among the stakeholders. It is important to manage the eSourcing life-cycle effectively to meet the delivery performance expectations of the stakeholders.

Requirements and delivery management systems enable clients and providers to communicate effectively and seamlessly throughout the delivery phase. For example, the information about the requirements that have been met can be shared with stakeholders to adjust resource allocation. In addition, the systems enable clients and providers to deal with breakdowns quickly. For example, whenever services are delayed, clients and providers may share knowledge to find out the reasons for the delays and to reach possible solutions. The systems record all the breakdowns to help both clients and providers to improve their performance.

3.2.1. Delivery execution

Requirements and delivery management systems support all the activities in the delivery phase. For example, they can track the services systematically and send service breakdown reports and real time delivery status information to stakeholders. They must collect metrics to measure the providers' performance in order to improve performance and to help clients to track service progress.

Delivery management is responsible for identifying and managing breakdowns, executing and monitoring the service delivery, maintaining the service routines, and storing the lessons learnt and the artifacts created into the database for reuse and adaptation during the delivery of services in future [12]. When breakdown information has been collected, it is stored in the database. Requirements and delivery management systems use the databases to provide delivery management teams with proven and reusable artifacts for executing routines and dealing with breakdowns, facilitating effective routine work.

Requirements management teams help delivery management teams to execute the services and to ensure the effectiveness of routines. Delivery management teams create reports and other artifacts, collect metrics, and store them in the databases. They deal with breakdowns and summarize the lessons learnt.

3.2.2. Service Breakdowns

Whenever service breakdowns happen, requirements and delivery management systems send relevant information to the stakeholders that need to communicate with clients and adjust the service plans as necessary. Clients will estimate the influence of breakdowns and decide whether to change their requirements and service plans. If they change the requirements, they have to adjust the service plans and negotiate with providers as necessary.

3.3. Completion

The completion phase starts when clients have received the services they need. Providers can then prepare for transferring the services to the clients or third parties. Clients need to check the services to determine whether they meet the service closure conditions. If the conditions have been met, the engagement between the client and the provider can be closed. The client needs to pay for the services according to the original agreement and the realized service quality (e.g., on time and within the budget). When the provider's financial department will receive the payment, the provider can close the engagement, summarize the lessons learnt, and compare the performance during the engagement with earlier measurement results to improve their service capabilities. For example, requirements and delivery management systems should be able to benchmark the performance with domain-specific industry standards and previous performances and report the results to delivery management teams and other stakeholders.

Requirements and delivery management systems store relevant information artifacts for future reuse. The artifacts to be stored include requirements, schedules, service plans and metrics, helping providers to improve their delivery management and performance.

4. A meta-design of the design product theory for the class of RDMS

This section describes the meta-design for the class of Requirements and Delivery Management Systems based on the meta-designs for the classes of RTDMS and OTWMS, the literature review, the eSourcing life-cycle, and the case studies. Figure 1 visualizes how the meta-design covers the eSourcing life-cycle outlined in section 3. The section concludes by explicating the linkages between the requirements management and delivery management subsystems to validate the meta-design and to justify its scope.

The two main classes of artifacts managed by the requirements and delivery management systems are the Requirement artifact and the Delivery artifact. The relationships between these artifacts are explained next. Requirement is based on client requirements and RFPs. Each Requirement needs at least one Delivery artifact to complete the service. Each Delivery artifact links with at least one Requirement.

This section introduces generic structures and attributes of the two classes of artifacts presented above. According to the design product theory, the subclasses and instances of RDMS should include at least these structures and attributes to be effective.

4.1. Requirement

Table 3 presents the generic structure of the Requirement artifacts. In the following, each class within the structure is explained.

Description describes what a requirement is about, the purpose of the requirement, and the schedule for its delivery. If there are service breakdowns that lead to changes in requirements, clients may send new requirements and providers need to renew the service plans to execute the services. Name and ID are used for identification and traceability.

Origin describes the client requirements the requirement is based on. One client requirement may be transformed to several executable requirements.

Analysis is used to probe the implications of the requirement. *Priority* is used to rank requirements and to allocate appropriate resources. During the service delivery phase, *status* can be used to check the requirements status (e.g., delivered or not).

Workflow describes what should be done next to this requirement and by whom. Requirements management teams need to allocate each requirement to one or more delivery management services.

History is used to provide information about the responsible managers and all prior edits of requirement attributes. As a result, the stakeholders can be held accountable for their actions and unexpected service breakdowns can be dealt with effectively. Changed requirements may necessitate unexpected revisions of service plans and raise service risks. History information helps service providers to eliminate many breakdowns proactively and to recover from breakdowns.

Table 3. Generic structure of a Requirement artifact

Class	Questions	Attributes
Description	What is the requirement about?	Name, ID, Description, Required date and time of delivery, Rationale
Origin	Which client requirements does the requirement refer to?	Author, Source, Date of creation
Analysis	What are the implications of the requirement?	Status, Required effort, Priority, Scheduled date and time of delivery
Workflow	What should be done to this requirement next? By whom?	Assigned Delivery services, Responsible person, Realized requirement closure date and time
History	What has been done to the requirement artifact? When?	Information about all prior edits, editors, and changes

4.2. Delivery

Table 4. Generic structure of a Delivery artifact

Class	Questions	Attributes
Description	What is the delivery service about?	Name, ID, Description, Rationale, Process, Required date and time of delivery
Origin	Which requirement does the delivery service refer to?	Author, Source requirement, Date of execution
Analysis	What are the implications of the delivery service?	Status, Required effort, Risk, Priority, Scheduled date and time of delivery
Delivery methods	Which tools and delivery methods are involved in this delivery service?	IDs of tools to be used
Workflow	What should be done to this delivery service next? By whom?	Allocation to delivery team members, Responsible person, Realized date and time of delivery
History	What has been done to this delivery artifact? When?	Information about all prior edits, editors, and changes

Table 4 presents the generic structure of the Delivery artifacts.

Description describes the purpose of an executed delivery. Process indicates the processes needed for the delivery service, including the expected results and any service breakdowns.

Origin describes the requirement(s) the delivery service refers to. One requirement may need more than one delivery service.

Analysis is used to probe the implications of a delivery service. If a delivery breaks down, the reasons for and influences of the breakdown, the solutions applied, and the effects of solutions can be documented here and/or in a separate incident management system. This information can be reused in future to help providers and clients to optimize service plans and to improve service effectiveness. *Priority* describes the priority of the delivery service and *status* refers to the delivery progress (e.g., routine, paused by a breakdown, or repairing a breakdown). Required effort describes the delivery costs, time, and resources, which can be used to calculate the total service cost of an engagement. This information can be reused to estimate the profitability and feasibility of future engagements.

The *delivery methods* class is used to provide traceability links to the tools and methods involved in the delivery services.

Workflow describes who is responsible for delivery processes and the realized time of delivery. If there is a breakdown, *Workflow* documents the workarounds that should be taken to provide the delivery and the responsible stakeholders. The delivery management team needs to communicate with the requirements management team and form a reasonable solution for the breakdown.

4.3. Validating and scoping the design product theory for the class of RDMS

According to the representatives of the case companies, the design product theories for the classes of RTDMS and OTWMS incorporate the designs of the eSourcing life-cycle processes and information systems that have helped the companies to transcend the limitations of the mediated eSourcing business model. As a result, they have gained projects that are more profitable and established direct communication with the end clients. The databases support, respectively, the associated RTDMS and OTWMS instances and accumulate the knowledge and lessons learnt, helping the case companies to enhance their service capabilities.

Specifically, an RTDMS instance aligns test requirements and related test cases and defects, helping test teams to monitor service progress and to locate defects efficiently. The RTDMS instance sends real time defect information to testers and clients, so all the involved stakeholders can communicate with each other as necessary, thus making the testing process transparent and seamless. Before Ltesting used a RTDMS instance, test teams needed to collect the defects and send them to clients in regular batches, creating unnecessary delays. Clients had to repair defects without sufficient background information. For example, clients did not necessarily know the relationships of defects to test requirements, the times when test teams had run particular test cases, and the order of test case execution. Therefore, the availability of defect information and aligned requirements and test cases reduce the costs incurred by providers and clients during the eSourcing life-cycle.

The databases have offered Ltesting defect information from previous similar projects and helped clients to repair defects quickly, impelling clients to outsource larger, longer-term, and more profitable projects to Ltesting. Ltesting has established strategic relationships with clients.

The HP Quality Center (QC), a commercially available integrated requirements, test and defect management product, served as the RTDMS instance in the case company. It required clear access rights for the different roles involved, causing some trouble for Ltesting in the early stages of adopting RTDMS. Depending on the project characteristics and the financial pressure, test analysts and test managers may have to do testers' work. RTDMS has impelled Ltesting to improve its organizational structure and, specifically, test team structure. Ltesting has established independent quality assurance teams and configuration management teams, facilitating the provisioning of more comprehensive and professional services.

"All the test projects use QC or other similar test platforms, but the other platforms mainly focus on test management. QC is the best because it can also offer complete requirements management and defect management services. The specific test tools can run on this platform, but they can also run on other platforms, so based on QC we can better control the entire service life-cycle. "-CEO

"QC is a good example of an RTDMS. We rely on it to improve our performance, communication, and service effectiveness. Clients are satisfied with our service quality and want to give better contracts to us, which means larger and more profitable projects." -CEO

PGL has developed an OTWMS instance for executing transparent logistics services. When an order is generated, the OTWMS provides the transportation team and the warehouse management team with the order information in real time, so they can prepare for service delivery. Clients can get all the information they need whenever they need it based on the order number. For example, they can track cargo information easily. The OTWMS has thus helped PGL to gain clients' trust.

"*Without the help of the OTWMS, it would be impossible to deal with breakdowns in one hour. This system enables us to communicate with all stakeholders including clients in real time, which is crucial to deal with breakdowns effectively.*" - An order manager

"*The timely information enhances our transportation and warehouse management effectiveness, helping us to improve performance. The OTWMS helps to make the life-cycle transparent and seamless, so we can earn clients' trust and establish strategic relationships with them.*" -CIO

The design product theories for the classes of RTDMS and OTWMS contribute in their respective domains and thus help the case companies to transcend the limitations imposed by the mediated outsourcing business model. The abstract class of integrated Requirements and Delivery Management Systems is based on and partly validated by RTDMS and OTWMS instances, so it can be expected to facilitate small and medium-sized eSourcing service providers in overcoming the limitations imposed by the mediated outsourcing business model.

5. Conclusions and future research

The research project provides several contributions. The design product theories for the classes of RTDMS and OTWMS contribute in their respective domains. The third main contribution is the design product theory for the abstract class of requirements and delivery management systems. This theory is partly derived (1) deductively from comprehensive kernel theories such as eSCM-SP and (2) inductively from the domain specific design product theories for the classes of RTDMS and OTWMS. The theory prescribes an abstract class of systems because instances of the class need not be built. The theory is primarily used to create more detailed domain-specific design product theories. The design product theories for the classes of OTWMS and RTDMS are such domain-specific theories used to prescribe information system subclasses of the class prescribed by the design product theory for the class of RDMS. The theory is expected to help eSourcing service providers and commercial software vendors to design domain-specific integrated systems for service provisioning and breakdown recovery throughout the eSourcing life-cycle in a variety of ICT and business process sourcing domains, helping clients and providers to manage and control the eSourcing life-cycle and to make the process transparent and seamless.

The project will continue the validation of the design product theory for the class of RDMS by making the theory bidirectionally traceable with the design product theories for the classes of OTWMS and RTDMS that specialize the abstract class for the subclasses in their specific domains. Future research is needed to use the design product theory to build a set of other domain-specific design product theories and to trace the theories back to the design product theory in order to further validate and revise it.

6. References

[1] Barthelemy, J. & Geyer, D. 2005. An empirical investigation of IT outsourcing versus quasi-outsourcing in France and Germany, Information & Management, 42(4), 533-542.

[2] Borman, M. 2006. Applying multiple perspectives to the BPO decision: a case study of call centres in Australia, Journal of Information Technology, 21(2), 99-115.

[3] Carmel, E. and Tjia, P. 2005. Offshoring Information Technology: Sourcing and outsourcing to a global workforce, Cambridge: Cambridge University Press.

[4] Carmel, E. and Agarwal, R. 2002. The maturation of offshore sourcing of IT work, MISQ-Executive, 1(2), 65-78.

[5] 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.

[6] Carmel, E. & Nicholson, B. 2005. Small firms and offshore software outsourcing: high transaction costs and their mitigation, Journal of Global Information Management 13(3), 33-54.

[7] Feeny, D., Lacity, M. & Willcocks, L.P. 2005. Taking the measure of outsourcing providers, MIT Sloan Management Review, 46(3), 41-48.

[8] Gunasekaran, A., and Ngai, E.W.T. 2003. The Successful Management of a Small Logistics Company, International Journal of Physical Distribution and Logistics Management, 33(9), pp. 825-842.

[9] Information Technology Advisory Committee (ITAC) 2003. Information Technology Outsourcing, www.cica.ca/itac.

[10] Järvenpää, S. and Mao, J. 2008. Operational Capabilities Development in Mediated Offshore Software Services Models, Journal of Information Technology (23), 3-17.

[11] Kaiser, K. & Hawk, S. 2004. Evolution of offshore software development: from outsourcing to cosourcing, MIS Quarterly Executive, 3(2), 69-81.

[12] Käkölä, T. and Taalas, A. 2008. Validating the Information Systems Design Theory for Dual Information Systems, in Proceedings of the 29th International Conference on Information Systems, Paris, Association for Information Systems.

[13] Koveos, P. & Tang, L. 2004. Offshore outsourcing: Japan, Europe and the rest of the world, Indian Journal of Economics and Business, Special Issue, 3(3), 43-62.

[14] Liu, J.J., So, S.C.K, Choy, K.L., Lau, H., and Kwok, S.K. 2008. Performance Improvement of Third-party Logistics Providers-An Integrated Approach with A Logistics Information System, International Journal of Technology Management, 42(3), pp. 226-249.

[15] Lu, Y. and Käkölä, T. (2012) An Information Systems Design Product Theory for Integrated Requirements, Test and Defect Management Systems, Hawaii International Conference on System Sciences (HICSS), January, 2012.

[16] Lu, Y. and Käkölä, T. (2013) An Information System Design Product Theory for Integrated Order, Transportation and Warehouse Management Systems, Hawaii International Conference on System Sciences (HICSS), January, 2013.

[17] Ltesting 2010. http://www.osourcing.net/.

[18] Ma, J., Li, J., Chen, W., Conradi, R., Li, J. and Liu, C. 2008. A state-of-the-practice study on communication and coordination between Chinese software suppliers and their global outsourcers, Software process: Improvement and Practice, vol.13, no. 3, pp. 233-247.

[19] Marc, J.S., Ashlyn, M.S., Johns and Dara, G.S. 2007. Outsourcing Management Information Systems, Idea Group Publishing, 1-17.

[20] Oshri, I., Kotlarsky, J. & Willcocks L. 2007. Managing dispersed expertise in IT offshore outsourcing: lessons from TATA consultancy services. MIS Quarterly Executive, 6(2), 53-65.

[21] Oshri, I., Kotlarsky, J., Rottman, J.W. & Willcocks, L.L. 2009. Global sourcing: recent trends and issues, Information Technology & People, 22(3), 192-200.

[22] The eSourcing Capability Model for Service Providers (eSCM-SP) V2.01 2010. http://www.itsqc.org/index.html.

[23] Walls, J.G., Widmeyer, G.R. and EI Sawy, O.A., 1992, "Building an Information System Design Theory for Vigilant EIS", Information Systems Research, 3(1), pp. 36-59.

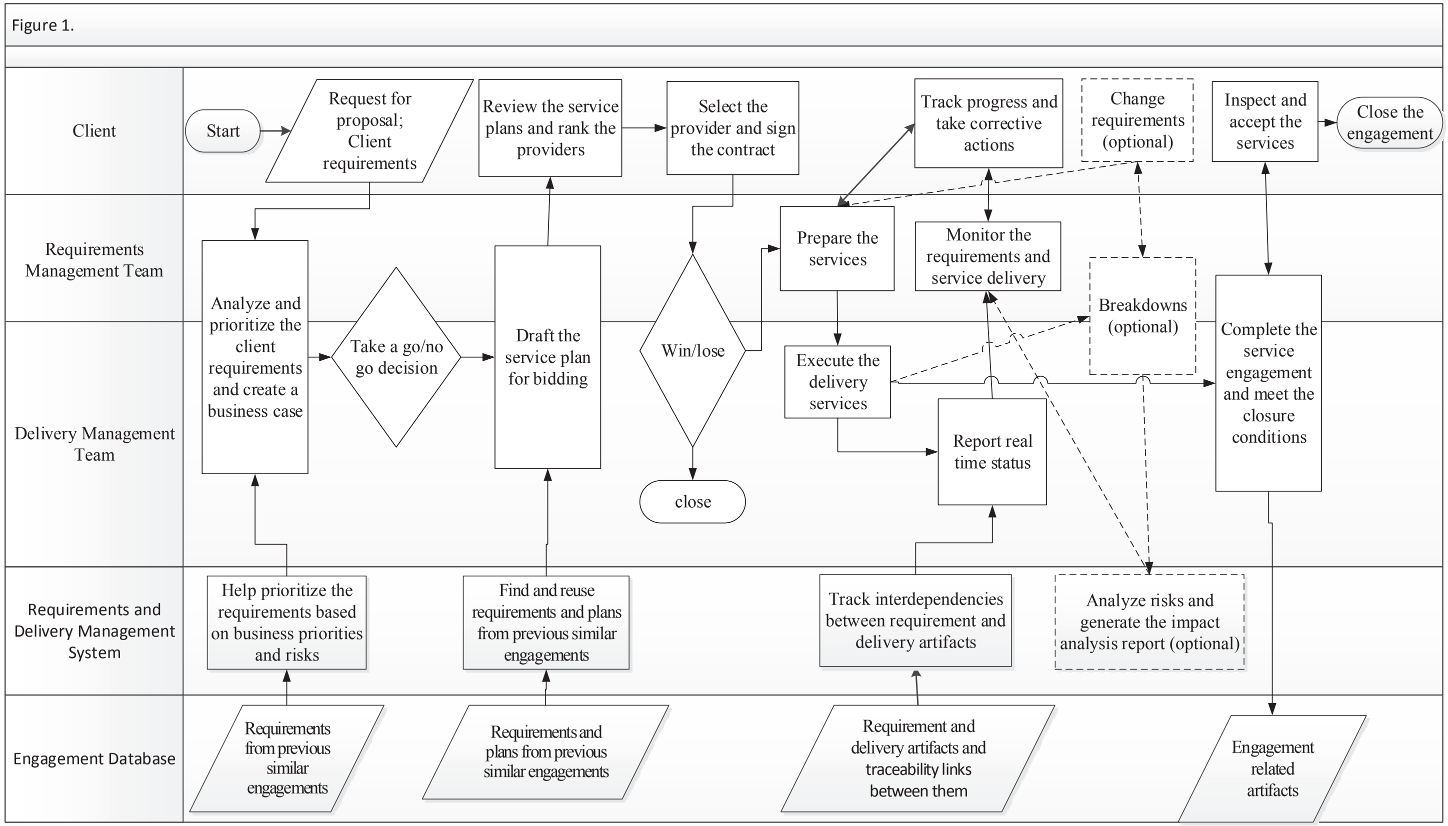

Figure 1. A meta-design for the class of integrated Requirements and Delivery Management Systems supporting the eSourcing life-cycle