Knowledge Transfer in Software-Maintenance Offshore Outsourcing

Inauguraldissertation zur Erlangung der Würde eines Doctor rerum oeconomicarum der Wirtschafts- und Sozialwissenschaftlichen Fakultät der Universität Bern

vorgelegt von
Oliver Krancher
von Regensburg, Deutschland

2013

Die Fakultät hat diese Arbeit am 23.05.2013 auf Antrag der beiden Gutachter Prof. Dr. Jens Dibbern und Prof. Dr. Sandra Slaughter als Dissertation angenommen, ohne damit zu den darin ausgesprochenen Auffassungen Stellung nehmen zu wollen.

TABLE OF CONTENTS

TABLE OF CONTENTS .... **V**

LIST OF FIGURES .... **VIII**

LIST OF TABLES .... **X**

LIST OF ABBREVIATIONS .... **XII**

CHAPTER I INTRODUCTION **1**

1 Knowledge Transfer in Software-Maintenance Offshore Outsourcing *1*

2 Motivation and Research Questions *3*

2.1 Studies 1 and 2: Learning Software-Maintenance Tasks 5

2.1.1 Study 1: Managing Cognitive Load 7

2.1.2 Study 2: Media Choice and Communication Performance 9

2.2 Studies 3 and 4: Governing the Learning of Software-Maintenance Tasks 9

2.2.1 Study 3: Governing Individual Learning 10

2.2.2 Study 4: The Impact of Client Management Decisions on Transition Outcomes 12

3 Overview of the Dissertation *13*

CHAPTER II LEARNING SOFTWARE-MAINTENANCE TASKS **18**

STUDY 1 MANAGING COGNITIVE LOAD IN SOFTWARE-MAINTENANCE OFFSHORING: A MIXED-METHODS STUDY 18

Abstract *18*

1 Introduction *19*

2 Theory *22*

2.1 Cognitive Load Theory 22

2.2 Theoretical Model 23

3 Methods *28*

3.1 Research Design 28

3.2 Data Collection 30

3.3 Data Analysis 32

3.3.1 Step 1: Initial Qualitative Analysis (Coding and Quantitizing) 32

3.3.2 Step 2: Quantitative and Qualitative Analysis of the Learning Task 35

3.3.3 Step 3: Quantitative and Recontextualized Qualitative Analysis at the Level of the Knowledge Transfer Project 36

4 Results *36*

4.1 Statistical Analysis of Learning Task Configurations 37

4.2 A Qualitative Perspective on Supportive Information 40

4.3 Alternative Expertise Measures 42

4.4 The Role of Knowledge Specificity 43

5 *Discussion* ... *48*
5.1 Corroboration with Prior Research ... 50
5.2 Theoretical Contribution ... 50
5.3 Limitations and Future Work ... 52
5.4 Implications for Practice ... 54
Appendix II-1: Coding Rules for Quantitizing ... *56*
Simplified Task Types ... 56
Component Complexity (Wood 1986) ... 57
Coordinative Complexity (Wood 1986) ... 58
Supportive Information ... 58
Expertise ... 59
Cognitive Load ... 62
Appendix II-2: Results of the Fixed-effects Panel Regression ... *63*
Appendix II-3: Comparison of Alternative Expertise Measures ... *64*

STUDY 2 MEDIA CHOICE AND COMMUNICATION PERFORMANCE DURING KNOWLEDGE TRANSFER IN SOFTWARE-MAINTENANCE OFFSHORING ... 68
Abstract ... *68*
1 *Introduction* ... *69*
2 *Theory* ... *71*
2.1 The Cognitive Theory of Multimedia Learning ... 71
2.2 Media Synchronicity Theory ... 73
2.3 Theoretical Model ... 75
3 *Methods* ... *79*
3.1 Case Contexts ... 79
3.2 Data Collection ... 81
3.3 Data Analysis ... 81
4 *Results* ... *83*
4.1 Dual-Channel Capability ... 83
4.2 Reprocessability and Intrinsic Cognitive Load ... 88
5 *Discussion* ... *90*
Appendix II-4: Cases Overview ... *94*
Appendix II-5: Coding Rules and Sample Quotes ... *95*
Appendix II-6: Interview Statements on Causal Relationships ... *97*

CHAPTER III GOVERNING THE LEARNING OF SOFTWARE-MAINTENANCE TASKS ... 98

STUDY 3 GOVERNING INDIVIDUAL LEARNING IN THE TRANSITION PHASE OF SOFTWARE-MAINTENANCE OFFSHORING: A DYNAMIC PERSPECTIVE ... 98
Abstract ... *98*
1 *Introduction* ... *99*
2 *Conceptual framework* ... *100*
2.1 Outsourcing Governance and Control ... 101
2.2 Cognitive Load Theory ... 102

2.3 Knowledge Transfer Theory ... 103
3 Methods ... 104
4 Case Analysis ... 107
4.1 Illustration of Construct Evaluation ... 107
4.2 Results and Discussion of Cases ... 109
5 Implications ... 114
Appendix III-1: Construct Definitions, Coding Rules, and Construct Instances in Case 1 ... 118

STUDY 4 MANAGEMENT DECISIONS IN SOFTWARE-MAINTENANCE OFFSHORING TRANISITIONS: INSIGHTS FROM A DYNAMIC MODEL ... 121
Abstract ... 121
1 Introduction ... 122
2 A Dynamic Model of Learning and Support in SMOO Transitions ... 124
2.1 The System Dynamics Paradigm ... 124
2.2 Model 1: A Simple Model of Cognitive-Load-Based Learning ... 125
2.3 Model 2: A Model of Cognitive-Load-Based Learning and Static Support ... 128
2.4 Model 3: A Model of Cognitive-Load-Based Learning and Dynamic Support ... 130
3 A Monte-Carlo Simulation ... 134
4 Results of the monte-carlo Simulation ... 136
5 Discussion ... 138
Appendix III-2: Model Assumptions ... 141
Assumptions in Model 1 ... 141
Additional Assumptions in Model 2 ... 141
Additional Assumptions in Model 3 ... 142
Appendix III-3: Simulation Code ... 142
parameters.m ... 142
Testdata.m ... 143
run.m ... 144
ServiceLevel.m ... 145
parallelSave.m ... 146

CHAPTER IV CONCLUSION ... 147
Implications for Research ... 147
Implications for Practice ... 155

ACKNOWLEDGEMENTS ... 158

LITERATURE ... 159

STATEMENT OF AUTHORSHIP ... 169

LIST OF FIGURES

FIGURE I-1: THE TRANSITION PHASE 3

FIGURE I-2: DISSERTATION OVERVIEW 4

FIGURE II-1: THEORETICAL MODEL 24

FIGURE II-2: DATA ANALYSIS 32

FIGURE II-3: THE EVOLUTION OF EXPERTISE IN THE FIVE CASES 43

FIGURE II-4: SPECIFICITY AND INITIAL EXPERTISE 44

FIGURE II-5: SUMMARY OF RESULTS 48

FIGURE II-6: COGNITIVE LOAD CATEGORIES 62

FIGURE II-7: THE COGNITIVE THEORY OF MULTIMEDIA LEARNING (SIMPLIFIED FROM MAYER AND MORENO 2003) 73

FIGURE II-8: A THEORETICAL MODEL OF MEDIA CHOICE AND COMMUNICATION PERFORMANCE 76

FIGURE III-1: CONCEPTUAL FRAMEWORK 101

FIGURE III-2: DATA ANALYSIS PROCESS 106

FIGURE III-3: PROCESS MODEL OF THE INTERACTION OF GOVERNANCE AND LEARNING 114

FIGURE III-4: MODEL 1: A SIMPLE MODEL OF COGNITIVE-LOAD-BASED LEARNING 126

FIGURE III-5: MODEL 1 BEHAVIOR (INITIAL EXPERTISE = .15, TASK COMPLEXITY = .5) 128

FIGURE III-6: MODEL 1 BEHAVIOR (INITIAL EXPERTISE = .3, TASK COMPLEXITY = .5) 128

FIGURE III-7: MODEL 2: A MODEL OF COGNITIVE-LOAD-BASED LEARNING AND STATIC SUPPORT...129

FIGURE III-8: MODEL 2 BEHAVIOR (INITIAL EXPERTISE = .15, TASK COMPLEXITY BEFORE SIMPLE-TO-COMPLEX SEQUENCING = .5) 129

FIGURE III-9: MODEL 3: A MODEL OF COGNITIVE-LOAD-BASED LEARNING AND DYNAMIC SUPPORT131

FIGURE III-10: MODEL 3 BEHAVIOR (FCC = 0) 133

FIGURE III-11: MODEL 3 BEHAVIOR (FCC = 1) 133

FIGURE III-12: MODEL 3 BEHAVIOR (COEXISTENCE ENDS AFTER 10 OUT OF 100 PERIODS (SEE DASHED LINE)) 134

FIGURE III-13: MODEL 3 BEHAVIOR (COEXISTENCE ENDS AFTER 30 OUT OF 100 PERIODS (SEE DASHED LINE)) 134

FIGURE III-14: TRANSITION DURATION IN FUNCTION OF COEXISTENCE DURATION AND INITIAL EXPERTISE (TASK COMPLEXITY = .6, FCC = .5) 137

FIGURE III-15: TRANSITION DURATION IN FUNCTION OF COEXISTENCE DURATION AND FCC (TASK COMPLEXITY = .6, INITIAL EXPERTISE = .2) 137

FIGURE III-16: TRANSITION DURATION IN FUNCTION OF COEXISTENCE DURATION AND FCC (TASK COMPLEXITY = .6, INITIAL EXPERTISE = 0) 137

FIGURE IV-1: A THEORY OF THE GOVERNANCE AND DESIGN OF EFFECTIVE KNOWLEDGE TRANSFER IN SMOO 147

LIST OF TABLES

TABLE I-1: OVERVIEW OF STUDIES 17

TABLE II-1: TASK TYPES (ADAPTED FROM VAN MERRIËNBOER ET AL. 2002) 26

TABLE II-2: RESEARCH DESIGN 29

TABLE II-3: CASES 30

TABLE II-4: DATA SOURCES 31

TABLE II-5: OPERATIONALIZATION OF CONSTRUCTS 33

TABLE II-6: RELIABILITY RESULTS 35

TABLE II-7: DESCRIPTIVE STATISTICS AND ZERO-ORDER CORRELATIONS 37

TABLE II-8: RESULTS OF ORDINAL REGRESSION 38

TABLE II-9: CORROBORATION WITH PRIOR STUDIES 51

TABLE II-10: EXAMPLES OF TASK TYPES 56

TABLE II-11: CATEGORIES OF SIMPLIFIED TASK TYPES 56

TABLE II-12: CODING SCHEME OF COMPONENT COMPLEXITY 57

TABLE II-13: CODING SCHEME OF COORDINATIVE COMPLEXITY 58

TABLE II-14: DURATION OF THE SUPPORTIVE-INFORMATION ACTIVITIES PER DAY 58

TABLE II-15: FICTITIOUS EXAMPLE OF THE CODING OF EXPERTISE 60

TABLE II-16: WEIGHTS OF BOK CATEGORIES IN CASE 1 60

TABLE II-17: EXPERTISE FORMULAE 61

TABLE II-18: SAMPLE QUOTES OF COGNITIVE LOAD 63

TABLE II-19: RESULTS OF THE FIXED-EFFECTS PANEL MODEL 64

TABLE II-20: RESULTS FROM ALTERNATIVE EXPERTISE MEASURES 65

TABLE II-21: OVERVIEW OF MST AND THE CTML 75

TABLE II-22: DATA SOURCES 82

TABLE II-23: DUAL-CHANNEL CAPABILITY AND COMMUNICATION PERFORMANCE 84

TABLE II-24: INTERACTION OF DUAL-CHANNEL CAPABILITY AND INTRINSIC COGNITIVE LOAD ON COMMUNICATION PERFORMANCE 84

TABLE II-25: REPROCESSABILITY AND COMMUNICATION PERFORMANCE............................88

TABLE II-26: INTERACTION OF REPROCESSABILITY AND INTRINSIC COGNITIVE LOAD ON COMMUNICATION PERFORMANCE 88

TABLE III-1: DATA SOURCES 105

TABLE III-2: CONSTRUCT INSTANCES 111

TABLE III-3: DEFINITIONS OF KEY TERMS 126

TABLE III-4: RESULTS OF SENSITIVITY ANALYSES 138

LIST OF ABBREVIATIONS

Agr.	Agreement
AIC	Akaike Information Criterion
Alpha	Cronbach’s Alpha
BoK	Body of Knowledge
CLT	Cognitive Load Theory
CL	Cognitive Load
Comp	Component Complexity
Cord	Coordinative Complexity
CTML	Cognitive Theory of Multimedia Learning
Ex	Expertise
Ex. Msr.	Expertise Measure
FCC	Formal and Clan Controls
IS	Information Systems
Kappa	Cohen’s Kappa
Learn. Eff.	Learning Effectiveness
MST	Media Synchronicity Theory
Regr. Coeff.	Regression Coefficients
SD	Standard Deviation
SI	Supportive Information
SME	Subject Matter Expert(s)
SMOO	Software-Maintenance Offshore Outsourcing

Std. Error	Standard Error
Std. Regr. Coeff.	Standardized-Regression Coefficients
STCS	Simple-to-Complex Sequencing
Task Compl.	Task Complexity

CHAPTER I INTRODUCTION

1 KNOWLEDGE TRANSFER IN SOFTWARE-MAINTENANCE OFFSHORE OUTSOURCING

Globalization has dramatically altered how large western companies organize their information systems (IS) services. Outsourcing—the delegation of tasks to domestic or foreign vendors—has emerged as a popular strategy. Globally, IS services worth $270 billion were outsourced in 2010 (Oshri et al. 2011). Significant labor cost differences and scarce domestic IS staff have helped offshore vendors (i.e. vendors in remote countries) increase their stake in the outsourcing business. India has emerged as the leading offshoring destination, offering a substantial talent pool at still moderate labor cost (ATKearney 2011). Given that labor expenses for software maintenance (i.e. changes to software after its first deployment) consumed a major fraction of the IS spending of large firms (Banker et al. 1993), it is no surprise that software-maintenance services have been popular candidates for offshore outsourcing. With the increasing maturation of software-maintenance offshore outsourcing (SMOO), observers see a recent trend towards outsourcing highly knowledge-intensive work such as the maintenance of complex software (Booth 2013).

Yet, offshore outsourcing knowledge-intensive software-maintenance tasks poses new challenges. Clients involved in knowledge-intensive SMOO projects can face substantial unexpected costs, which may exceed the savings from labor arbitrage (Dibbern et al. 2008). Tedious knowledge transfer (i.e. the process through which the vendor team acquires the knowledge required for the task) may lie at the heart of a considerable part of these costs, in particular when the required knowledge is highly specific to the client (Dibbern et al. 2008). Knowledge-related problems are particularly salient during the transition phase, the phase during which the offshore team takes over the responsibility for delivery at the outset of offshoring projects. During transition, offshore team members may be cognitively overloaded by the vast amounts application knowledge that they are expected to acquire (Chua and Pan 2008). As a consequence of overload, they may not be able to take over tasks according to the plans made prior to transition (Chua and Pan 2008; Dibbern et al. 2008). These observations are in line with software-maintenance research. Software maintenance has been described as a cognitively demanding task. Maintainers draw substantially from their knowledge of the software application to design and implement modifications such as defect corrections and software enhancements (Von Mayrhauser and Vans 1995). When corporate software applications have grown over years towards a substantial size, the required knowledge may frequently be vast. Experience in maintaining the same (rather than a related or an unrelated) software system has thus been identified to be among the strongest predictors of individual maintenance performance (Boh et al. 2007), suggesting that experience plays a key role in enabling engineers to cope with complexity. Vendor engineers in SMOO projects may typically lack experience in the same software system when they take over the maintenance of a client's software system during transition. Offshore-specific context factors such as cultural differences and language barriers may further complicate the knowledge transfer. Taken together, transitions of complex software-maintenance tasks present clients and vendors with the challenge of effectively transferring knowledge to vendor teams. The failures reported in the literature (Chua and Pan 2008; Dibbern et al. 2008; Wende and Philip 2011) indicate that practice has not found conclusive answers to this challenge yet.

SMOO transitions confront project members with at least two problems (see Figure I-1 for an overview). The first problem is how to effectively design the knowledge transfer to vendor staff. The process of knowledge transfer may be seen as a sequence of learning activities such as formal face-to-face presentation sessions, replay sessions (i.e. the vendor employees report their understanding), creating and reading documents, phone conferences, job-shadowing (i.e. the vendor engineer observes how the subject matter expert (SME) performs a task), on-the-job training (i.e. the vendor engineer works on a task), and informal face-to-face discussions (Blumenberg et al. 2009; Chua and Pan 2008; Gregory et al. 2009; Oshri et al. 2008; Tiwari 2009; Vlaar et al. 2008; Williams 2011). These learning activities aim at helping on-site coordinators and offshore team members acquire the task knowledge (the knowledge required to perform the software-maintenance tasks) and may involve SME, who have in-depth knowledge in the task domain due to their past experience. The design of knowledge transfer is thus concerned with combining learning activities so that vendor staff effectively acquires the task knowledge.

The second problem is the governance of knowledge transfer. If managers are aware in what activities vendor engineers should engage for effective learning, they may wonder how they can enforce that these activities eventually take place. Governance can be defined as structure and actions that align the behavior of actors with the client's objectives (Huber et al. 2011). For instance, clients management may prescribe procedures for knowledge transfer (Gregory et al. 2009), evaluate learning outcomes, or rely on the self-regulation of the learning process by the vendor engineers. Initial research suggests that client managers need to actively govern knowledge transfer to the vendor (Gregory et al. 2009). The governance of knowledge transfer is thus concerned with combining governance mechanisms in such a way that effective learning occurs.

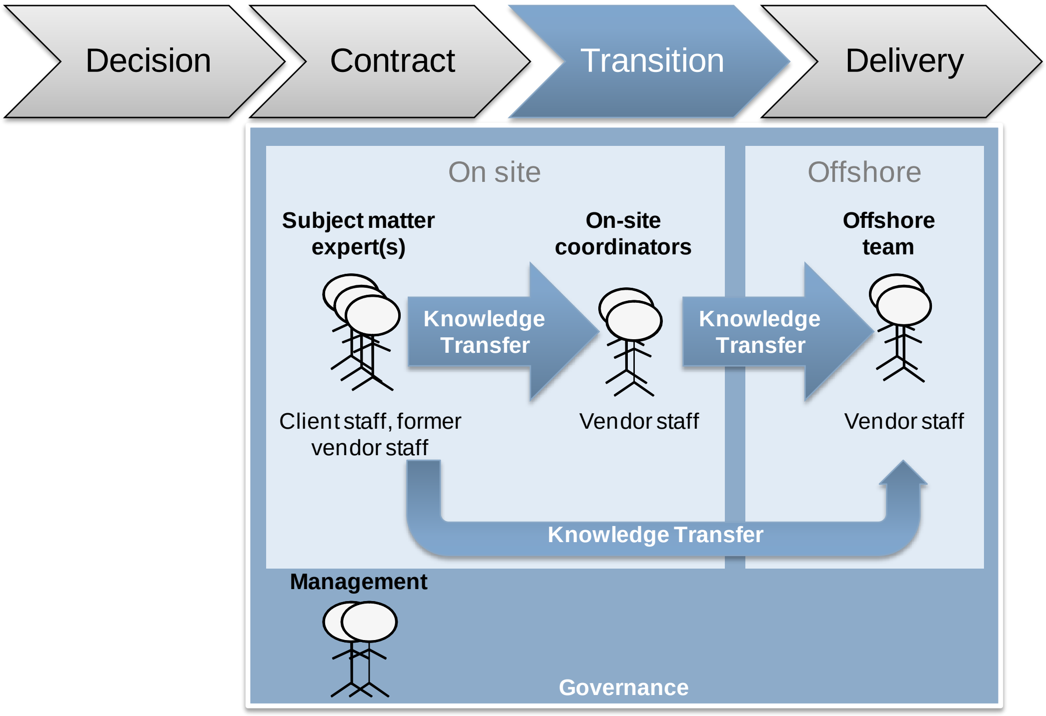

Figure I-1: The Transition Phase

2 MOTIVATION AND RESEARCH QUESTIONS

Practitioners involved in SMOO transitions find scant guidance in the existing literature on how to design and govern knowledge transfer. Although there seems to be agreement on the central role of knowledge transfer for the success of offshore outsourcing projects (e.g. Kotlarsky and Oshri 2005; Lacity et al. 2010; Westner and Strahringer 2010), we lack theoretically grounded understandings of how the vendor engineers can effectively acquire the task knowledge and how management can steer this process.

The dissertation at hand is an attempt to fill these gaps by contributing to a theoretical framework of the design and the governance of effective knowledge transfer in SMOO projects. The framework centers on the individual learning of vendor engineers. While other phenomena such as group learning (Oshri et al. 2008), managing cultural differences (Winkler et al. 2008) and motivation to share knowledge (Ko et al. 2005) may also be salient during knowledge transfer in SMOO transitions, issues of individual learning seem to play a pivotal role for several reasons. First, the existing literature on software-maintenance offshoring transitions stresses that issues of cognitive overload severely constrain the assimilation of information by vendor employees (Chua and Pan 2008; Dibbern et al. 2008). Cognitive overload may be primarily an issue of the cognition of the individual engineer. Second, software-maintenance research emphasizes the role of individual cognitions for performance (Banker et al. 1998; Boh et al. 2007; Pennington 1987; Von Mayrhauser and Vans 1995). Third, individual learning may be the foundations for group learning or organizational learning at a later stage (Kim 1993; Nonaka 1994). Individual learning may thus dominate during early transitions while team learning unfolds later in the project. For instance, successful individual learning by the on-site coordinators may enable them to establish organizational routines to improve team productivity. The focus on individual learning does not imply that other issues are not of relevance. The conclusion section of this dissertation will be used to discuss implications for these issues.

Four studies have been conducted to contribute to a theory of the design and the governance of knowledge transfer in SMOO projects. Figure I-2 gives an overview of how the four studies of the dissertation are related to each other to accomplish this goal. The studies 1 and 2 dealt with the question through what learning activities vendor engineers can effectively acquire the task knowledge and how this process is influenced by knowledge specificity (the degree to which the required knowledge is specific to the client). The studies 3 and 4 investigated how the stakeholders in SMOO projects can govern these activities and how governance may be dynamically influenced by the outcomes of learning activities.

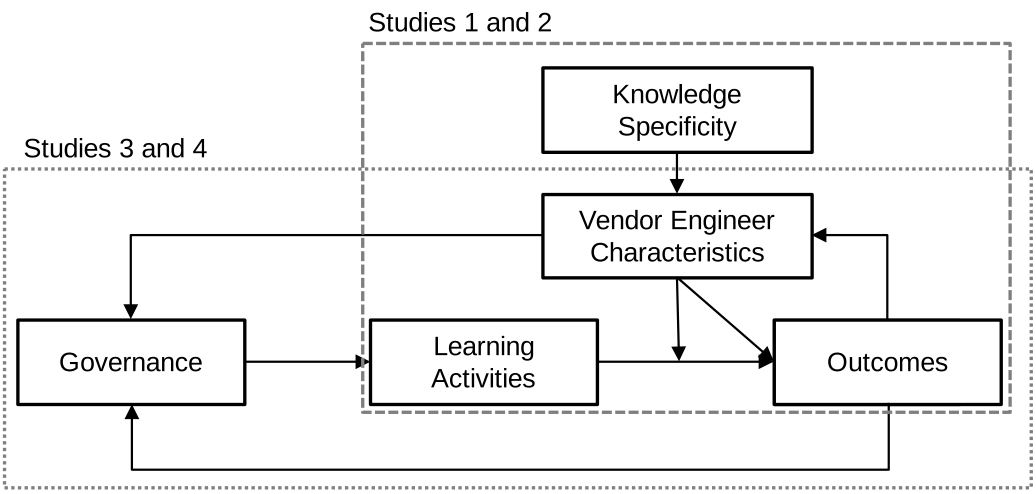

Figure I-2: Dissertation Overview

The remainder of this section gives an overview of the research questions that are addressed in the four studies. Because the studies built on each other, the study findings are outlined whenever they led to research questions that were addressed in subsequent studies of this dissertation.

2.1 Studies 1 and 2: Learning Software-Maintenance Tasks

With the advent of SMOO transitions, the question how maintainers can effectively acquire the task knowledge has gained importance. This is because transitions are expensive. During transition, projects have to bear the cost for both SME and vendor engineers. Long phases of coexistence of SME and vendor engineers can therefore quickly erode the business case behind offshoring. While a significant cost is at stake during transitions, case study evidence suggests that offshore staff frequently take over responsibility later and to a lesser extent than planned prior to transition (Chua and Pan 2008; Dibbern et al. 2008; Gregory et al. 2009; Wende and Philip 2011). These deviations from plan cause extra costs not only for knowledge transfer, but also for control, coordination, and specification (Dibbern et al. 2008). In other words, projects that succeed in designing effective knowledge transfer to vendor staff are more likely to result in positive value for both parties (see also Westner and Strahringer 2010 for empirical evidence).

Although effective knowledge transfer seems central to the success of SMOO projects, the existing literature grants only limited insight into how effective knowledge transfer can be designed. Prior work provides valuable insights into when and how the participants in SMOO projects communicate their knowledge (Kotlarsky and Oshri 2005; Oshri et al. 2008). However, the causal link between communicating knowledge to the vendor and enabling vendor engineers to perform their tasks remains largely unexplored. The communication of knowledge may not be a sufficient condition for the successful task performance by offshore engineers. In the case study by Chua and Pan (2008), the SME communicated vast amounts of knowledge to offshore engineers during presentation sessions, but the offshore engineers felt overwhelmed by the information and failed to apply it when they had to take over the more complex tasks. It is thus unlikely that the deviations from plan may have been prevented by sharing higher amounts of explicit knowledge, i.e. of knowledge that can be communicated (Nonaka 1994; Polanyi 1962). A vendor team in the case study of Dibbern et al. (2008) planned to acquire the task knowledge through reverse-engineering tools, making use of the explicit knowledge expressed in the source code. Contrary to the expectations, the team were not able to perform the task (Dibbern et al.

2008). In both cases, substantial explicit knowledge did not enable satisfactory task performance. These observations are not surprising in the light of theory from educational psychology. The term *inert knowledge* (Renkl et al. 1996) has been coined to describe the idea that the communication of knowledge does frequently not entail skillful action. Knowledge risks remaining inert when participants expect that explicit knowledge qualifies for problem-solving or when knowledge is not anchored in the context of a particular task (Renkl et al. 1996). Put differently, when engineers are expected to acquire knowledge through face-to-face presentations and documents, there is a significant risk that they will not be able to apply this knowledge to solve software-maintenance problems. The view of knowledge transfer as a communication process has also been called the information delivery view of learning (Mayer 2003). This term alludes to the presumable misconception of knowledge transfer as the process of delivering information to vendor engineers. Seeing knowledge transfer as a communication process may therefore be a too narrow view if the desired outcome is their ability to solve maintenance problems.

Educational research favors learning tasks as a means of preventing inert knowledge (Collins et al. 1991; Jonassen 1997; Merrill 2002; Van Merriënboer et al. 2002). Learning tasks are “authentic whole-task experiences based on real-life tasks” (Van Merriënboer and Kirschner 2007, p. 14). For instance, learning tasks are used when vendor engineers work on a software-maintenance request or when they try to make sense of how an expert solves a software defect during job-shadowing. Conversely, reading a document about the functionality of a software application or attending a presentation on the software architecture are not considered learning tasks because these activities are not centered on experiences of realistic maintenance tasks. This stream of literature would thus recommend knowledge transfer approaches that are built around a series of realistic maintenance tasks.

While learning tasks can have a range of beneficial effects (Van Merriënboer et al. 2003), they risk overloading inexperienced learners. This risk is substantial in SMOO given the accounts of cognitively overloaded engineers in the literature (Chua and Pan 2008; Dibbern et al. 2008). One stream of literature in education psychology has therefore emphasized the need to manage the cognitive loads imposed by learning tasks, in particular when there is a risk of cognitive overload. The findings from this research have given rise to cognitive load theory (CLT; Sweller et al. 1998; Van Merriënboer and Sweller 2005), the recently most popular theory in instructional design (Ozcinar 2009). CLT predicts learning outcomes based on the constraints of the human cognitive architecture (Plass et al.

2010). CLT suggests that both high and low cognitive loads (i.e. demands that a task imposes on a learner) on learners should be avoided (Schnotz and Kürschner 2007). When tasks impose high cognitive demands, they deprive learners of the mental resources that would be necessary for schema acquisition, one central learning process (Van Merriënboer and Sweller 2005). When tasks impose too low cognitive demands, they may not bear substantial learning opportunities (Schnotz and Kürschner 2007). The detrimental effects from too high and too low load could be replicated in a substantial series of controlled experiments (for overviews see Kirschner et al. 2006; Van Merriënboer and Sweller 2005). Effective learning environments therefore need to be designed so that the cognitive loads on learners are managed to be at a moderate level.

The focus on learning tasks and the implications from CLT bring a shift in perspective. Rather than being a question of the "communication of knowledge from a source so that it is learned and applied by a recipient" (Ko et al. 2005, p. 62), knowledge transfer becomes a matter of learning task design. In this view, knowledge transfer is a sequence of learning tasks and other related learning activities, in which the cognitive load imposed by each task should be managed to avoid too high and too low cognitive load.

2.1.1 Study 1: Managing Cognitive Load

Study 1 was intended to shed light on the design of this knowledge transfer process, accounting for the central role of cognitive load. Managers and engineers may want to understand how they should combine particular learning activities in the knowledge transfer process in such a way that vendor engineers acquire the task knowledge most effectively. Learning activities may include the engagements in particular types of learning tasks and in supportive information. Learning tasks confront the vendor engineers with problems that are realistic for the task domain (Van Merriënboer et al. 2003). Examples of learning tasks include the work on a conventional software-maintenance problem, the completion of a partially solved software-maintenance problem, and job-shadowing. Supportive information provides blueprints for schemas related to the non-recurrent aspects of a task (Van Merriënboer et al. 2003). Formal presentation sessions, documents, and informal discussions on the software architecture, business concepts and other subject matters are sources of supportive information. The existing literature describes what learning activities are typically involved in SMOO projects (Chua and Pan 2008; Nicholson and Sahay 2004). However, we lack a theoretically grounded understanding of how a particular learning activity undertaken by a particular vendor engineer in a particular context impacts outcomes such as cognitive load and learning effectiveness. For instance, why were cognitive loads so high after the offshore engineers in the case study by Chua and Pan (2008) ran through a series of formal presentations? Could these high cognitive loads have been avoided by an alternative knowledge transfer design? Do projects that differ in knowledge specificity, a prominent knowledge-related context factor in prior IS outsourcing research (Dibbern et al. 2008), require different learning activities? Study 1 aimed at developing a theoretical framework that helps answer these questions. The following research question was addressed:

> ***Research Question 1****: What determines the effectiveness of particular learning activities during SMOO transitions?*

The results from a mixed-methods multiple-case study indicated how particular learning activities in five SMOO transitions were related to effectiveness. Drawing on CLT, cognitive load was adopted as the criterion against which to judge effectiveness. The confirmatory first part of the case study tested whether the predictions of CLT were able to explain the cognitive loads associated with the learning tasks embedded in each of the cases. The predictors of CLT include (1) the learner's expertise, (2) task complexity, (3) the use of simplified task types such as worked examples (the full solution is given to the learner) and completion tasks (parts of the solution are given to the learner), and (4) supportive information (Van Merriënboer et al. 2003; Van Merriënboer and Sweller 2005). In the data of study 1, expertise, task complexity, and the use of simplified task types were strong predictors of cognitive load. Simplified task types and the selection of tasks based on their complexity such as in a simple-to-complex sequencing strategy may thus be effective means to manage the cognitive load on vendor engineers. These strategies may be needed in function of the vendor engineer's expertise, which emanates from prior related experience and from the engagement in learning tasks during transition. In contrast, the correlations between supportive information and cognitive load were weak and not significant. The exploratory part of study 1 also indicated how knowledge specificity is related to these predictions.

The weak correlations between supportive information and cognitive load may be surprising because evidence of the use of supportive information during transition phases abounds in the literature. In the case study by Chua and Pan (2008), the offshore engineers attended extensive presentation sessions on application knowledge and organizational knowledge. Dibbern et al. (2008) reported on high efforts for the communication of knowledge to vendor staff in the high-specificity cases. The heavy use of supportive information may seem contradictory given the weak correlations of supportive information and cognitive load. The qualitative analysis in study 1 suggested one substantive explanation for the weak relationships: Supportive information may have been differentially effective. While the vendor engineers perceived supportive information as effective in some episodes, they struggled in other episodes, in particular when documents were used as a medium for the communication of intrinsically complex knowledge. This indicates that media choice may impact how well the vendor engineers understand the supportive information and gave rise to the research question of study 2.

2.1.2 Study 2: Media Choice and Communication Performance

The qualitative analysis of study 1 suggested that supportive information may not always relieve the cognitive load associated with related learning tasks and that media choice may be one moderating factor in this realm. The existing literature does not make unambiguous claims on how media choice impact communication performance (the extent to which recipients are able to build or revise a mental model from a message) in supportive-information activities. Two prominent theories make divergent predictions of how media choice impacts communication performance when information is conveyed during knowledge transfer. Media synchronicity theory (Dennis et al. 2008) advocates media low in synchronicity, i.e. media that allow recipients to reprocess messages at a self-selected pace such as documents. Conversely, the cognitive theory of multimedia learning (Mayer and Moreno 2003)—a theory within the CLT framework—recommends using media that allow to simultaneously process visual and auditory signals such as during face-to-face sessions. Study 2 was intended to shed empirical light on these divergent predictions by addressing the following question:

> ***Research Question 2****: How does media choice influence communication performance in supportive-information activities during SMOO transitions?*

2.2 Studies 3 and 4: Governing the Learning of Software-Maintenance Tasks

The results from the studies 1 and 2 and the findings presented at a conference (Krancher and Dibbern 2012) indicate some of the ingredients of effective learning processes in SMOO transitions. Vendor engineers should engage in authentic learning tasks and the cognitive loads associated with these tasks should be managed by simple-to-complex sequencing, simplified task types, and, possibly, supportive information provided through appropriate media that collectively fit the expertise of the learners. Although it may be valuable to understand in what activities vendor engineers should engage for effective learning, client management may be interested in how they can influence this process through governance. In the context of this study, governance shall be understood as structures and actions (Huber et al. 2011) that align the behavior of individuals with the knowledge transfer goals of the client. The studies 3 and 4 investigated how governance influences knowledge transfer.

2.2.1 Study 3: Governing Individual Learning

While much recent research has empirically examined the governance of software services in outsourcing projects (Choudhury and Sabherwal 2003; Rustagi et al. 2008; Tiwana and Keil 2007; Tiwana and Keil 2009), few studies have considered the governance of knowledge transfer (see Gregory et al. 2009 for an exception). Knowledge transfer during transition may merit special attention as an object of governance for at least three reasons. First, knowledge transfer is central to the success of outsourcing projects (e.g. Kotlarsky and Oshri 2005; Lacity et al. 2010; Westner and Strahringer 2010). Client management has therefore strong interest in understanding how their actions influence knowledge transfer.

Second, initial evidence suggests that the governance of knowledge transfer may depart from the agency-theoretic logic that was frequently found to explain the governance of software services. The agency-theoretic thinking implied in control theory (Kirsch 1996; Ouchi 1979) and IS outsourcing governance research suggests that organizations align governance with transaction hazards (Anderson and Dekker 2005). In this view, asymmetric incentives between a principal (or a controller) and an agent (or a controllee) call for governance to align the agent's behavior of the principal's objectives (Eisenhardt 1989a; Kirsch 1996; Ouchi 1979). However, transaction hazards may not explain how clients governed knowledge transfer in case studies described in the literature. Clients were reported to actively govern knowledge transfer (Dibbern et al. 2008; Gregory et al. 2009). For instance, client management may specify procedures for knowledge transfer or communication guidelines (Gregory et al. 2009). On the other hand, both clients and vendors may benefit from successful knowledge transfer. The client may expect better delivery outcomes and the vendor may hope for lower delivery efforts if the vendor engineers have effectively acquired the task knowledge. Hence, the observed high need for governance of knowledge transfer appears paradoxical in absence of major incentive misalignments, putting existing explanations for outsourcing governance into question.

Third, the relationship between governance and knowledge transfer may be particularly complex. Both governance portfolios (Choudhury and Sabherwal 2003; Kirsch 2004) and knowledge evolve over time. Knowledge may be both an antecedent to and an outcome of governance (Sydow and Windeler 2003; Tiwana and Keil 2007). Yet, the mechanisms that explain how governance and knowledge transfer mutually influence each other are little known.

In sum, although knowledge governance seems to play an important role in SMOO projects, our current understanding of the governance of knowledge transfer in SMOO projects is fairly limited. Given the bidirectional causal mechanisms indicated in the literature (Sydow and Windeler 2003; Tiwana and Keil 2007), study 3 adopted a dynamic, process-oriented perspective to explore the links between governance and knowledge transfer. The study addressed the following question:

> ***Research Question 3****: How do governance and individual learning interact over time during SMOO transitions?*

The results of study 3 suggest that management controls may be needed to complement self-control (i.e. the self-regulation of the learning process by the vendor engineer) not because of asymmetric incentives, but because initially low expertise and trust constrain the self-regulation of learning by vendor engineers. While self-control was a highly salient mode of control towards the ends of transitions, it appeared initially hampered by low expertise and trust. Consistent with theory from social and educational psychology, it seemed that novice learners lacked the mental resources to self-control their learning processes (Baumeister et al. 1998; Moos and Azevedo 2008). Moreover, they refrained from self-regulation strategies such as help-seeking to avoid negative ability attributions (Lepine and Van Dyne 2001; Weiner 1985) when trust was initially low. Unlike self-control, formal and clan controls were initially more pronounced, presumably to compensate for the low amounts of self-control.

Collectively, the studies 1, 2, and 3 indicate a complex set of interactions between governance and learning activities. For instance, cognitive load reduction strategies such as task type simplification and supportive information are most needed when the expertise of the vendor engineers is low. Unfortunately, novice vendor engineers will be the least able to self-control their learning process, resulting in rather low amounts of task type simplification and supportive information emanating from self-control. Moreover, social interactions between novice vendor engineers and SME may decrease the experts' trust in the abilities of the vendor engineers when the experts make observations indicating low ability of the vendor personnel. Lower trust may in turn hamper the development of self-control. Put differently, there may be forces that impede cognitive load reduction strategies when they are needed most. This may give rise to a knowledge transfer blockade (Gregory et al. 2009).

2.2.2 Study 4: The Impact of Client Management Decisions on Transition Outcomes

Although the results of study 3 may help explain how governance and knowledge transfer during transition may influence each other over time, more research may be needed to understand how client management actions impact transition outcomes. Two gaps may be noted at that point. First, while study 3 indicated a complex set of dynamic interactions, the outcomes from theses dynamic interactions may not be self-evident. This is because the results of the interactions of complex dynamic systems may not always be easily accessible to human intuition even when the causal links between system elements are known (Forrester 1987). Second, study 3 focused on governance and learning activities during specific time periods of SMOO transitions as level of analysis. While this may have been helpful to capture the dynamics involved, outsourcing research and practitioner interest may rather be situated at the project level of analysis. Client management may want to understand how project-related decisions related to the governance of knowledge transfer influence transition outcomes. For instance, even if client managers are aware of the risk of a knowledge transfer blockade (Gregory et al. 2009), they may be interested in how their project-level decisions can help avoid a knowledge transfer blockade.

Study 4 considered three client management decisions that may influence the dynamics of governance and learning and thus transition outcomes. First, client management may engage in the selection of vendor staff to avoid initially low expertise values due to low amounts of prior related experience (Dibbern et al. 2008; Gregory et al. 2009). Second, they may choose the amount of formal and clan controls related to knowledge transfer (Dibbern et al. 2008; Gregory et al. 2009). Third, they may decide on the duration of coexistence, i.e. the phase during which the SME and the vendor engineers are available to the project. The literature indicates that management risks underestimating the duration during which the support by SME is needed (Dibbern et al. 2008), suggesting that client management struggle to anticipate the effects of coexistence duration. Because these three forms of management involvement may entail costs, it may be insightful to understand their effects. For instance, can higher initial expertise due to effective staff selection procedures reduce the need for formal and clan controls and/or shorten required coexistence phases? Can longer coexistence phases enable successful transitions even when initial expertise is low and formal and clan controls are difficult to implement? Study 4 was intended to help answer these issues by asking:

> ***Research Question 4****: How do the client management decisions (1) on the involvement in staff selection, (2) on organizational controls, and (3) on coexistence duration impact transition outcomes in SMOO?*

3 OVERVIEW OF THE DISSERTATION

This dissertation is an attempt to contribute to a theory of the design and governance of effective knowledge transfer to vendor staff during SMOO transitions. The theory was developed by combining the strengths of different methodological approaches. Rich qualitative longitudinal data were a fruitful empirical basis for this endeavor. Most of the data were collected real-time (Langley 1999) to reduce memory effects. Data collection involved interviews, observation, and document analysis of transitions of software-maintenance roles to vendor on-site coordinators at a Swiss bank. These qualitative data were subject to two groups of data analysis strategies. First, a mixed-methods paradigm was adopted to quantitize the qualitative data. The quantitized data served as an input for quantitative data analysis strategies such as fixed-effects panel regression model and an ordinal regression model. These strategies were useful to the confirmatory part of study 1. Specifically, they helped test whether CLT predicts outcomes in transition and measure the magnitudes of correlations in the data. Second, the rich qualitative data were subject to qualitative data analysis techniques such as temporal bracketing (Langley 1999) and pattern matching (Eisenhardt 1989b; Yin 2009). Qualitative data analysis techniques such as pattern matching were used in the exploratory part of study 1 to theorize how knowledge specificity, a prominent construct in IS outsourcing research (Dibbern et al. 2008), is related to the predictions of CLT on the effectiveness of learning activities. Pattern matching was also applied in the confirmatory research design of study 2 to examine the divergent predictions of two media theories. Study 3 relied on process-oriented qualitative data analysis strategies such as bracketing (Langley 1999) to build theory. Although the results of the studies 1, 2, and 3 illuminated what mechanisms may operate in SMOO transitions and how they may interact with each other, the outcomes of interactions in complex dynamic systems may not be immediately accessible to human intuition even when the mechanisms are known (Forrester 1987; Van de Ven and Poole 1995). Quantitative and qualitative empirical methods were therefore complemented by analytical modeling techniques in study 4, which drew on the system dynamics paradigm and Monte-Carlo simulations.

The dissertation is structured as follows. The second chapter describes the studies 1 and 2, which were concerned with designing effective knowledge transfer to vendor staff. The third chapter includes the studies 3 and 4, which explored the governance knowledge transfer. The fourth chapter summarizes the theory developed in the four studies and provides implications for practitioners. Table I-1 gives an overview of the theoretical foundations of, the methods of, and the author's contribution to the four studies. Next, the studies are briefly summarized.

Study 1 examined the effectiveness of particular learning activities for particular vendor engineers in SMOO transitions. It espoused the view of knowledge transfer as design of a series of learning tasks. In this view, effective learning is enabled by managing the cognitive loads on vendor engineers in such way that each learning task imposes neither too high nor too low cognitive load. Cognitive load was therefore used as the outcome against which to judge the effectiveness of learning activities. The study tested whether the antecedents proposed by CLT are able to predict cognitive load in SMOO settings. Moreover, the study explored how knowledge specificity is related to the predictions of CLT. To this end, a mixed-methods multiple-case study (Teddlie and Tashakkori 2010) drawing on hermeneutic content analysis (Bergman 2010) was conducted. The qualitative data from five transitions at a Swiss bank were quantitized by latent content analysis based on a coding scheme. The resulting numerical data were subject to panel data analysis and ordinal regression in order to test the predictions of CLT on how different learning activities are related to cognitive load. In a next step, the qualitative and the numerical data were combined and pattern matching was applied to explore how knowledge specificity is related to the predictions of CLT. Expertise (operationalized as former experience in the knowledge domains of a particular task), task complexity, and the use of simplified task types were strongly related to cognitive load in the direction anticipated by CLT. Conversely, supportive information showed weak relationships with cognitive load, giving rise to the research question of study 2. The results further suggest that knowledge specificity constrained the initial expertise values because the higher specificity was, the lower was the share of knowledge domains involved in tasks in which vendor staff can have prior experience without having worked for the same client before. This implies that high-specificity cases demand for high amounts of cognitive load reduction strategies because the strategies may partially compensate for low expertise.

Study 2 was conducted to shed light on the weak relationships between supportive information and cognitive load found in study 1. The qualitative data of study 1 suggested that media choice may cause different degrees of effectiveness of supportive information. Study 2 therefore examined how media choice may impact communication performance in supportive-information activities. The study tested the predictions made by two media theories: media synchronicity theory (MST) and the cognitive theory of multimedia learning (CTML). To this end, a confirmatory multiple-case study was conducted, using qualitative data from three additional cases to the data set of study 1. Pattern matching showed support for the prediction of the CTML according to which media capable of using both the auditory and the visual channel in working memory yield higher communication performance, in particular when intrinsic cognitive load (i.e. the ratio of task complexity and expertise) is high. The predictions of MST were not supported by the data. This indicates that media may need to be chosen based on intrinsic cognitive load.

Study 3 explored how knowledge transfer governance may affect and may be affected by knowledge transfer. An exploratory multiple-case study approach (Eisenhardt 1989b; Langley 1999) was applied to the qualitative data of study 1. CLT, control theory, and the knowledge transfer literature were used as sensitizing devices to select a priori constructs (Eisenhardt 1989b). The study capitalized on the in-depth longitudinal data to detect patterns of how the governance of knowledge transfer changed over time and how these changes were associated with learning activities and their outcomes. The results included that self-control was a central element towards the ends of transitions, whereas it may have been initially hampered by low amounts trust and expertise of the vendor engineer. Attributional theory (Lepine and Van Dyne 2001; Weiner 1985) and CLT may explain why low trust and expertise result in weak self-control. Only once higher amounts of trust and expertise manifested were the vendor engineers engaged in high amounts of self-control. This indicates that formal and clan controls may be initially needed to compensate for weak self-control.

The studies 1, 2, and 3 have put forward mechanisms of the interplay of governance and learning activities. The mechanisms suggest that CLT, control theory, and attributional theory (Lepine and Van Dyne 2001; Weiner 1985) may collectively explain the dynamics involved. However, the outcomes of a set of complex dynamics may not always be easily accessible to human intuition (Forrester 1987; Van de Ven and Poole 1995). It may thus not be self-evident how the dynamic interactions of governance and learning activities at different points in time translate into transition outcomes. Moreover, IS outsourcing research and practitioner interest may be situated at the project level of analysis rather than at the level of specific time periods or learning tasks within a project. Client management may want to understand how decisions related to the governance of knowledge transfer in a project relate to the outcomes of transitions. The empirical data of the studies 1, 2, and 3 gave right insights into interactions at levels of analysis embedded within a project, but allowed observing project-level decisions and transition outcomes only in a few particular cases.

Study 4 adopted analytical modeling techniques based on a system dynamics paradigm and on Monte Carlo simulations to help understand how interactions at embedded units of analysis translate into transition outcomes. The models embodied the hypotheses of interactions at embedded levels of analysis established in the first three studies, but its computations allowed to explore how outcomes of these interactions differ when different management decisions are made. Specifically, study 4 investigated how three types of client management decisions may affect these dynamics. The management decisions include whether client management should engage in staff selection to influence the initial expertise values of vendor personnel, how much control management should exert, and how long coexistence phases should be. A Monte Carlo simulation based on the system dynamics paradigm allowed exploring how the three management decisions affect the duration of the transition, i.e. the length of the period after which a vendor engineer is able to independently solve the maintenance tasks. The results suggest that the client management decisions may impact to what extent vicious and virtuous circles manifest during transition. The selection of vendor staff with prior related experience may be the most powerful tool to avoid vicious circles of ineffective learning and negative ability attribution, while the combination of formal and clan controls and longer coexistence phases may mitigate vicious circles.

Table I-1: Overview of Studies

	Study 1: Managing Cognitive Load in Software-Maintenance Offshoring: A Mixed-Methods Study	*Study 2: Media Choice and Communication Performance During Knowledge Transfer in Software-Maintenance Offshoring*	*Study 3: Governing Individual Learning in the Transition Phase of Software-Maintenance Offshoring: A Dynamic Perspective*	*Study 4: Management Decisions in Software-Maintenance Offshoring Transitions: Insights from a Dynamic Model*
Theoretical Foundation	Cognitive load theory	Media synchronicity theory, the cognitive theory of multimedia learning	Control theory, cognitive load theory, the knowledge transfer literature	Control theory, cognitive load theory, attributional theory
Method	Mixed-methods longitudinal multiple-case study	Confirmatory longitudinal multiple-case study	Exploratory longitudinal multiple-case study	Monte Carlo simulation based on the system dynamics paradigm
Data Sources	28 interviews, observation protocols, and archival documents from 5 cases of transitions of software-maintenance roles to on-site coordinators	Data sources of study 1, in addition: 7 interviews and archival documents from 3 further cases of transitions of software reengineering tasks to on-site coordinators and offshore staff	Data sources of study 1, one additional interview; but only 4 out of the 5 cases of study 1 were considered.	-
Data Analysis	Quantitizing of qualitative data through coding, quantitative data analysis through panel analysis and ordinal regression, qualitative data analysis through pattern matching	Coding, pattern matching	Coding, pattern matching	-
Publication Status	To be submitted to Management Information Systems Quarterly; a related paper has been presented at the International Conference of Information Systems 2012 (Krancher and Dibbern 2012)	A prior version has been presented at the Seventh Global Sourcing Workshop 2013.	A prior version has been presented at the 46th Hawaii International Conference on System Sciences 2013 and has been nominated for the best paper award.	To be submitted to the International Conference of Information Systems 2013
Contributors	Krancher, Dibbern	Krancher, Munz, Dibbern, Knolmayer	Krancher, Slaughter	Krancher
Own Contribution	Major	Major	Major	Major

CHAPTER II
LEARNING SOFTWARE-MAINTENANCE TASKS

STUDY 1
MANAGING COGNITIVE LOAD IN SOFTWARE-MAINTENANCE OFFSHORING: A MIXED-METHODS STUDY

ABSTRACT

Software-maintenance offshore outsourcing projects have been plagued by tedious knowledge transfer to vendor staff, in particular when the required knowledge was highly specific to the client. Despite the centrality of knowledge transfer for outsourcing success, the literature gives scant guidance on how effective knowledge transfer to vendor teams can be designed. In this study, we adopt the perspective of cognitive load theory, which posits that low and high cognitive loads on learners need to be avoided for effective knowledge transfer. Moreover, the theory suggests strategies for managing cognitive load. We conducted a mixed-methods multiple-case study of five transition projects to investigate whether the strategies anchored in cognitive load theory can help manage the cognitive loads on vendor engineers and what role knowledge specificity plays in this realm. Our results suggest that simple-to-complex sequencing of tasks based on coordinative complexity and the use of simplified task types such as completion tasks and worked examples were effective load management strategies. These strategies may partially compensate for initially low expertise, the strongest predictor of cognitive load in our data. Against our expectations, supportive information, such as formal presentations, informal discussions, and documents, had an only weak, non-significant negative relationship with cognitive load. Although we derive explanations for the weak relationship from the qualitative data, the results may imply that the choice and design of learning tasks during transition can be more central to outcomes than the communication of knowledge by means of supportive information. Our results also shed light on the role of knowledge specificity. High knowledge specificity constrained initial expertise and therefore entailed a high need for load management strategies. The results imply that knowledge transfer approaches should be designed based on knowledge specificity and overall coordinative complexity.

1 INTRODUCTION

Offshore outsourcing has become a popular strategy for procuring information systems (IS) services (Oshri et al. 2011). Companies relocate IS labor to vendors in remote countries such as India because of domestic labor shortage and wage cost differences. A considerable portion of offshore outsourcing projects refers to software-maintenance services such as correcting software faults and building enhancements to existing systems. While offshore outsourcing standard services approaches saturation, observers see a recent trend towards offshore outsourcing highly knowledge-intensive software services (Booth 2013).

Yet, many knowledge-intensive software-maintenance offshore outsourcing (SMOO) projects do not meet the initial expectations. Tedious knowledge acquisition by vendor teams lies at the heart of a considerable fraction of the unexpected costs (Dibbern et al. 2008). Consistent with the knowledge-based view of the firm (Conner and Prahalad 1996), Dibbern et al. (2008) found that cognitive limitations of vendor personnel explained extra costs to a greater extent than opportunistic vendor behavior did, in particular when knowledge specificity (the degree to which the required knowledge is specific to the client) was high. Cognitive limitations may be particularly salient during the transition phase, which succeeds the signing of the contract and during which the ownership of activities is transferred to the staff of the offshore unit (Chua and Pan 2008; Tiwari 2009). During transition, these engineers may frequently feel overloaded by the amounts of novel information that they encounter in the domain of their new software-maintenance task (Chua and Pan 2008). As a consequence, they may fail to take over tasks according to the plans made prior to transition (Chua and Pan 2008; Dibbern et al. 2008). Knowledge transfer—the process (Szulanski 1996) through which the vendor engineers acquire the knowledge to perform the software-maintenance tasks—is thus frequently tedious. The central role of cognitive limitations during knowledge transfer is not surprising. Software maintenance has been described as a cognitively demanding task, in which engineers heavily rely on their expertise to identify where maintenance actions need to be made and to conceive solutions (Von Mayrhauser and Vans 1995). However, such expertise may be scarce when vendor engineers take over the maintenance of a client's software system.

Although knowledge transfer is thus frequently problematic (Chua and Pan 2008; Dibbern et al. 2008; Gregory et al. 2009; Wende and Philip 2011) and at the same time essential for project outcomes (Westner and Strahringer 2010), the literature gives scant guidance on how effective knowledge transfer can be designed in SMOO transitions. Prior work described the learning activities in which offshore engineers typically engage during transitions such as formal presentations, document study, job-shadowing, and on-the-job training (Chua and Pan 2008; Nicholson and Sahay 2004). Moreover, the literature has produced valuable findings on when experts will communicate their knowledge in interorganizational settings (Ko et al. 2005; Kotlarsky and Oshri 2005; Oshri et al. 2008). However, the accounts of cognitively overloaded offshore engineers and the central role of absorptive capacity (the recipient's ability to assimilate and apply outside knowledge) in interorganizational knowledge transfer (Cohen and Levinthal 1990; Dibbern et al. 2008; Ko et al. 2005; Szulanski 1996; Szulanski 2000) suggest that the communication of knowledge may frequently not be a sufficient condition for skillful action. A better understanding of knowledge transfer in SMOO may thus hinge on understanding the knowledge recipients' cognitions. Yet, their cognitions have received limited attention in the existing literature (with the exception of Vlaar et al. 2008). Given the central role of cognitive constraints in SMOO transitions, the stakeholders involved in SMOO projects may want to understand whether they can design knowledge transfer in such a way that the vendor engineers can effectively acquire the required knowledge even though cognitive constraints exist. For instance, can cognitive overload on vendor engineers be avoided if vendor engineers study pertinent documentation or work on specific kinds of tasks? This study aims at providing theoretically grounded answers to such issues by addressing the following research question:

> *What determines the effectiveness of particular learning activities during SMOO transitions?*

Recent research in educational psychology may be a fruitful base to address this research question. Cognitive load theory (CLT; Sweller et al. 1998; Van Merriënboer and Sweller 2005) has emerged as a widely acknowledged perspective (Ozcinar 2009) to predict early skill acquisition in complex domains such as technical trouble-shooting. In a substantial series of controlled experiments, CLT research has repeatedly found that complex skills are acquired most effectively when high and low cognitive loads (i.e. the cognitive demands that a particular task impose on a particular learner) are avoided. Because the risk of high cognitive load is substantial during the acquisition of complex skills, CLT researchers have developed a set of strategies for managing cognitive load (Van Merriënboer et al. 2002; Van Merriënboer et al. 2003). These strategies operate at the level of a learning task (i.e. a real or fictitious task that is realistic for the domain). In this view, a learner may work on a series of learning tasks. Because expertise may evolve over time in this process and because the tasks may refer to different knowledge domains, the need for cognitive load management strategies may be different for each task. CLT thus suggests that vendor staff may effectively acquire knowledge when they engage in a series of learning tasks in which the cognitive loads are managed to be at a moderate level. Cognitive load is thus a central outcome against which to judge the effectiveness of particular learning activities.

This study adopted an integrated mixed-methods approach (Teddlie and Tashakkori 2010) in order to apply the perspective of CLT to knowledge transfer in SMOO. The confirmatory part of the study examined whether the cognitive load management strategies proposed by CLT were effective to manage the cognitive loads associated with the learning tasks embedded in five SMOO transitions at a Swiss bank. To this end, qualitative data was quantitized (Tashakkori and Teddlie 1998) based on a coding scheme and was subject to quantitative methods such as a fixed-effects panel model. The results suggest that two strategies were effective to manage cognitive load: (1) Choosing tasks with lower coordinative complexity and (2) using simplified task types such as completion tasks and worked examples. These strategies may have partially compensated for initially low expertise, the strongest predictor of cognitive load in our data. Contrary to our expectations, supportive information, such as formal presentations and documents, had a weak, non-significant negative relationship with cognitive load.

The exploratory elements of the mixed-methods study complemented these results in two major ways. First, they provided explanations for the weak relationship of supportive information and cognitive load. Second, they helped understand how knowledge specificity, a central knowledge-related construct of IS outsourcing research (Dibbern et al. 2008), is related to the predictions of CLT. This also involved expanding the level of analysis from the level of the learning task embedded within a project to the level of the knowledge transfer project, at which knowledge specificity is situated. Knowledge specificity was found to constrain the initial expertise values because under high specificity, the number of knowledge domains in which vendor engineers can have prior experience is lower than under low specificity. Given the strong relationship between expertise and cognitive load, high-specificity projects may face particularly high needs for cognitive load management strategies. These findings have implications for how effective knowledge transfer to vendor engineers in SMOO can be designed.

The paper is structured as follows. In section 2, we present the predictions made by CLT. In section 3, we describe the mixed-methods approach adopted to test the predictions and explore the role of knowledge specificity. We then present and discuss the results.

2 THEORY

CLT predicts cognitive load and skill acquisition for complex tasks (such as software maintenance) in contexts that risk imposing heavy cognitive load on individuals (such as SMOO transitions) (Sweller et al. 1998; Van Merriënboer and Sweller 2005). The theory may therefore be well suited to explain the role of cognitive limitations during knowledge transfer in SMOO transitions. This section gives a brief overview of the main assumptions of the theory before the hypotheses are presented.

2.1 Cognitive Load Theory

CLT bases its predictions on a widely accepted framework of the human cognitive architecture (Sweller et al. 1998; Van Merriënboer and Sweller 2005). It assumes a severely limited working memory and a virtually unlimited long-term memory (Baddeley 1992). When humans process complex information, they need to keep the information elements in working memory while they try to establish relationships between the elements (Sweller and Chandler 1994). However, working memory capacity may suffice for combining only two to four elements at the same time (Sweller et al. 1998; Van Merriënboer and Sweller 2005). This is why many real-world problems will overload humans unless they hold schemas in long-term memory that support their information processing. Experts have such powerful schemas in the domains of their expertise, which enable them to aggregate information to higher-order and therefore less numerous chunks (Chase and Simon 1973). For instance, experienced software engineers solve problems by drawing on powerful schemas of general programming knowledge and software-specific knowledge (Von Mayrhauser and Vans 1995). In CLT, learning is the acquisition of such expertise, hence the acquisition and automation of schemas that enable learners to solve more and more complex problems (Sweller et al. 1998; Van Merriënboer and Sweller 2005).

It is a main premise of CLT that the acquisition of schemas itself demands working-memory capacity (Van Merriënboer and Sweller 2005). This complicates the learning of complex problem-solving tasks. Novices' working memory may be overstrained by the cognitive load intrinsic to a given problem-solving task. They may thus lack additional working memory capacity for schema acquisition. As an unfortunate consequence, these novices may not only perform badly in the problem-solving task; their low performance may also stagnate over time because cognitive overload prevents schema acquisition. Learning environments should therefore be designed in a way that high cognitive loads are avoided (Kirschner et al. 2006; Sweller et al. 1998; Van Merriënboer and Sweller 2005). This does not imply that learners should not be confronted with realistic tasks such as real software defects or software enhancements. Cognitive load theorists concur with other theories on the benefits of learning tasks, which may be defined as "authentic whole-task experiences based on real-life tasks" (Van Merriënboer and Kirschner 2007, p. 14). However, CLT qualifies that learning tasks are only effective to the extent that they do not overload the learner. Three strategies are recommended to manage the cognitive load on learners: simple-to-complex sequencing, using simplified task types, and supportive information (Van Merriënboer et al. 2003). CLT therefore agrees with the prior literature on knowledge transfer and SMOO in that prior knowledge is highly influential for knowledge acquisition (Cohen and Levinthal 1990; Dibbern et al. 2008; Ko et al. 2005; Szulanski 1996; Szulanski 2000). But CLT extends this perspective by suggesting how the difficulties imposed by the lack of prior experience may be overcome.

2.2 Theoretical Model

We apply CLT to study knowledge transfer in SMOO. In this perspective, knowledge transfer is a sequence of learning tasks and supportive-information activities in which each learning task yields a particular cognitive load outcome. Effective learning then requires designing each learning task and related supportive information in such a way that the task places only moderate cognitive load on the learner. Understanding the antecedents to cognitive load is therefore essential for knowledge transfer.

CLT suggests that cognitive load is determined by the expertise of the learner, by the intrinsic complexity and the type of the learning task, and by the supportive information related to the learning task. Figure II-1 shows how these factors are hypothesized to influence cognitive load. The figure also illustrates that CLT makes predictions at the level of the learning task. Many learning tasks may be embedded within one knowledge transfer project, i.e. a transition of a software-maintenance role to one vendor engineer. Conversely, CLT does not make any predictions on how a project-level construct prominent in IS outsourcing research, knowledge specificity, is related to these predictions. We therefore did not specify any hypotheses with regard to knowledge specificity, but explored this issue inductively.

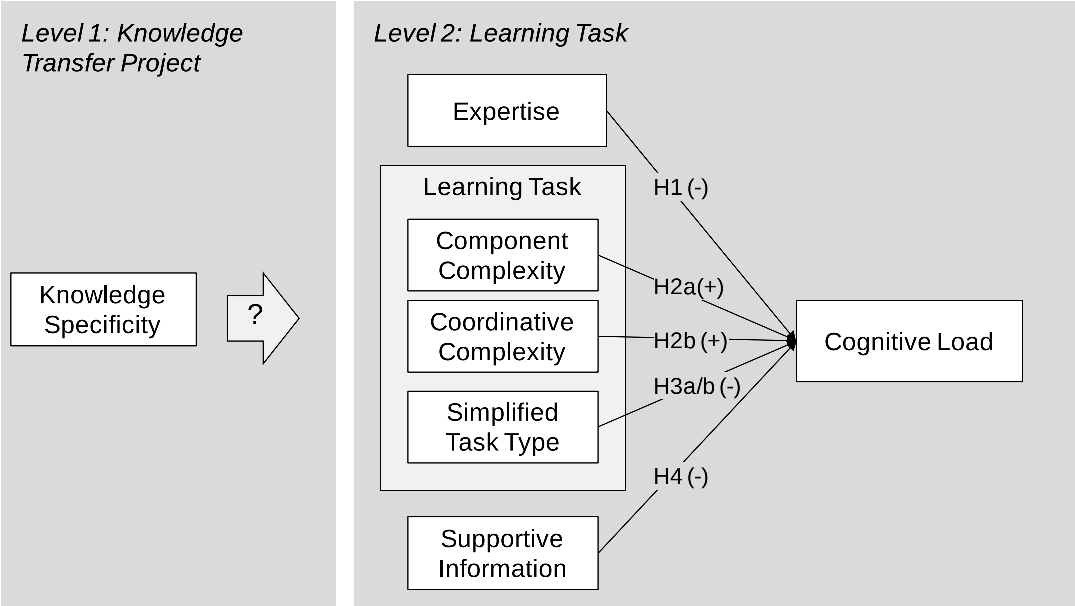

Figure II-1: Theoretical Model

The learner’s *expertise* is one driver of cognitive load. Expertise is held to decrease cognitive load because powerful schemas in long-term memory help aggregate information to higher-order and therefore less numerous chunks (Chase and Simon 1973). The expertise literature emphasizes two characteristics of expertise. First, expertise is highly domain-specific (Chase and Simon 1973; Ericsson et al. 1993). Experts do not perform better than novices outside the domains of their expertise. Second, expertise is acquired gradually through months and years of deliberate practice in the domain (Ericsson et al. 1993). It is therefore distinct from the understanding of a software system that is gained by reading a document or attending to a formal presentation. Following these arguments, vendor engineers who have prior experience in maintaining very similar software applications will perceive less cognitive load than vendor engineers with less experience with the particular software. Empirical studies on software maintenance and outsourcing lend support for this claim (Dibbern et al. 2008; Espinosa et al. 2007). This suggests:

H1: The higher the expertise of the vendor engineer, the lower is cognitive load.

The *complexity* intrinsic to a learning task is a further antecedent to cognitive load in CLT (Sweller and Chandler 1994; Sweller et al. 1998). The higher task complexity, the more information elements need to be processed at the same time. This increases the demands on working memory and thus cognitive load according to CLT. The software-maintenance literature has primarily used two dimensions of complexity in software environments: component complexity and coordinative complexity (Banker et al. 1993; Banker and

Slaughter 2000; Espinosa et al. 2007; Wood 1986). Component complexity refers to the number of distinct information elements and acts involved in a task. One may expect that, the more information elements and acts are involved in a task, the more elements need to be processed at the same time and thus the higher is cognitive load. Coordinative complexity refers to the number and strengths of interdependence relationships between the information elements and acts involved in a task. The more numerous and stronger the interdependence relationships are, the higher may be the probability that learners need to simultaneously process a high number of related information elements, resulting in higher cognitive load. We therefore anticipate:

H2a/b: The higher (a) component complexity and (b) coordinative complexity, the higher is cognitive load.

Given the role of task complexity, transition managers may relieve cognitive load by purposefully assigning tasks of rather low complexity at the beginning of transitions. However, this may not be feasible or sufficient under all circumstances. CLT suggests two further strategies to reduce cognitive load beyond manipulations of task complexity by task assignment. The first strategy is to use simplified task types.

CLT distinguishes several *task types* that differ in the extent to which the solution process or solution product are given to the learner and to which goal conditions of problems are relaxed (see Table II-1). In conventional tasks, learners are given an initial problem state, a desired end state, and are asked to identify the solution path from the given problem state to the desired end state. For instance, a vendor engineer may be given a requirements document (the desired end state) and the software (the given state) and she may be asked to independently design, implement, and test the modification to the software system (the solution path). Conventional tasks are expensive in terms of working-memory demands because the number of possible permutations of the information elements quickly explodes after a threshold of three or four elements is exceeded (Sweller et al. 1998). Conventional tasks thus result in high cognitive load and weak learning for novice learners according to CLT (Kirschner et al. 2006; Sweller et al. 1998).

Simplified task types are therefore the hallmark technique of effective CLT-based training. In worked examples, for instance, learners are given the full solution to a problem and are asked to study the solution. For example, a vendor engineer may study the solution to past maintenance requests or observe how an expert solves a maintenance problem. According to CLT, worked examples dramatically reduce working memory demands because they guide learners along the solution path and thereby avoid the combinatorial explosion in the search processes that are frequently associated with conventional tasks. We consider worked examples as full task type simplification. Other task types do not fully explicate the solution path, but may still relieve cognitive load. These task types include completion tasks, imitation tasks, and goal-free tasks (Van Merriënboer et al. 2002; Van Merriënboer et al. 2003). In completion tasks, a part of the solution process or product is given to the learner. For example, a client expert may take over the design of a maintenance request and leave only its implementation and testing to the vendor engineer (Dibbern et al. 2008). In imitation tasks, the learner is provided with the solution to a similar task. This is the case when a client expert indicates the vendor engineer how a similar maintenance request has been solved. In goal-free tasks, the goal conditions for a problem are relaxed, obviating the need for expensive search processes to a specific goal state. For instance, a vendor engineer may be asked to create a document on a software component. The engineer may then write what she knows and what she can learn about the software from reading the code without having to engage in search processes towards a specific goal state. We consider completion tasks, imitation tasks, and goal-free tasks as instances of partial task type simplification because they do not fully specify the solution, but simplify problem-solving. We submit:

H3a: Partially simplified task types are associated with lower cognitive load than conventional tasks.

H3b: Fully simplified task types are associated with lower cognitive load than partially simplified task types.

Table II-1: Task Types (adapted from Van Merriënboer et al. 2002)

Task Type Category	*Task Type*	*Goal State*	*Solution*
None	Conventional Task	Given	Not given
Partial Simplification	Completion Task	Given	Partially given
Partial Simplification	Imitation Task	Given	Solution to analog problem given
Partial Simplification	Goal-free Task	Relaxed	Not given
Full Simplification	Worked example	Given	Given

It is important to distinguish intrinsic task complexity from the use of simplified task types. Intrinsic task complexity refers to the complexity that is intrinsic to the problem addressed in the learning task irrespective of the task type. For instance, redesigning the interface of a central software component may be considered a complex problem because many software objects may need to be considered at the same time. A vendor engineer may either study how the SME solved this problem (a worked example), or implement the solution based on the design given by the SME (a completion task), or design and implement the solution herself (a conventional task). The complexity intrinsic to the problem is the same in all three cases, but the task type differs. While there may be limits in manipulating intrinsic task complexity e.g. when all maintenance tasks for a given role are rather complex, there is potentially full discretion in the choice of task type as long as the SME are available to the project.

Supportive information is a further strategy to reduce cognitive load. Supportive information is “supportive to the learning and performance of non-recurrent aspects of learning tasks” (Van Merriënboer et al. 2002, p. 43). It provides blueprints for schemas that can be used when working on learning tasks (Van Merriënboer et al. 2003). These schemas can be activated during the work on learning tasks and thereby reduce the cognitive load imposed by the tasks. Documents, face-to-face presentations, and informal discussions in SMOO projects (Blumenberg et al. 2009; Chua and Pan 2008; Tiwari 2009) may be examples of supportive information. Building codified knowledge repositories is thus a strategy to make supportive information available to vendor engineers. This suggests:

H4: The more supportive information is consulted by the vendor engineer, the lower is cognitive load.

Taken together, CLT predicts that vendor engineers who are unfamiliar with a particular software system will be prone to cognitive overload, but that high cognitive load may be avoided by three load management strategies: manipulating task complexity by a simple-to-complex sequencing strategy, using simplified task types, and providing supportive information. Although these predictions have been established through controlled experiments, we lack empirical evidence on whether they apply to the context of SMOO projects. Furthermore, we lack evidence of the strengths of the relationships in SMOO. However, client and vendor managers may be interested to understand, for instance, whether supportive information in knowledge repositories may compensate for fluctuations of expertise due to personnel turnover. Finally, the existing literature is silent on how knowledge specificity enters into the predictions made by CLT. The existing IS outsourcing literature ascribes a central role to knowledge specificity (Dibbern et al. 2008; Larsen et al. 2012).

Drawing on the knowledge-based view of the firm (Conner and Prahalad 1996), these studies indicate that knowledge-related issues are more likely under high knowledge specificity. It appears thus desirable to understand how knowledge specificity is related to the explanatory framework suggested by CLT. Yet, given that CLT has not yet been applied to IS outsourcing research to the best of our knowledge, we lack theoretical links between CLT and knowledge specificity, calling for exploratory research. We next describe the mixed-methods procedures undertaken address these gaps in knowledge.

3 METHODS

3.1 Research Design

We conducted an integrated mixed-methods study (Teddlie and Tashakkori 2010). The study involved collecting qualitative data, transforming the qualitative data into numbers by quantitizing (Teddlie and Tashakkori 2010), and analyzing the quantitized and the original qualitative data using qualitative and quantitative techniques on two levels of analysis: the learning task and the knowledge transfer project. Table II-2 gives an overview of the research design. Mixed-methods approaches based on quantitizing qualitative data have also been used in the past to study knowledge flows in software contexts (Slaughter and Kirsch 2006).

A mixed-methods approach was chosen for the purposes of expansion and complementarity (Greene et al. 1989). The research objective of this study implied a need for expansion beyond the domains of either qualitative or quantitative strategies. In particular, a need for expansion was indicated because applying the framework of CLT to our overarching research question entails a confirmatory and an exploratory subquestion (see Table II-2). Mixed-methods studies allow answering both confirmatory and exploratory questions within the same study (Plano Clark and Badiee 2010), expanding thereby the analytical scope beyond what can be achieved by either qualitative or quantitative strategies. Moreover, the two subquestions span two levels of analysis: the learning-task level prevailing in CLT research (subquestion 1) and the project level, prevailing in outsourcing research, to theorize on knowledge specificity (subquestion 2). Mixed-methods studies are one approach to link multiple levels of analysis (Teddlie and Tashakkori 2010). The use of mixed methods also followed a purpose of complementarity in two ways. First, consistent results of qualitative and quantitative analysis strategies helped increase the internal validity of our findings (Yin 2009). Second, qualitative analysis strategies helped obtain clarification and illustration for unexpected quantitative findings.

Table II-2: Research Design

	Subquestion 1	*Subquestion 2*
Question	Does the CLT framework allow predicting cognitive load in SMOO transitions?	How is knowledge specificity related to these predictions?
Research Question Type	Confirmatory	Exploratory
Unit of Analysis	Learning task (nested within a knowledge transfer project)	Knowledge transfer project
Data	Quantitized qualitative data (level: learning task), raw qualitative data for clarification and illustration	Quantitized data (level: knowledge transfer project), raw qualitative data for trian-gulation

The study used two levels of analysis (see also Table II-2). The knowledge transfer to one vendor engineer represented one case. A learning task within a case were the embedded units of analysis (Yin 2009). We adopted a theoretical sampling strategy (Yin 2009) by including three cases with rather high knowledge specificity and two cases with rather low knowledge specificity. This should help explore how knowledge specificity enters into the predictions of CLT. We also aimed at literal replication (Yin 2009) by including more than one case in each specificity category. This allowed saturation at the end of the data analy-sis. We did not pursue any purposeful sampling strategy at the level of the learning tasks. It was expected that learning task configurations would naturally vary, e.g. with regard to task complexity because of the stochastic nature of incoming software-maintenance re-quests. We also anticipated that expertise would vary within the cases because the knowledge domains required for a particular task would vary stochastically and because we expected an increase of expertise over time. Our longitudinal data collection covered a process of between three and five months in each knowledge transfer to allow some in-crease and thus variation in expertise. Although there was random variation in our inde-pendent variables, the study's statistical generalization is subject to the limitations that affect single-site quantitative studies (Lee and Baskerville 2003).

Table II-3 gives an overview of the cases. The transitions were conducted on site in Swit-zerland on the premises of a Swiss bank, which represented the client in all five knowledge transfers. The study only included on-site transitions because this was a natural control for factors not in scope of our theoretical model (such as geographical distance), because transitions are frequently conducted on site (Dibbern et al. 2008; Tiwari 2009) and because the knowledge transfer to these on-site staff has been suggested to be central for the success of SMOO projects (Gregory et al. 2009). The bank operated globally, held assets of over $1 trillion in 2011, and had considerable experience in offshoring IS work to India. Whereas the cases 2 and 3 were transitions from one vendor to another vendor, client teams were augmented by vendor staff in the cases 1, 4, and 5. The three vendors involved in the study were among the major Indian service providers. Each of the transitions 1, 2, and 3 referred to a different custom-built data warehousing application. These cases were considered high in specificity. Each of the cases 4 and 5 referred to the same software system, an instance of software package for controlling financial transactions. Because substantial knowledge of the commercial-of-the-shelf software package was involved in the maintenance tasks, the cases 4 and 5 were considered rather low in specificity. All knowledge transfers were considered successful by all stakeholders upon completion.

Table II-3: Cases

Case	*SME*	*Vendor Engineer*	*Software Application*	*Length of Process Captured by Data*
1	Four Swiss or Germans (client)	4 years of experience in data warehousing and in software projects, vendor A	Data warehousing application 1	5 months
2	One Indian (main SME, vendor C), one Swiss (client)	6 years of experience in data warehousing, 11 years in software projects, vendor A	Data warehousing application 2	3 months
3	Two Indians (main SME, vendor C), one Swiss (client)	5 years of experience in data warehousing, 11 years in software projects, vendor A	Data warehousing application 3	5 months
4	Three Swiss (client)	4 years of experience in the software package and in software projects, Vendor B	Implementation of a software package for the control of financial transactions	5 months
5	Three Swiss (client), one Indian (vendor B)	1 year of experience in the software package, 5 years in software projects, vendor B		3 months

3.2 Data Collection

Data were collected through semi-structured interviews, observation of sessions, and document analysis based on a case-study protocol (Yin 2009, p. 79). Table II-4 gives an overview of the data sources. In the interviews, the vendor engineers were asked to describe how they worked on a task and in what additional activities related to knowledge transfer they engaged. We conducted several interviews with the same vendor engineers at different points in time. This was intended to reduce the impact of memory effects on the longitudinal data. In addition, several encounters with the same project participants helped to build trustful relationships that allowed the participants to openly share their perceptions. Consistent with our goal of illuminating the vendor engineers' cognitive processes, these interviews thus helped gain rich insights into how vendor engineers work on a series of tasks during transition. Interviews with subject-matter experts and client management were the primary source for developing a coding scheme of task complexity and for coding the complexity of the tasks. However, they also served to triangulate the information provided by the vendor engineers. All interviews were tape-recorded and transcribed.

Table II-4: Data Sources

Case	*Interviews: No. of interviews/ No. of interviewees*[1]			*No. of Observed Sessions*	*No. of Documents*	*Data Points*	*Timing of Data Collection*
	Vendor engineer	*SME*	*Managers*				
1	5/1	2/2	2/1	4	20	19	Real-time
2	2/1	2/2	3/3	2	8	8	Real-time
3	3/1	2/2		2	16	16	Real-time
4	2/1	2/2	1/1	2	2	6	Retrospective
5	2/1					7	Real-time

A second data collection technique was the observation of sessions at the client's premises. In the cases 4 and 5, the sessions were formal presentations about components of the application, whereas they were coached knowledge elicitation sessions in the cases 1, 2, and 3[2]. In the latter, a coach of the client firm facilitated codifying knowledge based on a methodology adopted by the client (see Ackermann 2011 for details). The observations resulted not only in field notes, but also in an understanding of the software systems. Interview transcripts and observation notes amounted to 112,725 words.

Documents were a third data source. The documents studied included requirements specifications, design documents, peer review feedback, defect extracts, documents created as a result of knowledge elicitation sessions, software documentation, knowledge transfer

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

[2] Both types of sessions were coded as supportive information. See Appendix II-1 for the coding scheme.

plans, and email notes. When data from multiple sources of evidence diverged, clarifying questions were addressed in subsequent interviews. Using multiple sources of evidence was a strategy to increase construct validity (Yin 2009, p. 41).

3.3 Data Analysis

We drew on hermeneutic content analysis as a strategy for integrating mixed methods based on qualitative data (Bergman 2010). Figure II-2 shows the data analysis process. In hermeneutic content analysis, qualitative data is first quantitized (step 1) for the purpose of statistical analysis (step 2). The results of the statistical analysis are then subject to qualitative analysis to recontextualize the quantitative results within the context of the cases (step 3). We next describe the three steps of the data analysis process.

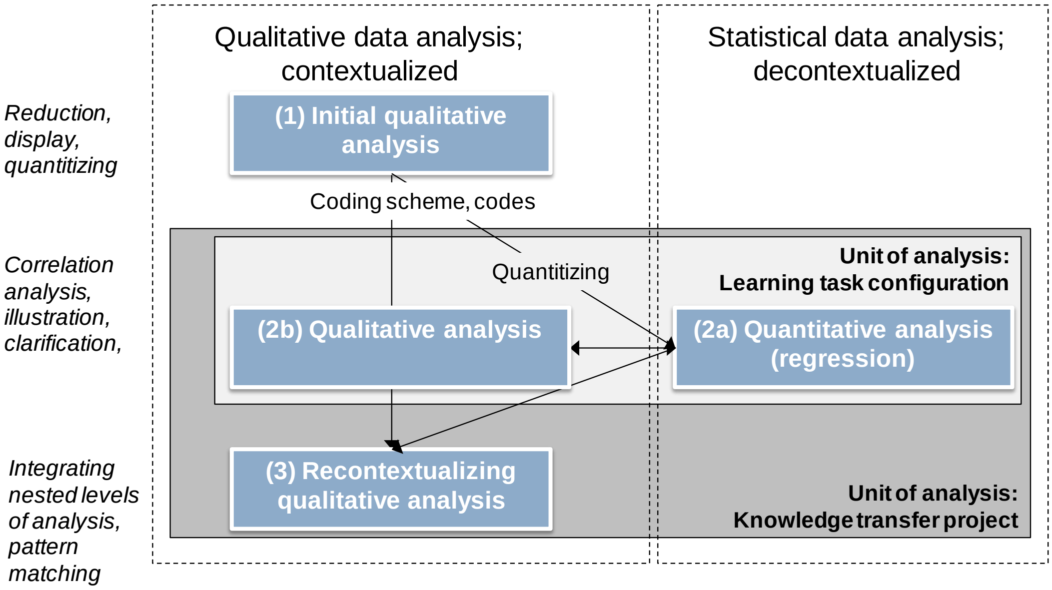

Figure II-2: Data Analysis

3.3.1 Step 1: Initial Qualitative Analysis (Coding and Quantitizing)

In a first step, data were coded in NVivo 9. In the first coding run, data were coded to categories that represented the constructs of the theoretical model. Thus, this first coding run aimed at organizing material according to conceptual categories. During coding, event-flow networks (Miles and Huberman 1994) were drawn. They depicted the events in each of the cases and related these events to time. Events included the starts of transitions; supportive-information activities such as knowledge-elicitation sessions, informal discussions, and formal presentations; the start and the end of the work on a particular learning task; and any other events that were considered relevant for the vendor engineer's learning process. In addition, the networks showed relationships between events. For instance, when informal discussions were related to a particular task, an arrow between both events was drawn. The event-flow networks were validated by the vendor engineers. They helped unitize data by identifying learning tasks and relating supportive-information activities to learning tasks. We found 56 distinct learning task configurations that were associated with cognitive load outcomes. They were used as the data points for the quantitative analysis to answer research question 1 (see also Table II-4 for the number of data points per case).

Next, a coding scheme was developed for the purpose of quantitizing the organized qualitative data. Table II-5 gives an overview of the operationalization of the constructs. Appendix II-1 provides more details. We next briefly describe how expertise and cognitive load were coded.

Table II-5: Operationalization of Constructs

Construct	*Operationalization*
Expertise	Weighted average of the natural logarithms of the numbers of prior related experiences in all knowledge domains that were relevant for a particular learning task. Weighting involved determining expertise values for each of the categories of the IS book of knowledge (BoK) (Iivari et al. 2004) (application knowledge, application domain knowledge, IS development process knowledge, technical knowledge, organizational knowledge) and multiplying these values by weights that reflect the relative importance of each category in each case. The relative importance of the categories was estimated based on the fraction of the number of codes of each category over the number of codes for all five categories. See the Appendices II-1 and II-3 for more details.
Task Type	Coded by two dummy variables: (1) At least partial simplification: The task is partially simplified (completion task, imitation task, goal-free task) or fully simplified (worked example). (2) Full simplification: The task is fully simplified (worked example).
Supportive Information	The estimated time (in hours) dedicated to consulting supportive information that is related to a particular learning task; calculated as the product of the number of weeks during which the supportive information was consulted, the number of days per week on which the supportive information was consulted, and the estimated duration per day dedicated to supportive information classified according to four categories: (1) very short duration (up to 5 minutes; numerical value: 2.5/60), (2) short duration (more than 5, but at most 30 minutes; numerical value: 17.5/60), (3) medium duration (more than 30 minutes, but less than 90 minutes; numerical value: 1), (4) long duration (90 minutes or more; numerical value: 1.5). When supportive information referred to more than one learning task, it was equally distributed to the related learning tasks.
Coordinative Complexity	Coded on a 3-points-scale based on the coding scheme in Appendix II-1
Coordinative Complexity	Coded on a 3-points-scale based on the coding scheme in Appendix II-1
Cognitive Load	Coded on a 5-points-scale based on mental effort and task performance (see Appendix II-1 for details)

Expertise. Expertise was operationalized as the average logarithmic amount of prior experiences in the specific domains of the learning task. This is consistent with the contention of expertise research that expertise develops through practice (Ericsson et al. 1993). Coding involved determining the knowledge domains in each case and creating a matrix of associations between knowledge domains and learning tasks. Determining associations between knowledge domains and tasks was facilitated by evidence from the interviews in which the vendor engineers were asked to relate tasks to elements of the knowledge maps that were created during the knowledge elicitation sessions. We obtained between 50 and 65 knowledge domains in each case. The knowledge domains were assigned to the five categories in the IS Body of Knowledge (IS BoK) (Iivari et al. 2004) to allow weighting the importance of the domains based on the number of codes of each category in each knowledge transfer project. The associations between learning tasks and knowledge domains and the information on prior experience in the domains allowed counting the number of previous experiences in each knowledge domain for each learning task. Because gains from experience decrease with the cumulated amount of experience according to the expertise literature (Ericsson et al. 1993), we applied the natural logarithm to the amount of prior experience in each knowledge domain. The expertise for one particular learning task was then calculated as the average of the logarithms of prior experiences in each domain relevant for the learning task. The intuition behind first applying the logarithm and only then calculating the average of the logarithms was that we considered expertise as a characteristic of the interaction of a knowledge domain and a person. The vendor engineer therefore had expertise values in each of the knowledge domains. Each expertise value is estimated by the logarithm of the number of prior experiences in the domain. Then expertise for one task is the calculated as the average of the expertise values in all domains of the task. This is consistent with the view of learning in CLT as the acquisition and elaboration of schemas specific to the domains of the task (Sweller et al. 1998). Sensitivity analyses with alternative expertise measures corroborated these assumptions and are reported in Appendix II-3.

Cognitive Load. Cognitive load was measured based on mental effort and task performance (Paas et al. 2003). Mental effort was coded based on statements of perceived complexity and of the type of problem-solving heuristic or algorithm reflected in the interview statements such as means-ends-analysis or forward-working problem-solving approaches.

The reliability of the coding procedure was tested by comparing the coding with the coding of a student who was not familiar with the hypotheses of the study. To this end, the student was first trained with fictitious data. A first set of randomly selected learning task configurations was then coded independently. Disagreements were resolved by consulting supplementary information from the data or by eliminating ambiguities in the coding scheme. A second set of randomly selected learning task configurations was coded for the dependent variable *cognitive load* and for the constructs in which the first data set did not yield satisfactory results. Table II-6 shows the reliability results. Cohen's kappa is given for ordinal variables; Cronbach's alpha is given for variables that were assumed metric. A Cohen's kappa value of larger than 0.6 denotes substantial agreement (Landis and Koch 1977). A Cronbach's alpha value of larger than 0.9 denotes excellent consistency (Kline 1999). The second data sets exceeded these thresholds for all constructs expect for expertise. Expertise was only coded by the first author because relating the tasks to knowledge domains benefited from knowledge of the software systems.

Table II-6: Reliability Results

	First data set			*Second data set*		
	Agr.	*Kappa*	*Alpha*	*Agr.*	*Kappa*	*Alpha*
Cognitive Load	.69	-	.95	.69	-	.95
Component Complexity	.86	-	.93	-	-	-
Coordinative Complexity	.80	-	.89	1.00	-	1.00
Supportive Information (Days per Week)	.64	-	.80	.85	-	.93
Supportive Information (Duration per Day)	.82	.76	-	.90	.80	-
Task Type Category	.69	.15	-	.86	.73	-

(Agr. = Agreement, Kappa = Cohen's Kappa, Alpha = Cronbach's Alpha)

3.3.2 Step 2: Quantitative and Qualitative Analysis of the Learning Task

The quantitized data at the level of the learning task were then subject to statistical analysis to answer research question 1. We used two statistical techniques to this end. First, we ran ordinal regression analyses. We regressed cognitive load initially only on four case dummy variables to control for any influence on the level of the knowledge transfer project (such as cognitive ability or any individual attitudes). We then added the independent variables of our model. All non-dichotomous independent variables were standardized to help compare the magnitudes of relationships. Ordinal regression was chosen because we lacked evidence that our measure of cognitive load resulted in a metric scale. The nested nature of the data was, however, not consistent with the assumption of independence of the drawings. We therefore ran a fixed-effects panel model[3] to assess the bias that may result from this violation. This model included a random case-specific intercept. It thus explained variation from the case-specific means of cognitive load in each case by constant fixed effects of the independent variables. This technique is thus largely immune to the bias caused by nested data. The results of the two techniques were then compared. The qualitative data were used to clarify ambiguities that emerged from the quantitative analysis such as explanations for the weak relationship between supportive information and cognitive load.

3.3.3 Step 3: Quantitative and Recontextualized Qualitative Analysis at the Level of the Knowledge Transfer Project

In a third step, the results of the quantitative and qualitative analyses were consolidated and aggregated to the level of knowledge transfer projects in order to address research question 2. This involved diagramming the evolutions of the variables over time across cases. Interview statements on causal relationships, the aggregated quantitative data, and the diagrams were used to explore how knowledge specificity entered into the predictions of CLT.

Although our data gathering and analysis focused on the constructs of our theoretical framework, the interview questions were also related to other constructs that have been found influential in prior research such as the vendor engineers' and subject-matter experts motivation; cultural, semantic and geographic distances; encoding competence; relationship quality; and organizational controls. While qualitative evidence suggests that they may have influenced the independent variables of our study, we did not find substantial indications for them to relate to cognitive load or to moderate our predictions.

4 RESULTS

The results section is organized as follows. We first present the results of hypotheses testing. The results show strong associations of expertise, coordinative complexity, and simpli-

[3] Although the panel model assumed a metric dependent variable, the relatively homogeneous step sizes in the thresholds of the ordinal regressions suggest that this assumption may not be problematic in our case.

fied task types with cognitive load, but indicate a comparably weak role of supportive information. We then provide qualitative evidence that may explain the weak relationship. Given the pivotal role of expertise in our statistical results, alternative expertise measures were compared to examine the robustness of the results and the theoretical assumptions that underpin the measure. The corroboration of the expertise measure helps then to explore how knowledge specificity is related to the predictions of CLT. Our analysis suggests that specificity constrains the start values of expertise and thus drives the need for load reduction strategies.

4.1 Statistical Analysis of Learning Task Configurations

Table II-7 shows the descriptive statistics and the zero-order correlations. The figures show that the projects made ample use of simplified task types and supportive information. 80% of the learning tasks were at least partially simplified. The vendor engineers dedicated more than eight hours per learning task to consulting supportive information. The zero-order correlations indicate that cognitive load was strongly related with expertise and coordinative complexity. Interestingly, supportive information was positively related with cognitive load, although this zero-order correlation was not significant.

Table II-7: Descriptive Statistics and Zero-Order Correlations

	Mean	*SD*	*(1)*	*(2)*	*(3)*	*(4)*	*(5)*	*(6)*	*(7)*
(1) Cognitive load	2.61	1.50	1						
(2) Component Complexity	1.88	.79	.30*	1					
(3) Coordinative Complexity	1.75	.88	.49**	.61**	1				
(4) Expertise	1.78	.57	-.61**	-.21	-.19	1			
(5) Supportive Information	8.2	11.2	.14	.03	-.06	-.38**	1		
(6) Task Type: At Least Partial Simplification	.80	.40	.-.19	.04	-.09	-.14	.24	1	
(7) Task Type: Full Simplification	.13	.33	-.30*	.20	.05	.06	-.15	.19	1

*(*significant at .05, **significant at .01, n = 56, unit of analysis: learning task)*

The results of the ordinal regression analyses are shown in Table II-8. Model 1 contains only dummy variables to control for any case-specific influence that remains constant during transition such as cognitive ability or tendencies to over- or underreport cognitive load. The regression coefficients of the dummy variables indicate that cognitive load was overall lower in the low-specificity cases 4 and 5 than in the high-specificity cases 1 to 3. Model 2 includes the predictors of CLT. Adding these predictors increases the variance explained (Cox & Snell) from .323 to .706. This suggests that CLT is able to predict a considerable fraction of the variance of cognitive load and that taking the learning task as level of analysis helps explain cognitive load. It is noteworthy that the systematic pattern of lower dummy variables for the low-specificity cases 4 and 5 disappeared in model 2. A possible explanation for this is that the predictors of CLT mediate the influence of specificity.

Table II-8: Results of Ordinal Regression

		Model 1: Case-level Controls Only		*Model 2: Full Model*		
	Variables	*Estimate (Std. Error)*	*Wald*	*Estimate (Std. Error)*	*Wald*	*Support for Hypotheses*
Thresholds	CL = 1	6.85 (2.08)	10.82**	3.46 (3.07)	1.27	-
	CL = 2	7.88 (2.15)	13.42***	5.42 (3.17)	2.93†	-
	CL = 3	8.95 (2.22)	16.34***	7.85 (3.25)	5.81*	-
	CL = 4	9.67 (2.25)	18.46***	9.11 (3.27)	7.74**	-
Independent Variables	Expertise	-	-	-2.81 (.679)	17.1***	H1 supported
	Component Complexity	-	-	-.09 (.40)	.06	H2a not supported
	Coordinative Complexity	-	-	1.50 (.44)	11.5***	H2b supported
	Task Type: At Least Partial Simplification	-	-	-1.62 (.83)	3.86 *	H3a supported
	Task Type: Full Simplification	-	-	-3.80 (1.28)	8.79**	H3b supported
	Supportive Information	-	-	-.51 (.33)	2.36	H4 not supported
Controls	Case 2	-.65 (.75)	.74	-1.84 (.93)	3.94*	-
	Case 3	-1.44 (.63)	5.17*	.50 (.801)	.39	-
	Case 4	-3.08 (.97)	10.03**	1.83 (1.69)	1.17	-
	Case 5	-3.84 (1.24)	9.63**	-2.40 (1.37)	3.07†	-
Variance Explained	Cox & Snell		.323		.706	-

*(† p = .10, * p = .05, ** p = .01, *** p = .001, n = 56, logit function, unit of analysis: learning task, all non-dichotomous variables have been standardized)*

The regression results help test our hypotheses. Expertise was very strongly related to cognitive load, supporting H1. Component complexity was not significantly related to cognitive load, giving thus no support for H2a (p = .81). Conversely, coordinative complexity was substantially positively related to cognitive load. This supports H2b. The coefficients of the dichotomous variables that coded task types were as expected. At least partially simplified task types were associated with lower cognitive loads that non-simplified task types (supporting H3a). Fully simplified task types were associated with lower cognitive loads than partially simplified task types (supporting H3b). Supportive information was only marginally related with lower cognitive load (p = .12). H4 can thus not be supported based on the data.

We further ran a fixed-effects panel model to examine whether the results of the ordinal regressions are biased by within-case interdependencies. The results are displayed in Appendix II-2 and are consistent with the results from the ordinal regression, indicating that the bias through within-case interdependence was tolerable in the data.

Both regression models produce consistent evidence of the strengths of relationships. Among the non-dichotomous variables, expertise had the strongest relationship with cognitive load. Coordinative complexity had a somewhat weaker, but still highly significant relationship with cognitive load in our data. The relationship between supportive information and cognitive load was weak and below statistical significance. Partially simplified task types were associated with a slight decrease of cognitive load, whereas the additional decrease of cognitive load through fully simplified task types (i.e. worked examples) was stronger.

We also tested for interactions between the predictors and cognitive load. No interaction effect was significant in any model even when the non-significant predictors were omitted. Although the low sample size may have limited our ability to detect interactions, the lack of significant interactions may suggest that there are rather strong independent associations between the predictors and cognitive load. The effects may thus be at least partially additive.

Overall, the results from the statistical analysis are largely consistent with CLT. Expertise, coordinative complexity, and simplified task types had strong associations with cognitive load. Component complexity did not seem to be related to cognitive load. Although we expected that both task complexity dimensions would be related with cognitive load, the discrepancy of our results for component complexity and coordinative complexity may somewhat align with CLT. The CLT literature uses the construct of element interactivity (the degree to which the information elements involved in a task interact) when theorizing on task complexity (Sweller and Chandler 1994). Element interactivity may be largely equivalent to coordinative complexity given the focus on interdependencies between information elements. The results thus indicate that the relationships between information elements (coordinative complexity) may be of far greater importance than the amount of information elements and acts (component complexity). The associations with supportive information were below statistical significance, but in the expected direction. The qualitative data suggests some explanations for the weak relationships with supportive information. These are reported next.

4.2 A Qualitative Perspective on Supportive Information

The non-significant relationships between supportive information and cognitive load may appear surprising given the substantial efforts that are often made to communicate knowledge to vendor engineers by face-to-face presentations, informal help, and codified knowledge directories (Chua and Pan 2008; Dibbern et al. 2008; Oshri et al. 2008). Two explanations can be proposed for the weak relationships. First, supportive information may not be fully exogenous. While supportive information had a positive zero-order correlation with cognitive load, the relationship turned negative (but remained non-significant) when controlling for the other CLT predictors. The positive zero-order correlation between expertise and supportive information may reflect one reason for this. Thus, the lower expertise, the more supportive information was consulted, leading to a pattern in which more supportive information was consulted when cognitive load was high due to low expertise. If we assume that our expertise measure was not free of measurement error, the regression model may not be able to fully control for the share of variance of cognitive load that should be attributed to variation of expertise. Our qualitative data suggests two mechanisms behind this correlational evidence. On the one hand, supportive information was triggered in planned manner when it could be anticipated that low expertise could yield high cognitive load. This is reflected in vast amounts of planned formal sessions and document study that took place at the beginning of the cases, when expertise levels were relatively low. Besides this planned, proactive use, supportive information was also triggered in a reactive manner in response to high cognitive load as exemplified in the following statements:

> *"Whenever I have doubts, I go and talk with [the SME] and try to get the details from them because they are the ones who have been here for a long, long time. So they know the system. Whenever I have doubts, I go and talk with them." (Vendor engineer, case 1)*

> *"If I don't understand it from the code, I contact [the SME]." (Vendor engineer, case 2)*

> *"If it is a hands-on – like he is implementing the change request and getting some problems – then I sat with him. (…) I then gave the theory he's trying to implement." (SME 1, case 3)*

In sum, possible endogeneity of supportive information made it less likely that a relationship between supportive information and cognitive could be detected in the model. To a lesser extent, this argument also applies to the use of simplified task types. We observed both the proactive planned use of simplified task type (e.g. by providing vendor engineers with the design of a maintenance request from the beginning of the work on this task) and a reactive increase in some episodes of initially high cognitive load.

Besides this methodological aspect, there is also a substantive explanation for the weak relationships of supportive information. Our qualitative data suggest that supportive information may have been differentially effective. On the one hand, there are episodes in our data during which additional supportive information seems to have relieved cognitive load, such as:

> *"I was not able to understand the language in the specification. But it was very clear when I saw the diagram. It just tells me the flow of data. (…) It was very clear then. (…) When I go to the diagram it is a lot easier, it is much faster." (Vendor engineer, case 1)*

> *"Whatever queries I had in implementing the change requests, I used to ask [the subject-matter expert]. (…) In the beginning, much input was required. Once I felt comfortable with it... well, it was during reading the design that I asked many doubts, so how the design was to be read." (Vendor engineer, case 3)*

On the other hand, some attempts to consult supportive information did not seem to relieve cognitive load. This was particularly salient for document study at the beginning of the high-specificity cases:

> *"I started reading one document. I didn't get any context of this. Then I asked [the SME]: 'What is this?' Then he said: 'Ok, you do not understand this now, you need to read first these three documents.'" (Vendor engineer, case 3)*

> *"How was this document helpful for you?" (First author) – "(Laughing) Ok, I went through the document and first I did not understand anything. Most of the things I could not understand. Whichever is easy to understand, I did not do that. I did check with [the SME] on some parts. Other parts, when I worked on it, I realized that this is how it fits into the picture." (Vendor engineer, case 1)*

CLT offers explanations for the differential effectiveness of supportive information. The CLT literature emphasizes that supportive information serves to provide blue-prints for schemas and that the information should be embedded in the learning tasks or task environment (Mayer and Moreno 2003; Van Merriënboer et al. 2003). A diagram that illustrates the data flow such as in the first interview quote may be a more effective blue-print for a mental model of the data flow than a written text. Moreover, when the subject-matter expert provided help while the vendor engineer struggled with the task, the information provided may have been more embedded into the context of the task than the information in a document that had not been purposefully created for this particular learning task. As Van Merriënboer et al. (2003, p. 6) put it: "When novice learners encounter problems while working on a learning task, the last thing they are inclined to do is further increase their already high-cognitive load by processing and mentally integrating additional information from [manuals or job aids in] a support system.". These observations suggest that different types of supportive information may be differentially effective in different situations and that subsuming these types of supportive information under one construct may therefore result in rather weak statistical relationships.

Taken together, the quantitative and qualitative results do not reject that supportive information reduces cognitive load. They rather draw a more multi-faceted picture of differential effectiveness of supportive information that may depend on the type of supportive information and further contingency factors. This invites future research.

4.3 Alternative Expertise Measures

Given the pivotal role of expertise for cognitive load implied in the results, we compared the models using alternative expertise measures. This not only served as a robustness check, but also illuminated whether our theoretical conceptualization of expertise as ensemble of highly domain-specific schemas was appropriate, which may help better understand the role of knowledge specificity. The procedure and the results are displayed in Appendix II-3.

The comparison of expertise measures suggested that alternative expertise measures that made identical theoretical assumptions, but were operationalized slightly differently yielded highly similar results to those reported in the previous chapter. This corroborates the robustness of the analysis process. Conversely, the expertise measures that made different theoretical assumptions about the nature of expertise were able to explain less variance. This strengthened the validity of our conceptualization of expertise as ensemble of domain-specific schemas and increased our confidence in using the expertise measure to explore how knowledge specificity impacts the predictions of CLT. This is reported next.

4.4 The Role of Knowledge Specificity

Although the predictors of cognitive load theory were able to well explain cognitive load in our data, it is not yet clear how knowledge specificity is related to these predictions. However, understanding the link between knowledge specificity and the predictions of CLT is essential to establish a link between CLT and the existing body of IS outsourcing literature, and to satisfy the interest of practitioners who may want to tailor CLT-consistent knowledge transfer approaches to context characteristics such as knowledge specificity.

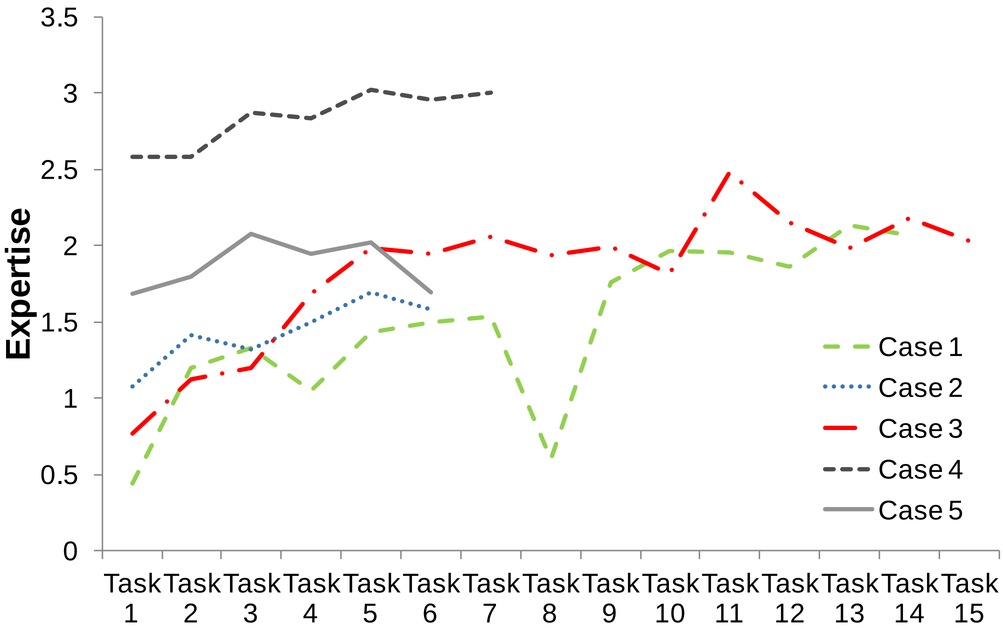

Figure II-3: The Evolution of Expertise in the Five Cases

Our analysis suggests that specificity impacts the start values of expertise because it constrains the degree to which prior related experience may translate into expertise. Figure II-3 may help illustrate this proposition. The figure shows the evolution of the expertise measure exp1 over learning tasks for the five cases. The very strong statistical relationship between expertise and cognitive load, the analysis of alternative expertise measures, and the consistency of our conceptualization of expertise with CLT research corroborate the validity of the measure. Although the expertise values fluctuated stochastically in function of the knowledge domains required for the learning task, the diagram depicts systematic variation. As the figure suggests, the cases 1 to 3 started at low expertise values, whereas initial expertise was higher in the case 5 and, in particular, in case 4. This discrepancy is noteworthy because the engineers in the cases 1 to 3 had substantial prior experiences in software projects of four (case 1), and eleven years (cases 2, 3) respectively (see also Table II-3). Moreover, their prior experiences were related to data warehousing, the domain of the role that they took over. The vendor engineers in the cases 4 and 5 had similar overall experiences in prior software projects. Moreover, they did not have more prior experience in the focus areas of the projects than the engineers in the cases 1, 2, and 3 had. Hence, differences in the amounts of prior experience cannot explain the difference in expertise.

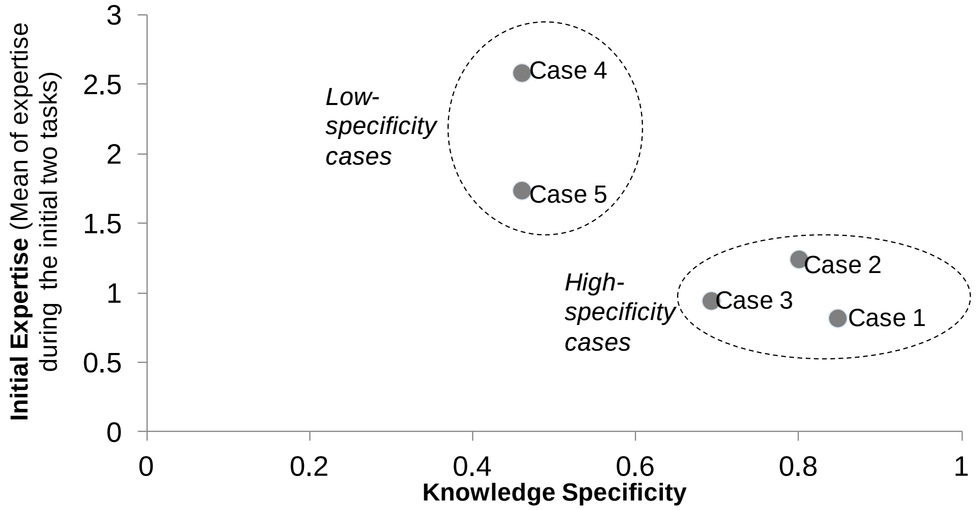

Figure II-4: Specificity and Initial Expertise

Conversely, we can explain considerable variation in initial expertise if we account for knowledge specificity. Figure II-4 illustrates this relationship by diagramming initial expertise against knowledge specificity. Knowledge specificity was operationalized as the amount of knowledge domains that were specific to the client over the total amount of knowledge domains in each case. For instance, 45 out of the total 65 knowledge domains in case 3 were coded to be client-specific, resulting in a knowledge specificity value of 45/65 = .69. Initial expertise was calculated as the mean of the expertise values in the initial two tasks to mitigate random fluctuations. The diagram is indicative of a strong relationship between knowledge specificity and initial expertise. The cases 4 and 5, which were of rather low specificity, were associated with higher initial expertise values than the high-specificity cases 1, 2, and 3. This association is a consequence of our theoretical conceptualization of expertise as ensemble of domain-specific schemas. Whereas vendor engineers can have prior experience in domains that are not specific to the client, they cannot have prior experience in domains specific to the client unless they worked for the same client before. Hence, specificity constrains the possible initial expertise values. For instance, when 85% of the knowledge domains in a role are specific to the client such in case 1, the engineer can draw on prior experience in a maximum of 15% of the domains that are relevant for a role. Put differently, our conceptualization of expertise implies that initial expertise values are largely determined by the interaction of knowledge specificity and the amount of prior experience.

Qualitative evidence underscores the central role of knowledge specificity for expertise. In the high-specificity cases 1, 2, and 3, interview statements indicating initially low expertise abound, in particular with regard to client-specific application knowledge and application domain knowledge. This was despite the substantial prior experience in data-warehousing projects and despite the substantial track record of the vendor organization in offshoring software work to India. The following statements illustrate situations in which the vendor engineers were engaged in tasks that involved client-specific knowledge:

> *"[When I read the requirement document,] at first it was like for a layman." (Vendor engineer, case 1)*
>
> *"At first, you do not have any knowledge. It is like a layman." (Vendor engineer, case 1)*
>
> *"It is a totally new thing on which I had not worked before. I had to get the details." (Vendor engineer, case 1)*
>
> *"But this task is very low-level. It is about a particular table and how it gets loaded." (Vendor engineer, case 1) – "If you had plenty of time and nobody was available to help you, could you solve the task by looking up information in the source code?" (First author) – "That would take a lot more time because coding is very complex. Doing that alone is very tough – almost impossible I would say without any help." (Vendor engineer, case 1)*
>
> *"I don't know that this is the business logic of the report or this is how the data mart is set up. I currently don't have the whole knowledge." (Vendor engineer, case 2)*
>
> *"I was not aware of the functionality." (Vendor engineer, case 2)*
>
> *"In terms of application knowledge, you get it here." (SME 1, case 2)*
>
> *"At that point, [the vendor engineer] was overloaded because [the vendor engineer] did not know the functionality of the application." (SME 2, case 2)*

Interestingly, the client employees, who had been involved in prior knowledge transfers, were not astonished or disappointed by the low expertise values at the beginning of transitions:

"He has been here for 2.5 or 3 months or so. You cannot expect that he is able to do anything at that point. Given that he has only been here for such short time, he has done a great job so far." (SME 2, case 1)

The self-perceptions of the engineers in the high-specificity cases 1, 2, and 3 as novices in many domains of their tasks contrast sharply with the perceptions of the engineers in the low-specificity cases 4 and 5. They had prior experience in the software package that they were to maintain. Asked how application knowledge was transferred to him at the beginning of the transition, the vendor engineer in case 4 seemed to perceive the question as somewhat absurd:

"I also had knowledge on those things. In IT, if you're not having knowledge, I don't think you're able to be trained on tools. This tool knowledge you should come up with. (…) We're not an IT tool training company." (Vendor engineer, case 4)

"I was already knowledgeable about the product and everything. Even before I got the knowledge transfer, I started working. Maybe for a fresher, it might be a different story what you are asking. Fresher in the sense of fresher for the project." (Vendor engineer, case 4)

His statements are indicative of high expertise due to prior experience in maintaining the same software package ("the tool"). Likewise, the vendor engineer in case 5 seemed to hold a naïve theory surprisingly similar to CLT according to which schemas from prior experience in the same software package lowered his cognitive load:

"My mind could easily map what the difference [to other implementations of the same software package] is. (…) If I have been through something, it always stays in the memory. My subconscious always has some images which never get lost." (Vendor engineer, case 5)

His information-processing at the beginning of the transition appeared to be in contrast to the information-processing when he made his first experience with the software package before this project (not in scope of this study). Although he had substantial experience in information technology projects at that time, he lacked experience in the software package and seemed to perceive high cognitive load:

"At the time, they used to go through the screens and hardly 10% went inside me – because I didn't have any picture in my mind which I could map those words to." (Vendor engineer, case 5)

In sum, substantial evidences from the operationalization of expertise and from the qualitative data suggest that expertise during transition is largely explained by the interaction of knowledge specificity and prior experience. When knowledge specificity is high, even experienced software engineers will identify themselves as novices in many domains of the maintenance tasks, notably those specific to the project. Conversely, engineers in low-specificity projects may capitalize on their prior experience from knowledge domains that overlap with the domains of the new software task. Although expertise values are likely to increase during transition, Figure II-3 illustrates that differences in initial expertise between the cases may not greatly diminish over the transition phase. According to the expertise literature, this is because expertise increases with the amount of practice and practice takes time. Hence, the interaction of specificity and the amount of prior related experience not only affects the initial values of expertise, but also expertise values at later points in time during transition. This suggests:

P1a: *The higher the amount of prior related experience, the higher is expertise.*

P1b: *The relationship between the amount of related experience and expertise is moderated by knowledge specificity. It is strong under low knowledge specificity and weak under high knowledge specificity.*

Figure II-5 recapitulates the results of regression analyses and illustrates how knowledge specificity is related to the predictions of CLT based on our findings. Expertise was shown to be very strongly related to cognitive load (H1). Our exploratory analysis suggested that expertise will initially be severely constrained in very specific software applications (P1a, b). Such applications thus pose a challenge because cognitive load risks being high as a consequence of the low expertise. To avoid cognitive overload in such settings, substantial strategies to reduce cognitive load may be required. These include initially assigning tasks with lower coordinative complexity (H2b), providing task type simplification (H3a, b) e.g. through the use of worked examples or completion tasks, and, possibly, supportive information (H4). The need for cognitive load reduction strategies may thus be driven by knowledge specificity.

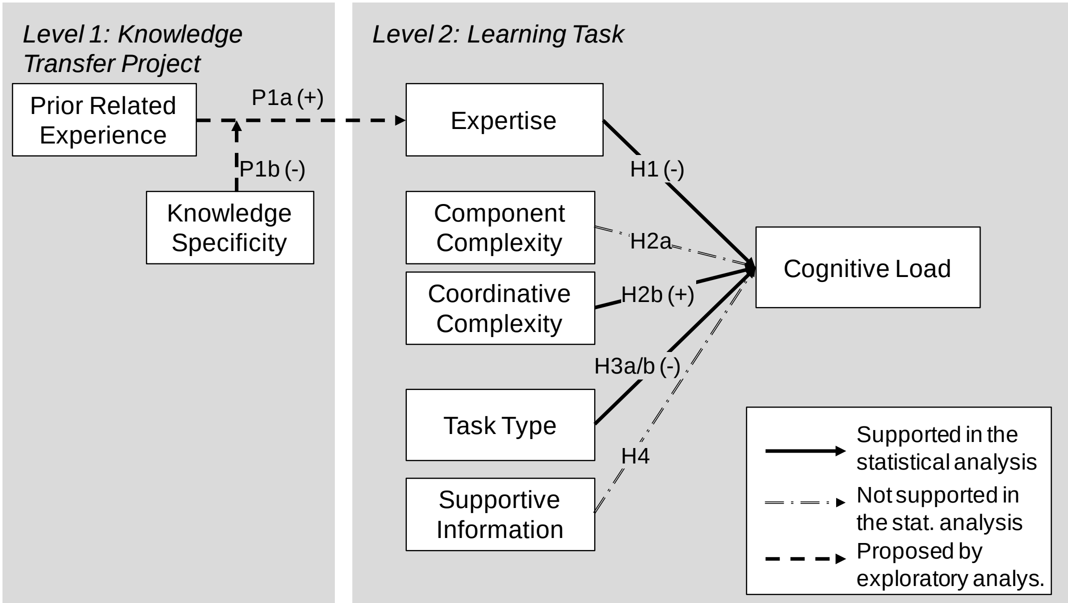

Figure II-5: Summary of Results

5 DISCUSSION

Although the existing literature highlights the central role of knowledge transfer for the success of SMOO projects, the literature provides scant guidance on how knowledge can be effectively transferred to vendor staff. In this study, we proposed that CLT may explain how effective knowledge transfer can be designed. CLT suggests that the engineers may learn effectively when their cognitive loads are neither too high nor to low, and that cognitive load may be a function of their expertise and of strategies to reduce cognitive load. Drawing on rich longitudinal data from five SMOO transitions, we made a first step to examine the predictions of CLT in the realm of SMOO by testing whether CLT can predict the cognitive load on vendor engineers. Moreover, we explored how knowledge specificity—a central construct in IS outsourcing research— is related to these predictions.

Our statistical analyses show that CLT can well predict cognitive load in SMOO. In particular, taking the learning task within a transition as the unit of analysis helped explain substantially more variance of cognitive load than could be explained by any individual-level or project-level characteristics, on which prior research has frequently focused (Kankanhalli et al. 2012; Ko et al. 2005; Williams 2011). In other words, projects with unfavorable characteristics such as high knowledge specificity are not foredoomed to fail (see our successful high-specificity cases 1, 2, and 3). Instead, the design of learning tasks within one transition may make a substantial difference.

It is thus worth understanding how the design of learning tasks impacts outcomes such as cognitive load and learning. Our study is a first step to this end because it examined the antecedents to cognitive load. In our data, expertise, coordinative complexity, and the use of simplified task types (such as worked examples or completion tasks) were strongly associated with cognitive load. Supportive information had weaker relationships with cognitive load, which did not reach statistical significance. Our qualitative analysis indicated that supportive information may have been differentially effective according to the degree to which it was embedded into the task environment and according to degree to which the presentation of the information qualified for schema blue-prints. We did not find any interactions between antecedents to cognitive load in our data. This may imply that the effects of the antecedents on cognitive load are largely additive, suggesting that—within the limits imposed by the strengths of the relationships— more of one load reduction strategy may compensate for less of another or for lower expertise. Taken together, low expertise may thus pose a challenge to projects because there is a significant risk of cognitive overload. However, learning tasks may be designed so that they compensate for the high cognitive load caused by low expertise. The strategies to this end include the assignment of learning tasks with lower coordinative complexity, the use of simplified task types, and, possibly and to a lesser extent, supportive information. If one accepts the contention of CLT that learning is most effective when cognitive load is neither too high or too low (Schnotz and Kürschner 2007; Sweller et al. 1998; Van Merriënboer and Sweller 2005), then our findings have implications for enabling effective learning. When the expertise of the vendor engineer is low, there will be a substantial need to compensate for low expertise by load reduction strategies to avoid high cognitive load. Conversely, when the vendor engineer's expertise is high, such load reduction strategies will not be needed or even be harmful (Kalyuga et al. 2003).

We also explored how knowledge specificity is related to the predictions of CLT. We found that knowledge specificity constrained the initial expertise values and, because expertise may not develop instantaneously, also subsequent values. While vendor engineers may draw on valuable prior related experience in low-specificity projects, they may start at rather low expertise values in high-specificity projects even when they bring substantial prior related experience. This is because they will be novices in the domains that are specific to the client unless they have experience with the same client. These domains will be salient to greater extent in high-specificity than in low-specificity projects. Cognitive load reduction strategies may thus be needed to be aligned with knowledge specificity. High-specificity projects are associated with low expertise and will thus need extensive load reduction strategies to compensate for low expertise and achieve moderate cognitive load. Conversely, low-specificity projects may recruit experienced vendor staff and then begin with higher expertise levels. The engineers may then face moderate cognitive loads even if no load reduction strategies are used. Extensive load reduction strategies may not pay off or be even harmful in such contexts.

5.1 Corroboration with Prior Research

Our findings are corroborated by prior studies on IS offshoring and software maintenance. Table II-9 shows their findings and how the CLT framework offers new interpretations of the findings. The consistency of the findings with the theoretical framework may strengthen its validity. The table also suggests that the theory may be worth being tested outside the boundary conditions of this study, which were SMOO transitions.

5.2 Theoretical Contribution

Our study makes primarily a contribution to the IS outsourcing literature, but has also implications for software maintenance, knowledge management, and cognitive load research. We contribute to the IS outsourcing literature by proposing a theoretically grounded framework of how knowledge may be effectively transferred to vendor teams in SMOO. In addition, we tested a part of this framework (the antecedents to cognitive load) and expanded it to include knowledge specificity, a central construct of the outsourcing literature. The study thus fills a knowledge gap on the design of effective knowledge transfer in SMOO contexts. SMOO projects may have a particularly strong fit with the boundary conditions of CLT given abundant accounts of overloaded offshore engineers. Some caution may thus be advised in generalizing from our findings in SMOO projects to more general software-maintenance environments or to software development outsourcing. However, our study may have implications for other domains in that it suggests a theoretical framework that may be worth being tested in these domains.

Table II-9: Corroboration with Prior Studies

Study	*Finding*	*Explanation by CLT*
Chua and Pan (2008)	Although the freshly recruited offshore engineers were provided substantial information in formal presentations, information interpretation was not feasible before an “on-the-job learning-by-doing period” (Chua and Pan 2008, p. 278).	The freshly recruited engineers lacked prior related experience in the bank’s software applications, which translated into low expertise. Even if supportive information is provided by formal presentations, they may not compensate for the low expertise given the weak relationship between supportive information and cognitive load. Only after an increase in expertise through practice were the engineers not cognitively overloaded any more.
	The bank had to abandon plans to hand over analysis and design tasks to the offshore team.	The expertise levels of the offshore team allowed them to solve completion tasks (i.e. tasks in which the design of the software modification is given), but not conventional tasks. Thus, only under partial task type simplification was the cognitive load on the offshore team within manageable amounts.
Dibbern et al. (2008)	Projects with high specificity and low absorptive capacity (operationalized as the amount of prior experience) faced significant extra costs for specification and design and for knowledge transfer	High knowledge specificity and low prior experience yielded low expertise values and thus made cognitive load reduction strategies necessary. Extra costs for specification and design imply that completion tasks (i.e. partial task type simplification) were reactively applied as a strategy to reduce cognitive load. Extra costs for knowledge transfer imply that more supportive information, another potential load reduction strategy, had to be provided than planned.
Wende and Philip (2011)	The reengineering of a custom-developed software system was delegated to an offshore vendor in a distributed setting. After nine months of disappointing results, the project was cancelled.	High knowledge specificity (caused by a custom-developed software system) yielded low initial expertise and thus high requirements for cognitive load reduction strategies. The distributed setting may not have been favorable to allow substantial load reduction strategies. Enduringly high cognitive load may have resulted in weak learning and thus stably low task performance over a period of nine months.
Boh et al. (2007)	Experience in maintaining the same software predicts individual task performance more strongly that experience in a related software system, which again is more strongly related to task performance than experience in unrelated systems.	The more strongly the software application is related to the software on which the engineers have prior experience, the more will the knowledge domains overlap (similar to the effects of low specificity in outsourcing projects). This results in more prior experience within the domains relevant to the tasks, in higher expertise, lower cognitive load and thus higher task performance.
Singh et al. (2011)	The learning effectiveness of different types of activities in open source programming communities depends on the programmer’s expertise (“learning state” in their terms).	Different types of activities may have different impact on cognitive load. These impacts may not satisfy, satisfy, or exceed the requirements for cognitive load regulation imposed by expertise and thus differ in learning effectiveness.

The study may thus also make a contribution to software maintenance research. For decades, the software-maintenance literature has aimed at explaining the cognition of software maintainers (e.g. Darcy et al. 2005; Pennington 1987; Von Mayrhauser and Vans 1995). While significant progress has been made, some observations have remained difficult to explain using existing theory (Espinosa et al. 2007). Our study indicates that the software-maintenance literature may potentially benefit from advances in education psychology by applying CLT. While the notions of expertise and task complexity are not new to software-maintenance research, task types have been rarely looked at. At this point, our study contributes by suggesting how tasks in software-maintenance environments may be mapped to the concept of task types. This may be used in future studies to test the predictions of CLT outside offshore outsourcing environments. Our results may also have implications for the literatures on knowledge management. As Alavi and Leidner (2001) summarized the state of research when referring to elements of knowledge transfer: "The least controllable element is the fifth: knowledge must go through a recreation process in the mind of the receiver." (Alavi and Leidner 2001, p. 120) Our study suggests that research on knowledge transfer can benefit from opening the black box of the "receiver's" cognition. This may help design knowledge management systems that are consistent with evidence on how people learn. CLT may serve as a valuable lens to this end. Our study may also point out that a too narrow focus on supportive information or explicit knowledge may frequently not help predict skill acquisition. Instead, seeing knowledge transfer as a learning process composed of learning opportunities such as learning tasks may also be fruitful perspective in settings outside SMOO. The study also contributes to the CLT literature by providing evidence of the boundary conditions of the theory. The results suggest that CLT may apply to professional learning contexts beyond highly structured training programs. The outcomes of informal learning processes may be well predicted by CLT. Given that most of CLT research has adopted an experimental paradigm (Van Merriënboer and Sweller 2005), our study thus strengthens the external validity of the theory in the field setting of this study.

5.3 Limitations and Future Work

Our study has several limitations, which may open avenues for future research. The data are from a single client organization. Moreover, while there was some random variation in our independent variables at the level of learning tasks, they were not fully free to vary because they were embedded in the cases. Both aspects may limit the generalization of our results although much empirical work may be subject to similar limits to generalization

(Lee and Baskerville 2003). A further limitation lies in the semi-structured nature of our data. While previous research in CLT has mostly used questionnaire items to measure cognitive load, our quantitizing strategy may involve some unknown measurement error. Yet, we are confident that this error does not change the nature of our results given the triangulation of our findings by using different sources of evidence and data analysis strategies and given the consistency of our theoretical framework with previous empirical findings. Moreover, we applied quantitative analysis strategies on a rather small sample size at the embedded level of analysis. This may have limited the inference on the results on supportive information and interactions. Yet, we believe that the rich nature of our qualitative data and the collection of longitudinal data during transitions may be positive downsides of our methodological approach and that they have been valuable for the purpose of the study. Furthermore, the research focused on on-site coordinators who were closely embedded in previous delivery teams during transition. Such settings are not uncommon in transitions (Dibbern et al. 2008; Gregory et al. 2009). However, future research may apply our theory to vendor personnel who are placed offshore during transition. Such research may also help better understand how further context factors specific to offshore outsourcing are related to the prediction of CLT. While we saw some indications in our data that, for instance, cultural and semantic distances may affect the independent variables such as the amount of supportive information, this was not included into the scope of this paper. Finally, the presence of the first author could have influenced the participants. However, given that he was present at the research site only for a very minor fraction of the total transition time, this influence may be manageable.

Future studies may thus have opportunities to heal weaknesses and expand the work of this study. Future work may replicate our study in different environments and using more structured data collection techniques that may be inspired by this work. Such studies may also include the team members placed offshore to theorize how further context factors specific to offshore outsourcing may be related to the theoretical model. Future research may also help clarify issues within theoretical framework that remained unresolved. For instance, our qualitative data showed that supportive information may have been differentially effective, e.g. based on media choice. Future work may examine such propositions. We were also unable to detect interactions in our model. Studies with larger sample sizes may help improve our understanding on whether interaction effects exist and what their sizes are. While our study has focused on the links from cognitive load reduction strategies to cognitive load, future work may examine antecedents to cognitive load reduction strategies or outcomes of cognitive load. The amounts of load reduction strategies may be influenced by issues of relationship quality or organizational controls. Research that sheds light into these antecedents could help transition managers effectively manage transitions. The consequences of cognitive load may also merit attention. It is the central contention of CLT that too high or too low cognitive load impairs learning, which, however, was not tested in this study. More controlled research settings such as experiments may be useful methods to test this. Our findings may also bring new questions up. The weak relationships of supportive information suggest that making explicit knowledge available to vendor engineers may not be most central to the skill acquisition of vendor staff in early transitions when knowledge is at least moderately specific. Instead, task sequencing strategies and the use of task type may be important aspects that have not received much attention in prior research. These strategies are only accessible as long as the SME are available to the project. This may leave client management with the central decision how long SME and vendor engineers should coexist in transitions. Answering this question may involve a delicate trade-off. While long phases of coexistence may erode the business case of offshoring, short coexistence phases may interrupt load reduction strategies at stage at which they would be much needed. Future research could address such questions by examining the dynamics that such decisions may entail. Finally, future work may help connect CLT to issues of group cognition and team learning, which have also been found influential in offshoring projects (Oshri et al. 2008).

5.4 Implications for Practice

Businesses are expanding their offshore outsourcing strategies towards knowledge-intensive software services (Booth 2013) and may make efforts for developing methodologies for effectively transferring knowledge to offshore teams during transition. An informal search of transition methods published on vendor homepages shows a strong reliance on knowledge codification. Our study draws a pessimistic picture of the effectiveness of such approaches when knowledge specificity and coordinative complexity are at least moderate. Supportive information such as information available in knowledge portals is unlikely to substitute for expertise. Extra costs and failures to hand over tasks despite extensive communication of knowledge (Chua and Pan 2008; Dibbern et al. 2008) were outcomes of projects that may have relied too heavily on the benefits of explicit knowledge. Codifying knowledge to make supportive information available to vendor teams may be sufficient to manage the cognitive load for simple tasks or when knowledge specificity is low and vendor engineers are experienced. However, beyond these conditions, projects may be well advised to broaden their perspectives on knowledge transfer from communicating knowledge to facilitating effective learning by managing cognitive load.

Projects should confront vendor engineers with authentic learning tasks, but manage the cognitive load imposed by these tasks by three strategies: simple-to-complex sequencing of tasks based on their coordinative complexity; using simplified task types such as worked examples, completion tasks, or imitation tasks; and, possibly, providing supportive information. High-specificity projects will frequently benefit from the extensive use of these strategies over a substantial period of time. Vendor engineers may thus first by confronted with worked examples of maintenance tasks (e.g. by job-shadowing or by studying the solutions to old maintenance requests) in loosely coupled software components (i.e. tasks with lower coordinative complexity), before working on completion or imitation tasks, and, still later, on conventional tasks in these domains. Such scaffolding strategies may then be repeated in more and more complex domains of the software. Although long periods of scaffolding may not sound appealing to clients interested in reducing their costs, vendors could differentiate from competitors by openly communicating the need for substantial guided practice in such settings. Conversely, low-specificity projects may recruit experienced vendor engineers, make supportive information available, and refrain from a scaffolding strategy.

Although this paper does not consider how managerial control influences knowledge transfer, some implications for the governance of at least moderately specific SMOO projects can be drawn. Given the central role of expertise and the high cost that may be necessary for extensive guided practice during transition, contractual knowledge governance should focus on reducing turnover rather than on knowledge codification. Service level agreements on personnel turnover may be a strategy to this end. Clients or vendors may also want to include coaches trained in CLT-based instructional designs such as the Four-Component Instructional Design Model (Van Merriënboer et al. 2002) into the transition process.

APPENDIX II-1: CODING RULES FOR QUANTITIZING

This appendix gives a shortened overview of the coding rules used for quantitizing. Sample quotes are also included.

Simplified Task Types

(1) Determine the task type according to Table II-1 (Van Merriënboer et al. 2002; Van Merriënboer et al. 2003). Examples are given in Table II-10.

Table II-10: Examples of Task Types

Task Type	Example Data
Worked Example	*"I tried to understand it for one particular case like one change request. I went through it starting from the requirements understanding, I tried to understand how the requirement has been given to us. What was the design, the proposed solution, how it was implemented. Then I saw the design and went to the code…."*
Imitation Task	*"[The SME] gave me the requirement. He said he has implemented this solution in a different setting. Maybe we can use it the same way."*
Goal-free Task	*"I started with creating some documentation. … They said I can create a document for them so they can share the documents with the Pune team."*
Completion Task	*"The design was already ready for these change requests. I went through the requirement and the design documents. Then implementation, testing, and everything was left to me. "*
Conventional Task	The requirements document is short and does not give any hints on the design of the software enhancement. The interview data suggest that no solution hints have been given by the SME.

(2) Assign the learning task configuration to a category based on Table II-11.

Table II-11: Categories of Simplified Task Types

Simplified Task Type Category	*Task Type*
No task type simplification	Conventional Task
Partial task type simplification	Completion task, imitation task, goal-free task
Full task type simplification	Worked example

Component Complexity (Wood 1986)

The coding scheme is given in Table II-12.

Table II-12: Coding Scheme of Component Complexity

Code	*Coding Rule*	*Examples*
1: Low	The acts and information cues involved in the task are few or highly similar. Example include: • Modification requests that impact only one or two software components or at most distinct 10 objects • Modification requests in which only simple database components such field values, columns, tables, views, and grants were altered, but these components appear homogeneous for the task and no changes to mapping rules or code were required. • Modification requests that involved configuration changes in few distinct objects, but no changes to code.	• *„He now has to create two or three new tables. He does not need to know the content of any package for this.”* (involved only simple homogeneous database components) • *“Creating a new company. Creating balance pools. Creating matching rules. All these things are within the tool itself. ... In a single session you can pass on this knowledge to someone. This is how to set up a company.“* (basic configuration change)
2: Medium	A moderate number of distinct acts or information cues are involved in the task. Examples include • Modification requests in which some distinct elements (e.g. a control table) add to simple database-level changes or configuration changes	• *“I went through all the packages and saw the data flow and how it happens from [component 1] to [component 2] and from [component 2] to [component 3]. And I checked where all the changes needed to be made and I made the changes according to that.”* (three components involved) • The column “affected objects” in the section “impacted objects” of the design document lists 18 distinct objects.
3: High	A considerable number of distinct acts or information cues are involved in the task. Examples include • Modification requests that impacted at least six software components or many (more than 20) distinct objects	• The column “affected objects” in the section “impacted objects” of the design document lists 27 distinct objects.

Coordinative Complexity (Wood 1986)

The coding scheme is given in Table II-13.

Table II-13: Coding Scheme of Coordinative Complexity

Code	*Coding Rule*	*Examples*
1: Low	The acts and information cues involved in the task are mostly independent or sequential	• *"This task was like an island."; "Frankly speaking I can implement it now, even without detailed knowledge… I just put a body on that by creating a package for it."* (independent relationships emphasized)
2: Medium	Some acts and information cues are in an independent or sequential relationship, whereas others are interdependent.	• The design document indicates that two conditions for joining database tables (sources of interdependence) needed to be identified, but the further acts involved in the change were sequential database changes.
3: High	The acts and information cues involved in the task are mostly interdependent of each other	• The design document describes the following task. Previously one view served as a data source at one specific point in the data flow of the data warehouse. Now, two sources shall serve as data sources. The existing loadings, mappings, and transformation logics that applied to this view needed to be changed. The design document indicates that the impact on each of the existing transformations needed to be assessed.

Supportive Information

(1) Code the duration of the supportive-information activity per day (see Table II-14).

Table II-14: Duration of the Supportive-Information Activities per Day

Code	*Coding Rule*	*Examples*
Very short (numerical value: 2.5/60)	Up to 5 minutes, very brief durations	*"Now, [the informal help] is very less. Once in a day, 1-2 questions a day."*
Short (numerical value: 17.5/60)	More than 5 and up to 30 minutes, short durations	*"My project manager has appointed [SME]… She was helping me a lot for initial set-up activities, initial brief introduction about how the overall processes work, technically as also functionally. Every day a half an hour meetings were scheduled."*
Medium (numerical value: 1)	More than 30 and up to 90 minutes, moderately long durations	*„When I have a free hour, then I take them and we have a session. More than hour would be too much … if you talk about a specific area of the application. 30 or 45 minutes and then it is ok." (SME)* (Per default, the longer duration was taken, i.e. 45 minutes in this case)
Long (numerical value 1.5)	More than 90 minutes, long durations	According to field notes, the knowledge elicitation session was from 9 a.m. to 11 a.m.

(2) Code the number of days per week during which the supportive-information activity took place.

(3) Code the number of weeks during which the supportive-information activity took place.

(4) Calculate the numerical value as duration * days per week * number of weeks.

(5) Assign the supportive-information activity to learning task configurations. If the activity refers to several learning tasks, distribute the duration equally on learning tasks.

Expertise

Table II-15 shows a fictitious example to illustrate how expertise was coded. The coding involved the following steps:

(1) Identify knowledge domains per case (see column 2 in Table II-15). To this end, identify statements in interviews, observation nodes, and documents that indicate a need for knowledge or skill.

(2) Assign the knowledge domains to the five categories of the IS body of knowledge (BoK; Iivari et al. 2004) (see column 3 in Table II-15).

(3) Identify statements on prior experience in each of the knowledge domains and enter the number of years of experience for each domain. When there is no precise information on the years of experience and the statement suggests substantial prior experience, choose 5 years of experience (i.e. the amount of prior experience in key areas demanded by the client in all five cases) (see column 4 in Table II-15).

(4) Identify the knowledge domains relevant for each learning task based on interview statements, and documents. Fill the matrix of associations between knowledge domains and learning tasks (see columns 5 to 7 in Table II-15).

Table II-15: Fictitious Example of the Coding of Expertise

No.	*Knowledge Domain*	*BoK Category*	*Prior Experience (Years)*	*Task 1*	*Task 2*	*Task 3*
1	Customer segmentation in client organization	Application Domain	0	1		
2	Adjustments in client organization	Application Domain	0			1
3	Component A	Application	0		1	
4	Component B	Application	0			1
5	Component C	Application	0	1		1
6	Peer review process in client organization	IS Development	0	1	1	1
7	Environments in client organization	Organizational	0	1	1	1
8	PL-SQL	Technical	5	1	1	1
9	Java	Technical	5		1	

(5) Determine the weights of each BoK category by determining the number of codes in each category per case. Table II-16 shows the results in case 1.

Table II-16: Weights of BoK Categories in Case 1

	Application Domain Knowledge	*Application Knowledge*	*IS Development Process Knowledge*	*Organizational Knowledge*	*Technical Knowledge*	*Sum*
Number of Codes	20	294	28	30	16	388
Weight	20/388 = .05	.76	.07	.08	.04	1

(6) Determine how to transform years of prior experience in the non-specific domains into numbers of prior experience. This involved estimating how many task experiences vendor engineers make in a non-specific domain per year. This was estimated based on the average ratio of number of experiences by time in non-specific domains in the data. For instance, a vendor engineer may have worked on six tasks in a non-specific knowledge domain over a period of half a working year. This corresponds to a ratio of twelve tasks per year. The average ratio in all non-specific knowledge domains in our data was 10.9 experiences per year. Hence, 10.9 experiences were accredited for each year of prior experience.

Table II-17: Expertise Formulae

Expertise Measure	*Formula*
Exp1 (the measure chosen for the results reported in the article): Expertise as weighted average of expertises in the individual domains (using the natural logarithm)	$\sum_{\substack{for\ all \\ BoK\ cat.i}} (w_i * \frac{1}{d_i} \sum_{\substack{for\ all\ relevant^* \\ domains\ j\ wihtin\ i}} \ln(1+k_j))$
Expertise for Task 3 (select the relevant domains for the tasks, which are 2, 4, 5, 6, 7, and 8): Exp1 = .05 * 1/1 * ln (1+0) + .76 * 1/2 * (ln (1+0) + ln (1+1))+ .07 * 1/1 * ln (1+2) + .08 * 1/1 * ln (1+2) + .04 * 1/1 * ln (1+5*10.9+2)	
Exp2: Expertise as weighted average of expertises in the individual domains (using the common logarithm)	$\sum_{\substack{for\ all \\ BoK\ cat.i}} (w_i * \frac{1}{d_i} \sum_{\substack{for\ all\ relevant \\ domains\ j\ wihtin\ i}} \lg(1+k_j))$
Expertise for Task 3 (select the relevant domains for the tasks, which are 2, 4, 5, 6, 7, and 8): Exp2 = .05 * 1/1 * lg (1+0) + .76 * 1/2 * (lg (1+0) + lg (1+1))+ .07 * 1/1 * lg (1+2) + .08 * 1/1 * lg (1+2) + .04 * 1/1 * lg (1+5*10.9+2)	
Exp3: Expertise as non-weighted average of expertises in the individual domains (using the natural logarithm)	$\frac{1}{d} * \sum_{\substack{for\ all\ relevant \\ domains\ j}} \ln(1+k_j)$
Expertise for Task 3 (select the relevant domains for the tasks, which are 2, 4, 5, 6, 7, and 8): Exp3 = 1/6 * (ln (1+0) + ln (1+0) + ln (1+1)) + ln (1+2) + ln (1+2) + ln (1+5*10.9+2))	
Exp4: Expertise as the weighted sum of prior experiences	$\sum_{\substack{for\ all \\ BoK\ cat.i}} (w_i * \frac{1}{d_i} \sum_{\substack{for\ all\ relevant \\ domains\ j\ wihtin\ i}} k_{j)}$
Expertise for Task 3 (select the relevant domains for the tasks, which are 2, 4, 5, 6, 7, and 8): Exp4 = .05 * 1/1 * 0 + .76 * 1/2 * (0+1) + .07 * 1/1 * 2 + .08 * 1/1 * 2 + .04 * 1/1 (5*10.9+2)	
Exp5: Expertise as the logarithmic weighted sum of prior experiences	$\ln(1+ \sum_{\substack{for\ all \\ BoK\ cat.i}} w_i * \frac{1}{d_i} \sum_{\substack{for\ all\ relevant \\ domains\ j\ wihtin\ i}} k_j)$
Expertise for Task 3 (select the relevant domains for the tasks, which are 2, 4, 5, 6, 7, and 8): Exp5 = ln(1 + .05 * 1/1 * 0 + .76 * 1/2 * (0+1) + .07 * 1/1 * 2 + .08 * 1/1 * 2 + .04 * 1/1 (5*10.9+2))	
Exp6: The weighted average of the natural logarithms of the number of experiences in all domains that have been coded within one case (including thus the domains that are not relevant for a particular learning task)	$\sum_{\substack{for\ all \\ BoK\ cat.i}} (w_i * \frac{1}{d_i} \sum_{\substack{for\ all \\ domains\ j\ wihtin\ i}} \ln(1+k_j))$
Expertise for Task 3: Exp6 = .05 * 1/2 * (ln (1+1) + ln (1+0)) + .76 * 1/3 * (ln (1+1) + ln (1+0) + ln (1+1))+ .07 * 1/1 * ln (1+2) + .08 * 1/1 * ln (1+2) + .04 * 1/2 * (ln (1+5*10.9+2) + ln (1+5*10.9+1))	
w_i: Weight for a BoK domain based on the relative number of codes of the BoK category over all BoK categories within a case d_i: The number of knowledge domains within the BoK category i that are considered in the formula d: The number of knowledge domains that were considered relevant for a particular learning task k_j: The number of prior tasks the knowledge domain j (i.e. the number of experiences before the work on this particular task)	

** When the data did not indicate that any particular knowledge domain within a BoK category was relevant for a particular learning task, all knowledge domains within the BoK category were considered relevant for this learning task.*

(7) Calculate expertise using the formulae displayed in Table II-17. Different measures of expertise have been calculated to check the robustness of the procedure and to illuminate the theoretical assumptions (see Appendix II-3). The table includes the

calculation of expertise for task 3 in Table II-15 as an example (assuming the weights of case 1).

Cognitive Load

Cognitive load was measured based on the ratio of mental effort and task performance (Paas et al. 2003). Figure II-6 illustrates the coding categories. Cognitive load was considered highest (category 5) when the onsite coordinator was not able to make sense of information or solve a task despite high mental effort. Cognitive is load was somewhat medium (category 3) when the onsite coordinator was able to make sense of information or solve a task with high mental effort. Cognitive load was lowest (category 1) when the onsite coordinator was able to make sense of information or solve a task with low or virtually no mental effort. Hence, the measurement scale assumes that, when being confronted with a task, the onsite coordinator increases mental effort until she/he is able to make sense of the information/solve the task or until the limits of her/his working-memory capacity have been reached. When the demands imposed by the task exceed working-memory capacity, this additional cognitive load is reflected in the decrease of task performance (categories 4 and 5). This assumes that the combinations in the lower left quadrant of Figure II-6 do not occur. They would reflect an onsite coordinator who does not successfully complete a task or make sense of information and is not willing to increase mental effort. Because all onsite coordinators in our research cases have been characterized as highly motivated (see also other accounts of motivated, but overloaded vendor staff (Dibbern et al. 2008)), we assumed that the combinations in the lower left quadrant did not occur. Sample quotes are given in Table II-18.

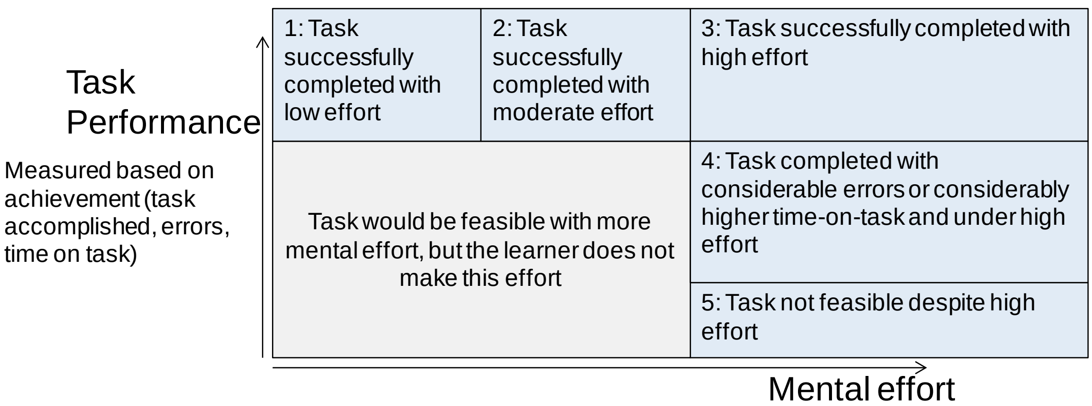

Figure II-6: Cognitive Load Categories

Table II-18: Sample Quotes of Cognitive Load

CL	Coding Rule	Sample Quotes
1	Successful task performance and statements highlight low perceived difficulty or the use of forward-working problem-solving strategies	• *"That was more of a robot."* • *"This was a manual thing, again not a brainy activity."*
2	Successful task performance and statements express medium perceived difficulty or there are no indications of high or low difficulty despite long accounts of the task.	• *"This defect? Ok. This is a reference table. … The reference data updates need to be done based on different documents. … I had a discussion with the requirements engineer directly because he is the one who has raised the defect. I got from him what exactly he wants and what he is looking for. He also gave me inputs on how to go ahead with that. Based on that and some discussions with [SME 1] and [SME 2], I took it forward"* (The statement describes how the vendor engineer worked on the task, but the engineer did not make any claims on high or low perceived difficulty)
3	Successful task performance and statements highlight high perceived difficulty or the use of backward-working problem-solving strategies	• *"The data is so messed up that we have to look into many things. You fix one thing, one thing gets disturbed and then you again have to see what's happening."*; in the subsequent interview: *"How did that evolve, were there any further actions?" (first author) – "Not really. Those were the production incidents which came and we closed them. Then we put them into production. That was the end of it. After that we have not got any related issues so far."* (The statements indicate successful task performance, but high mental effort due to backward-working problem-solving strategies.)
4	The engineer could complete the task but with considerable errors or high time-on-task and high effort	• *"Did any issues come up after the deployment to production?" (first author) – "There was only an issue with running the job for running one particular job. It was not getting kicked off…. What happened ultimately was there was some wrong condition given like some of the conditions were missing. … The instruction [that I had given to the other team in the deployment procedures] was not proper."*
5	The engineer could not solve the task or make sense of the worked example.	• *"He did not know how he could test that."* • *"[Before I got any help] it was a totally new thing. I had to get the details."* (Cognitive load was coded 5 for the learning task configuration that did not include the help.)

APPENDIX II-2: RESULTS OF THE FIXED-EFFECTS PANEL REGRESSION

A fixed-effects panel model was run to examine whether the results of the ordinal regressions are affected by within-case dependencies. The results are given in Table II-19.

Table II-19: Results of the Fixed-Effects Panel Model

	Fixed-Effects Parameter Estimate (Std. Error)	*t-value*
Constant	3.12 (.40)	7.65***
Expertise	-1.05 (.19)	-5.52 ***
Comp. Complexity	-.09 (.16)	-.55
Coord. Complexity	.63 (.16)	3.99 ***
Task Type: At Least Partial Simplification	-.54 (.33)	-1.65†
Task Type: Full Simplification	-1.24 (.39)	-3.20**
Supportive Information	-.14 (0.14)	-1.04

*(† p = .10, * p = .05, ** p = .01, *** p = .001, n = 56, random constant term, restricted maximum likelihood method, unit of analysis: learning task, all non-dichotomous variables have been standardized)*

APPENDIX II-3: COMPARISON OF ALTERNATIVE EXPERTISE MEASURES

We examined how the regression results changed when alternative expertise measures were used. This served both as robustness check, and as assessment whether our theoretical conceptualization of expertise as ensemble of highly domain-specific schemas was appropriate. Table II-17 displays the formulae of the alternative measures. Appendix II-1 illustrates how these measures are calculated in a fictitious example. Table II-20 shows how explained variance changes when these alternative measures are used while all further model variables are identical and correspond to the model reported in section 4.1. The subsequent analysis is based on changes of the Akaike Information Criterion (AIC) in the fixed-effects panel model. This model was chosen because it is least sensitive to nested data. A widely used rule is that one model makes weaker predictions than an alternative model when its AIC value exceeds the AIC value of the alternative model by more than 2 (Jansen 1993). Because expertise is seen as a causal dimension of cognitive load in CLT (Paas et al. 2003), higher explained variance of cognitive load when using a particular expertise measure may signal higher validity of the expertise measure. Indicators for explained variance in the ordinal logistic regression model are also reported for triangulation. Exp1 was the measure used for the results reported above.

Table II-20: Results from Alternative Expertise Measures

Ex. Msr.	*AIC*	*Cox & Snell*	*Distinct from Exp1*	*Implication*
Exp1	159.9	.706	-	-
Exp2	158.2	.706	No support	The results are robust to alternative choices of the base of the logarithm.
Exp3	161.7	.698	No support	Discretion in the coding of knowledge domain may not change the nature of the results.
Exp4	183.0	.596	Support	Expertise gains from experience decrease with the amount of experience.
Exp5	174.7	.614	Support	Assuming one overall learning curve rather than one learning curve in each knowledge domain yields lower predictive validity. Substantial experience in one domain may thus only partially compensate for lacking experience in another domain.
Exp6	163.7	.692	Support	A broad measure of expertise that encompasses experience in all domains of the role predicted cognitive load worse than a more narrow measure that includes only domains relevant for a particular learning task. This is consistent with the assumption of domain specificity of expertise.

(AIC = Akaike Information Criterion of the fixed-effects panel model; Cox & Snell = Cox & Snell values of the ordinal logistic regression model; all models included the same variables as the models reported in section II-4.1 except for different expertise measures.)

The measures exp2 and exp3 serve as robustness checks of the results. Exp2 distinguishes from exp1 by using the common logarithm instead of the natural logarithm. Comparing exp1 and exp2 thus helps assess whether the results are sensitive to changes of the base of the logarithm. The models yield similar AIC values of 159.9 and 158.2. This suggests that our results are not contingent on the choice of the basis of the logarithm. In exp3, the averages of the logarithms are not weighted per BoK category whereas exp1 uses weights that indicate the importance of each of the five BoK categories in each of the cases. The weights are calculated based on the relative number of codes of the BoK category over all BoK categories within a case. For instance, in case 3, 87 codes referred to application knowledge and totally 190 codes referred to any of the five BoK categories. The weight for application knowledge in case 3 was thus 87/190 = 0.46. Whereas the calculation results were aggregated and weighted at the level of BoK categories in exp1, no aggregation and weighting was used for exp3. Because exp1 includes weights based on the number of codes, it may be less susceptible to discretion the identification of knowledge domains. The panel model yields similar AIC values of 159.9 and 161.7 for both measures. The minor difference indicates discretion in weighting knowledge domains may not account for a large share of variance explained. More importantly, exp1, exp2, and exp3 produce not only similar explained variances, but also consistent significances of the independent model variables. The regression results are thus unlikely to be an artifact of the base of the logarithm or the weighting procedure.

Exp4 helps examine whether assuming a linear rather than a logarithmic relationship between the amount of experience and expertise yields higher predictive validity. Consistent with the learning curve literature (e.g. Kim et al. 2012), assuming a logarithmic relationship resulted in far better model fit than under a linear relationship. This is signaled by a substantially lower AIC of 159.9 compared to 183.0. Thus, gains from experience decreased with the amount of experience.

If decreasing gains from experience are assumed, the question arises whether the decreasing benefits manifest at the level of the overall experiences pooled together from all relevant knowledge domains or on the level of each knowledge domain. Consider the example of a vendor engineer who works on a software task that relates to three modules of a software system. The engineer may have substantial experience in the modules A and B of the software package, but no experience in a custom-built module C that enhances the software package. Does the experience made on module C in this task add to the overall rather high experience in the whole software system, which would imply a rather low increase in expertise, or does the experience add to the non-existent experience in module C, which signals a rather high increase of expertise because of the low start level of experience in module C? Put differently: Is there one expertise to which experiences from diverse subject areas of the maintenance role contribute equally; or are there expertise developments in each of the knowledge domains? Exp5 sheds empirical light on these questions. Whereas exp5 assumes one expertise for the software-maintenance role, exp1 assumes expertise development in each knowledge domain of the role and calculates the overall expertise based on the average of these expertise values. As Table II-20 indicates, exp1 made considerably better predictions (AIC of 159.9) in our data than exp5 (AIC of 174.7). In the example, it seems more appropriate to consider the engineer an expert in the modules A and B and a novice in module C than to consider him to be close to an expert status in the overall software system. The result is consistent with the conceptualization of expertise as the ensemble of domain-specific schemas in CLT, which implies that schemas need to be acquired and refined in each knowledge domain. This issue is of relevance in SMOO. Engineers may frequently have significant prior experience in some domains of a given software-maintenance role and no experience in other domains of the same role because they domains are specific to the company. Our results imply that these engineers are novices in the domains in which they have no prior experience.

Exp6 helps examine the assumption of domain specificity of expertise. Whereas exp1 includes only experiences in knowledge domains that are relevant for a particular learning task, exp6 includes prior experience in all knowledge domains of a software-maintenance role. For instance, a software defect may refer to the module A and B of a system, but be unrelated to the modules C and D. Should only the prior experience in the modules A and B be included for the expertise measure or should the measure account for the experience in all four modules? Our results show better predictions when only the relevant knowledge domains are considered (i.e. exp1; AIC of 159.9) than when all knowledge domains are considered (i.e. exp6; AIC of 163.7). This finding is consistent with the assumption of rather narrow domain specificity of expertise in CLT.

In sum, our analyses of alternative expertise measures show similar results for exp1, exp2, and exp3, which share the same theoretical assumptions. This corroborates the robustness of our analysis. Conversely, the expertise measures exp4, exp5, and exp6 make different assumptions of the nature of expertise than exp1, and they made weaker predictions. The results lend support for the conceptualization of expertise prevailing in CLT, according to which expertise reflects the ensemble of domain-specific schemas. Consistent with CLT, learning processes may thus manifest separately in each of these domains rather than on an overall aggregated level. Although these findings are based on comparisons of the AIC values in the fixed-effects panel model, the ordinal logistic regression models show the same pattern. Understanding the conceptual nature of expertise aided the analysis of the effects of knowledge specificity.

STUDY 2

MEDIA CHOICE AND COMMUNICATION PERFORMANCE DURING KNOWLEDGE TRANSFER IN SOFTWARE-MAINTENANCE OFFSHORING

ABSTRACT

Insufficient knowledge transfer to vendor teams is one of the major reasons for failure of software-maintenance offshore outsourcing projects. In this study, we investigate how media choices impact the performance of conveyance processes during knowledge transfer in the transition phase of these projects. Our theoretical lenses, the cognitive theory of multimedia learning and media synchronicity theory, make divergent predictions of how media choices impact communication performance. We examine the predictions using qualitative data from eight software-maintenance cases. Our results lend support for the prediction of the cognitive theory of multimedia learning according to which media that involve both visual and auditory channels are associated with higher communication performance, in particular when intrinsic cognitive load is high. Conversely, the predictions of media synchronicity theory find only weak support in our data. Implications for media choice in knowledge transfer to vendor teams and for the reference theories are discussed.

1 INTRODUCTION

Businesses continue to offshore outsource software services such as application maintenance to vendors in countries such as India (Oshri et al. 2011). Yet, many offshore outsourcing endeavors do not meet the initial expectations (Booth 2013; Dibbern et al. 2008). Ineffective knowledge transfer to vendor teams is one of the major reasons for failure (Dibbern et al. 2008; Westner and Strahringer 2010). Knowledge transfer is particularly salient in the transition phase at the outset of projects, during which the offshore team takes over the responsibility for delivery (Chua and Pan 2008; Tiwari 2009). In this context, knowledge transfer may be seen as the process through which the vendor team acquires the knowledge required to perform their tasks. This process is frequently problematic. Although vendor personnel may travel to the client site to closely interact with subject matter experts (SME), they may struggle to assimilate vast amounts of information about the client's software applications, business, and software-maintenance processes (Chua and Pan 2008). As a consequence, they may fail to take over the software tasks or require substantial direction (Chua and Pan 2008; Dibbern et al. 2008). Transition managers may therefore be highly interest to understand how to effectively transfer knowledge to vendor teams.

Recent research has contributed to our understanding of how vendor engineers acquire the task knowledge (i.e. the knowledge required to perform the software-maintenance tasks) in software-maintenance offshore outsourcing (SMOO) projects (Krancher and Dibbern 2012, see also study 1). Consistent with research from education psychology (Merrill 2002; Sweller 1994; Van Merriënboer et al. 2002; Van Merriënboer and Sweller 2005), vendor engineers were found to learn most effectively by engaging in authentic learning tasks that imposed neither too high nor too low cognitive load on them (Krancher and Dibbern 2012). Cognitive load denotes the cognitive demands that a given task puts on a given learner (Paas et al. 2003). Three strategies can be used to manage cognitive load: simple-to-complex sequencing of tasks, using simplified task types such as worked examples and completion tasks, and providing supportive information (information that helps learners to understand the non-recurrent aspects of the task domain) (Van Merriënboer et al. 2003). The use of these strategies in function of expertise may enable engineers to learn effectively despite low expertise in the task domain (Krancher and Dibbern 2012). However, whereas the use of intrinsically simpler tasks and of simpler task types was associated with considerably lower cognitive load, the association between supportive information and cognitive load was surprisingly weak (see also study 1). This weak relationship contrasts with the frequently substantial efforts to make supportive information available to the members of the offshore unit through face-to-face presentations, informal help, and codified knowledge directories (Chua and Pan 2008; Dibbern et al. 2008; Oshri et al. 2008). Do these projects waste resources for activities that do not meaningfully contribute to the engineers' learning?

The qualitative analysis of the data in study 1 suggested a more differentiated interpretation of the weak relationship of supportive information and cognitive load: Supportive information may have been differentially effective. Media choice emerged as one theme that may impact whether the supportive information can be effectively communicated to vendor engineers (see study 1). Communication processes during knowledge transfer in offshore outsourcing may be particularly sensitive to media choice because of the barriers that lie in the nature of offshore outsourcing (Wende and Philip 2011; Wende et al. 2010). These barriers include cultural, semantic, and geographical distances, and scarce prior experience in the task domain (Dibbern et al. 2008). The actors involved in transitions choose between an array of media such as documents, instant messaging, email, phone calls, video conferences, face-to-face meetings, screen-sharing conferences, and others for the purpose of conveying knowledge to the vendor team (Wende and Philip 2011; Wende et al. 2010). While it has been asserted that these choices influence communication performance (the extent to which recipients are able to build or revise a mental model from a message) in application offshoring transitions (Wende and Philip 2011; Wende et al. 2010), few initial studies have empirically investigated this claim.

Empirical evidence of the effectiveness of different media is particularly desirable because media theories from different fields make divergent predictions about media performance. Media synchronicity theory (MST) (Dennis et al. 2008) from computer-supported collaborative work research holds that the communication performance is contingent on media synchronicity—the capability of a medium to enable individuals to achieve a state in which individuals are working together at the same time with a common focus. MST predicts that the conveyance of information benefits from media that support low synchronicity such as documents or emails and suffers from media of high synchronicity such as face-to-face conversations. In contrast, the cognitive theory of multimedia learning (CTML) (Mayer and Moreno 2003) from educational psychology favors the use of media that allow the simultaneous use of auditory and visual channels to convey information, recommending thereby face-to-face conversations rather than text-based media. In this study, we examine the predictions made by the two theories using qualitative data on communication processes within eight case studies of SMOO transitions. We thereby address the following research question:

How does media choice influence communication performance in supportive-information activities during SMOO transitions?

This paper is structured as follows. We first present the predictions made by MST and CTML. Next, we describe the embedded-case study approach adopted for theory testing. Finally, we present the results and discuss implications for knowledge transfer in offshore outsourcing and for MST and CTML.

2 THEORY

In this study, we test the predictions about communication performance in conveyance processes made by MST and the CTML in the context of SMOO transitions. The two theories were selected for three reasons. First, both theories make claims on how media capabilities can affect communication performance in conveyance processes and thus help address our research question. Second, both theories base their predictions on cognitive rather than affective arguments. Research on software-maintenance offshoring transitions emphasizes how cognitive limitations constrain knowledge transfer (Chua and Pan 2008; Dibbern et al. 2008). The focus on cognitive arguments may thus be well aligned with the boundary conditions of this study. Third, both theories resulted from extensive theory development efforts in different disciplines. MST has been developed after empirical data were difficult to be explained by prior media theories such as media richness theory (Daft and Lengel 1986) popular in information systems research (see Dennis et al. 2008 for an overview). The CTML has been developed through two decades of research on the conveyance of complex information in multimedia learning research (Clark and Mayer 2011; Mayer and Moreno 2003). Hence, using these two theories may allow making use of the results of substantial theory development efforts in different fields. We next present the assumptions and theoretical arguments made by both theories. We then develop hypotheses from the two theories.

2.1 The Cognitive Theory of Multimedia Learning

The CTML aims at predicting how media use in multimedia learning environments impacts learning outcomes such as the acquisition of mental models of the communication content (Mayer and Moreno 2003). It holds that media choice positively impacts learning when it reduces the cognitive load on learners (the demands that material or tasks impose on the cognitive systems of learners) provided that a need for load reduction exists. The predictions of the CTML have been replicated in a series of controlled experiments and are corroborated by a related stream of research on cognitive load theory (CLT) that stresses the essential role of cognitive load management for learning outcomes (Sweller et al. 1998; Van Merriënboer et al. 2003; Van Merriënboer and Sweller 2005).

The CTML makes two assumptions that are central for media selection in the context of our study (Mayer and Moreno 2003). First, it assumes limited working memory capacity and virtually unlimited long-term memory (limited capacity assumption). The conscious processing of incoming information occurs in working memory. The more information elements need to be simultaneously processed in working memory, the higher is the cognitive load. Cognitive load may be relieved by schemas in long-term memory. Experts hold powerful schemas and are thereby able to aggregate information to higher-order and therefore less information elements. The limited capacity assumption suggests that vendor employees may be quickly overstrained by information that they cannot relate to schemas from prior experience such as the structure of a custom-built software system. This effect may be exacerbated when learners cannot control the timing of information presentation because it deprives the learners of the possibility to intellectually digest one chunk of information before processing the next, increasing thereby cognitive load (the segmentation effect in the CTML).

Second, the CTML assumes that the cognitive load imposed by messages may be distributed on an auditory channel and a visual channel, each of which has their own capacities (dual-channel assumption), and that the use of both channels helps offload the channel that else would be overloaded. Figure II-7 may help explain this information-processing framework. For instance, a software engineer may process the words and diagrams in a software document. Both types of information will enter the engineer's cognitive system through her eyes. They will then be processed in her visual channel, while no substantial information processing occurs in the auditory channel. When the material is complex or the engineer lacks prior related experience, her visual channel will be overloaded and she will struggle to build a mental model of the information. Shifting some of the load to the auditory channel can then help avoid overload and improve learning. Consider a face-to-face conversation in which the engineer listens to the explanations given by an SME while the

SME anchors his explanations in a diagram visible to the vendor engineer. In this case, some of the cognitive load is shifted to the auditory channel, reducing thereby the risk of overloading the visual channel. These assumptions suggest that media will ease the conveyance of information when they enable using both the auditory and the visual channel (the modality effect in the CTML). We refer to the capability of a medium or a set of media to simultaneously transfer both auditory and visual information as dual-channel capability. We next present the assumptions of MST before contrasting the predictions made by both theories in our theoretical model.

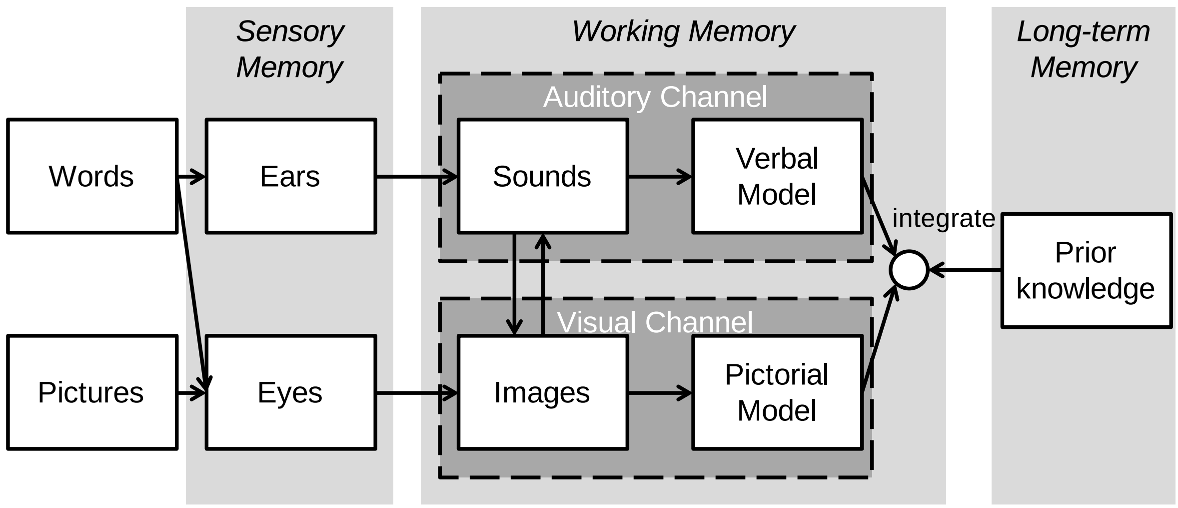

Figure II-7: The Cognitive Theory of Multimedia Learning (simplified from Mayer and Moreno 2003)

2.2 Media Synchronicity Theory

We next describe MST according to the most recent version of the theory (Dennis et al. 2008). MST aims at predicting the performance of communication processes in organizational tasks. It assumes that organizational tasks involve conveyance and convergence processes. Conveyance processes are "the transmission of a diversity of new information (…) to enable the receiver to *create* and *revise* a mental model of the situation" (Dennis et al. 2008, p. 580, emphasis in the original). Convergence processes are the "discussion of pre-processed information about each individual's *interpretation* of a situation, not the raw information itself" (Dennis et al. 2008, p. 580, emphasis in the original). MST holds that these communication processes require different degrees of media synchronicity, the capability of a medium to enable individuals to achieve a state in which individuals are working together at the same time with a common focus. MST posits that convergence processes benefit from high synchronicity, whereas conveyance processes benefit from low synchronicity.

In this study, we limit our attention to conveyance processes. In transitions, vendor employees frequently struggle to understand the new, diverse and large amounts of supportive information they are confronted with (Chua and Pan 2008; Dibbern et al. 2008; Krancher and Dibbern 2012). The participants in transitions therefore face the challenge of conveying supportive information to the vendor engineers, i.e. of enabling vendor engineers to build a mental model of the domain of their task. This does not imply that convergence processes are absent in SMOO transitions. We focus on the conveyance of supportive information because they seem an essential element in knowledge transfer during transitions. MST predicts that conveyance is more effective under low media synchronicity. This is because conveyance involves the transmission of new, diverse, and larger information and retrospective and deliberative processing of this information by the recipient. These processes benefit from low synchronicity because it will allow "more time for information processing to analyze the content of a message or to develop meaning across messages" (Dennis et al. 2008, p. 582) and because it enables the sender to craft messages more carefully to ease their digestion by the recipient.

Media synchronicity is proposed to mediate the impact that five media capabilities have on communication performance (Dennis et al. 2008): transmission velocity (the speed at which a medium can deliver a message to the recipients), parallelism (the number of simultaneous transmissions that can effectively take place), symbol sets (the number of ways in which a medium allows information to be encoded), rehearsability (the extent to which the media enables the sender to fine-tune a message before sending), and reprocessability (the extent to which the medium enables a message to be reexamined again during decoding, either within the context of the communication event or afterwards). Synchronicity is suggested to be highest under high transmission velocity, low parallelism, large symbol sets, low rehearsability, and low reprocessability. Because synchronicity is held to be detrimental for conveyance processes, media that meet these conditions should be avoided for the goal of conveyance.

Table II-21 summarizes central characteristics of the two theories.

Table II-21: Overview of MST and the CTML

	MST	*The CTML*
Publication	Dennis et al. (2008)	Mayer and Moreno (2003)
Domain of Origin	Computer-supported collaborative work (information systems)	Multimedia learning (educational psychology)
Motivation	Lack of empirical support for media richness theory and other theories	Failure of technology-driven multimedia learning projects (Clark and Mayer 2011)
Main Assumption	Convergence and conveyance processes can be distinguished and require different media capabilities.	Media needs to account for the architecture of human cognition: limited working memory capacity, two separate channels.
Communication Processes	Convergence, conveyance	Conveyance (although not explicitly stated)
Dependent Variable	Communication Performance	Meaningful learning (deep understanding of the material)

2.3 Theoretical Model

The constructs of MST and the CTML overlap in both independent and dependent variables. Both theories aim at predicting the effects of symbol sets such as the existence of both visual and auditory channels and of reprocessability. Likewise, the dependent variables are highly similar. The CTML targets at predicting deep understanding of the presented material, which "involves the construction of a mental model" (Mayer and Moreno 2003, p. 43). In the realm of conveyance, MST predicts when recipients are able to create or revise a mental model of the situation. Both theories are therefore concerned with media choices that enable a recipient to understand material by building a mental model of it.

In this study, we concentrate on the two media capabilities that are common to both theories: dual-channel capability (which is analogous to, but more coarse-grained than the construct of symbol sets in MST) and reprocessability because we expect these media capabilities to be more relevant in the realm of SMOO transitions than the other three. For instance, high amounts of parallelism such as parallel, simultaneous discussions between some vendor engineers and SME may be rather an exception during knowledge transfer. Likewise, although media with high rehearsability would allow SME to fine-tune messages before sending them, they may face difficulty in finding time to carefully craft learning material during transitions, in which they face high workloads by their responsibilities for both software delivery and knowledge transfer. Finally, the arguments made by MST about transmission velocity seem to focus rather on convergence than on conveyance processes, indicating that this capability might be less relevant for this research given our focus on conveyance. In contrast, the theoretical arguments of the CTML and MST suggest that dual-channel capability and reprocessability may be influential in software-maintenance offshoring transitions because the arguments are related to the cognitive load on the recipients and the cognitive load on recipients has been found to constrain communication in software-maintenance offshoring transitions (Chua and Pan 2008; Krancher and Dibbern 2012). Figure II-8 shows the resulting theoretical model. We next explain its hypotheses.

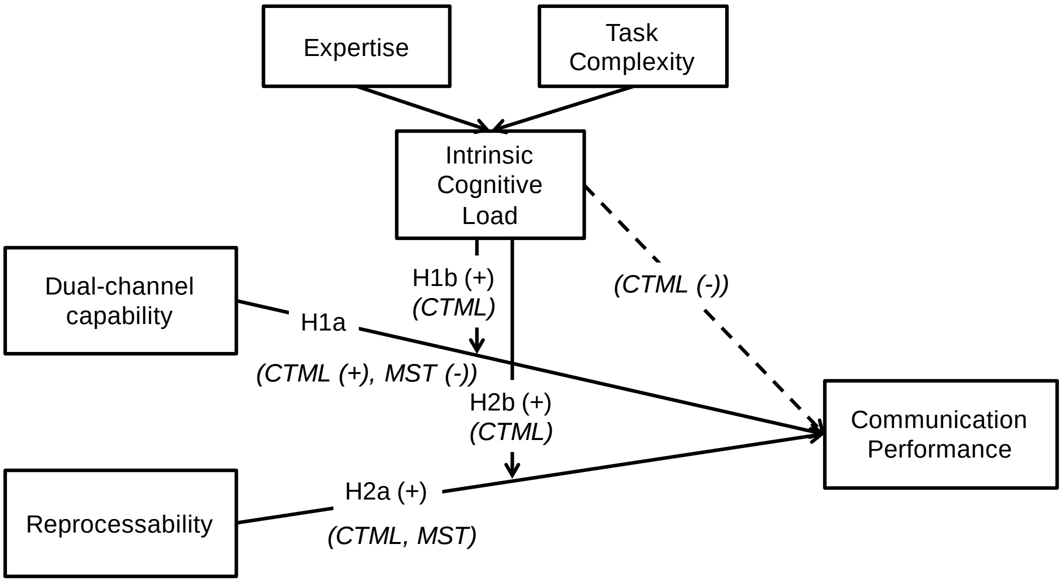

Figure II-8: A Theoretical Model of Media Choice and Communication Performance

Both theories suggest that dual-channel capability may impact the communication performance of conveyance processes. However, they suggest different causal explanations and directions of this effect. According to the CTML, using media with dual-channel capability improves communication performance because they allow relieving cognitive load by using the information-processing capacities of both channels. In contrast, MST posits that media with more symbol sets such as media with dual-channel capability will constrain communication performance in conveyance because they are associated with higher synchronicity. MST also emphasizes that written or typed media is lower in media synchronicity than physical, visual, and verbal symbol sets, advocating thereby the use of media that lack dual-channel capability for the purpose of information conveyance. In short, while the CTML predicts a positive association with dual-channel capability and communication performance, MST assumes a negative relationship. We anticipate:

H1a: The use of media with dual-channel capability is associated with either higher (CTML) or lower (MST) communication performance.

Although the CTML suggests that dual-channel capability may impact communication performance, this relationship may not universally manifest. Dual-channel capability is beneficial when it helps to shift cognitive load from one channel to another and this helps to avoid overload. Some material may be easy to understand and thus the processing capacity of one channel will suffice. Distributing this low load to two channels is not expected to yield significant benefits. The impact of dual-channel capability is therefore anticipated to be moderated by the intrinsic cognitive load of the learning material (Mayer and Moreno 2003; Sweller et al. 1998). Intrinsic cognitive load denotes the complexity inherent to a material for a particular learner. It abstracts from the extraneous cognitive load imposed by the way how material is presented to a learner (such as in text versus face-to-face). It is solely determined by the interaction of task complexity and the expertise of the recipient (Paas et al. 2003). The higher task complexity and the lower the expertise of the recipient, the higher is intrinsic cognitive load. Following the arguments made above, dual-channel capability is expected to yield benefits rather for high intrinsic cognitive load than for low intrinsic cognitive load. This suggests

H1b: The association between dual-channel capability and communication performance is moderated by intrinsic cognitive load. The higher the intrinsic cognitive load, the stronger is the association.

The CTML and MST make consistent predictions and provide similar causal arguments for the effect of reprocessability. MST suggests that reprocessability aids deliberate and retrospective information processing because it helps individuals to revisit messages to develop understanding. In a similar vein, the segmentation effect in the CTML suggests that understanding is eased when information is presented in a learner-controlled pace, which allows learners to revisit information. Conversely, when information is presented continuously, learners have to process additional information while they still attempt to mentally organize previously presented information. This adds extraneous cognitive load and therefore impairs understanding and learning. Media with high reprocessability enable learner-controlled pace and are therefore expected to yield better communication performance in conveyance processes. To summarize, we anticipate

H2a: The use of media with higher reprocessability is associated with higher communication performance.

Assuming the information-processing framework of CTML, the impact of reprocessability is anticipated to depend on intrinsic cognitive load. When intrinsic cognitive load is high, recipients will need substantial time to mentally organize the information provided in one message before proceeding with the next one. Interrupting this deliberate processing by continuously presenting additional information will add extraneous load to high intrinsic load and therefore quickly overload the recipient. Conversely, material with low intrinsic cognitive load may be efficiently processed by recipients, eliminating the need for reprocessing or segmenting. We therefore expect

H2b: The association of reprocessability and communication performance is moderated by intrinsic cognitive load. The higher intrinsic cognitive load, the stronger is the association.

Although MST specifies further moderating factors such as familiarity, trainings, past experience, and norms, these moderating factors are not proposed to moderate the impact of synchronicity on communication performance in conveyance processes. They are rather suggested to influence the degree to which conveyance versus converge processes will be salient. Because we focus on conveyance processes only, these moderating factors are not included into our model.

According to CTML, intrinsic cognitive load not only moderates the benefits from dual-channel capability and reprocessability, it will also have a direct impact on communication performance. Material that is intrinsically more complicate will be more difficult to understand. This relationship is indicated by the dashed line in Figure II-8. Although it is not in the realm of our research question, we have to account for this relationship in our data analysis.

Taken together, the theories make conflicting recommendations for media choice in application offshoring transitions. MST suggests that emails and documents may be most effective to allow vendor engineers to build mental models of the domain of their tasks, whereas the use of phone conference and, even more so, of face-to-face conversations will have detrimental effects for their understanding. In contrast, the CTML advocates the use of multimedia self-learning technologies and it would predict somewhat better communication for face-to-face sessions than for phone calls when intrinsic cognitive load is high. Moreover, the CTML does not recommend using documents under high intrinsic cognitive load because documents lack dual-channel capability.

3 METHODS

We conducted a multiple-case study (Yin 2009) to examine our theoretical model. The case study approach helped test our hypotheses within the context of the phenomenon of interest, i.e. SMOO transitions. This may contribute to the external validity of the findings. The method therefore promises insights into whether and how the context of our phenomenon shapes the associations between media choice and communication performance. The case study method also allowed gathering longitudinal data by observing how media choices change over time within cases and how these changes impact communication performance, which may increase the internal validity of our findings. Although the case-study method may have drawbacks with regards to the precision of measurement and causal inference, our research may have the potential to corroborate experimental research on both theories in particular with regard to the boundary conditions under which these theories may apply.

Our case selection enables theoretical replication. A case was the transition of a software-maintenance role to one vendor engineer. The software-maintenance tasks included software enhancements, defect corrections, and reengineering tasks. Within the cases, we used conveyance processes that were associated with communication outcomes as embedded units of analysis. One data point was therefore the communication performance associated with a conveyance process. The case selection allowed theoretical replication of intrinsic cognitive load because the cases varied in the amount of prior related experience and in task complexity. The cases also comprised both co-located and geographically dispersed transitions, allowing variance in the use of media between projects. This also helps obtaining a more realistic picture of knowledge flows in offshore outsourcing because case studies report that SMOO projects frequently involve both co-located knowledge transfers from SME to on-site coordinators and distributed knowledge transfers from SME or on-site coordinators to offshore staff (Dibbern et al. 2008; Gregory et al. 2009; Tiwari 2009; Wende and Philip 2011). Moreover, media use changed over time within the cases, which allowed further theoretical replication in our independent variables.

3.1 Case Contexts

A brief overview of the eight cases may help understand the contexts of the communication processes. The same Swiss bank was the client in all cases. The vendors were among the major Indian outsourcing service providers. While client teams were augmented by vendor staff in the cases 1, 4, 5, 6, 7, and 8, tasks were handed over from one vendor to another vendor in the cases 2 and 3 although SME of the client were also involved. The SME were described as at least moderately motivated to share their knowledge across cases, suggesting that issues of motivation did not overrule the cognitive issues on which this study is focused.

In the *cases 1 to 3*, software-maintenance roles in three data warehousing applications were handed over to three on-site coordinators of vendor A. In these cases, knowledge transfer was initially assisted by a coach. In each of the cases, he conducted two or three knowledge elicitation sessions based on the methodology described in Ackermann (2011) together with one or two SME and the vendor on-site coordinator. During the sessions, he drew a conceptual map of the knowledge involved in the maintenance role using the ontology described in the methodology. The map was created in Microsoft Visio and projected to a wall so that all participants could observe how the map was developed during the session. The pattern of the conversation was mainly driven by the coach asking questions and the SME answering these questions. The on-site coordinators sporadically asked questions. Before and after the sessions, the on-site coordinators interacted informally with the SME, participated in other formal presentations for the purpose of knowledge transfer, studied documents, and worked on maintenance tasks.

In the *cases 4 and 5*, software-maintenance roles in one banking transaction control software package were handed over to two on-site coordinators of vendor B. There were no coached knowledge elicitation sessions. The first on-site coordinator participated in several formal presentations held by a client SME at the beginning of his stay. In addition, both on-site coordinators interacted with client SME and with each other, studied documents, and worked on maintenance tasks.

In the *cases 6 to 8*, reports had to be migrated from one data warehousing technology to another. In the *cases 6 and 7*, the vendor engineers were responsible to define the architecture of the new technology. After unsuccessful cooperation with the first vendor engineer *(case 6)*, the second vendor engineer took over his responsibility *(case 7)*. Whereas the engineer in *case 6* was located in India and communicated with the client SME via email, documents, and phone, the engineer in *case 7* interacted on site with the client team. In *case 8*, two vendor engineers located in India designed and implemented the reports. They were provided with the existing report queries and examples of requirement specifications. In addition, they interacted with the vendor engineer of *case 7*, who was on site. The communication focused on the knowledge required for the low-level design and implementation of the report migration such as the meaning of database fields.

Appendix II-4 gives more information on the cases.

3.2 Data Collection

Data were gathered through semi-structured interviews, the observation of sessions, and document analysis. Table II-22 gives an overview of the data sources. The interviews served to understand the process of knowledge transfer and the associated outcomes from the perspectives of vendor engineers, SME of the client or another vendor, client management, and vendor management. The interviews were tape-recorded and transcribed. The first author participated as an observer of sessions conducted for knowledge transfer. Interview transcripts and fields notes from observed sessions amounted to 141,142 words. The analyzed documents included requirements specifications, design documents, documents created as a result of knowledge elicitation sessions, software documentation, knowledge transfer plans, and email notes. Requirements specifications and design documents helped to code intrinsic cognitive load by shedding light on task complexity. However, they were not considered supportive information because they described the goal state and initial state of the software-maintenance problems rather than being an additional aid to make sense of the task domain. Conversely, software documents such as descriptions of the software architectures were potential sources of supportive information. Multiple informants and multiple types of data sources served for triangulation, increasing the internal validity of our analysis.

3.3 Data Analysis

Our data analysis process comprised several steps. First, we screened the data for conveyance processes. We identified those communication processes as conveyance processes in which helping the vendor engineers to create or revise a mental model of the task domain was the goal. We thereby excluded processes that aimed at, for example, agreeing on how to solve a software-maintenance problem or how to proceed with knowledge transfer because we considered them to be convergence processes and thus outside the scope of this study. We thus classified communication processes based judgments of the goals associated with these processes. This choice has been made because the main propositions of MST distinguish between "communication processes in which convergence on meaning is the goal" (Dennis et al. 2008, p. 583) and "communication processes in which the conveyance of information is the goal” (Dennis et al. 2008, p. 583). Hence, using the goal of communication as the criterion for discriminating conveyance from convergence is consistent with MST. Conveyance processes may therefore also include episodes during which recipients ask questions or triangulate information by consulting additional sources (Dennis and Valacich 1999, p. 4) as long as the goal of these activities is to help the recipient to understand the task domain. We included all conveyance processes for which communication performance could be coded based on the available data and the coding scheme shown in Appendix II-5. Two authors were involved in the process of screening the data for conveyance processes. Both agreed on the classification of each of the processes as conveyance processes. This process resulted in 20 data points.

Table II-22: Data Sources

Case	*Interviews: number of interviews/ number of interviewees*				*Ob-served sessions*	*Docu-ments*	*No. of Data Points*	*Timing of Data Collection*
	Vendor engineers	*SME*	*Client mgrs.*	*Vendor mgrs.*				
1	5/1	2/2	2/1	-	4	20	8	Real-time
2	2/1	2/2	3/3	-	2	8	2	Real-time
3	3/1	2/2		-	2	16	2	Real-time
4	2/1	2/2	1/1	-	2	2	2	Retrospective
5	2/1			-			1	Real-time
6	-	1/1	1/1	1/1	-		1	Retrospective
7	2/1				-	5	1	Real-time
8	2/2				-		3	Real-time
Total	18/8	9/9	7/6	1/1	10	51	20	

Second, we coded the data according to the dimensions of our theoretical model. It was evident that many communication processes made use of a set of two or three media rather than one medium only. Although MST focuses on the capabilities of a single medium rather than of sets of media, Dennis et al. (2008) emphasize that MST may be used to predict how media complement each other when they are used together. Consistent with this argument, we coded the media capabilities of the set of media involved in a conveyance process. We also coded intrinsic cognitive load and communication performance in each conveyance process. Appendix II-5 shows the coding rules that we developed to this end. Two authors coded the data independently and compared their coding. All disagreements could be settled either by identifying more evidence from the cases or by clarifying ambiguities in the coding rules.

Third, we tested whether the predictions of our theoretical model were able to explain the data. To this end, we performed pattern matching (Yin 2009) and we looked for claims of causal relationships explaining communication performance in the interview data as a source of triangulation (Miles and Huberman 1994). In pattern matching, we diagramed the interactions of dual-channel capability and intrinsic cognitive load against communication performance and the interactions of reprocessability and intrinsic cognitive load against communication performance. We further looked for episodes of conveyance processes in which changes of independent variables over time were associated with changes in communication performance. We also conducted cross-case analyses to see how the context of the cases is associated with the patterns in the embedded units of analysis.

4 RESULTS

The results strongly support the predictions made by the CTML in H1a and H1b, while we see only weak support for H2a and H2b. That is, communication performance was able to be explained by the interaction of dual-channel capability and intrinsic cognitive load anticipated in the CTML and does not follow the pattern expected in MST in H1a and H1b. We next give a quantitative overview of our results and present qualitative data that sheds further light on these results.

4.1 Dual-Channel Capability

Table II-23 shows the number of conveyance processes by dual-channel capability and communication performance. As predicted by the CTML, we find a relatively high number of 11 data points that involved media with dual-channel capability and yielded high communication performance, and a relatively high number of 6 data points with media without dual-channel capability and low communication performance. This pattern stands in contrast to MST, which predicts that the media with larger symbol sets (i.e. the media with dual-channel capability) are less suited for conveyance than media with smaller symbol sets (i.e. the media without dual-channel capability). However, H1a cannot fully explain the data in Table II-23. Three conveyance processes yielded successful communication although media without dual-channel capability were used. The moderation effect formulated in H1b is able to explain these values. Table II-24 shows the data by intrinsic cognitive load, dual-channel capability, and communication performance. Consistent with H1b, dual-channel capability only made a difference in our data when intrinsic cognitive load was high. As the lower part of Table II-24 indicates, communication was always successful when intrinsic cognitive load was low or medium, irrespective of the media used. It was only in those data points with high intrinsic cognitive load that dual-channel capability was associated with higher communication performance. It is striking that the interaction of dual-channel capability and intrinsic cognitive load anticipated in the CTML was able to fully explain communication performance.

Table II-23: Dual-Channel Capability and Communication Performance

	Low or Medium Communication Performance	*High Communication Performance*
Dual-Channel Capability: Yes	0	11
Dual-Channel Capability: No	6	3

Table II-24: Interaction of Dual-Channel Capability and Intrinsic Cognitive Load on Communication Performance

Number of Data Points with High Intrinsic Cognitive Load		
	Low or Medium Communication Performance	*High Communication Performance*
Dual-Channel Capability: Yes	0	6
Dual-Channel Capability: No	6	0
Number of Data Points with Low or Medium Intrinsic Cognitive Load		
	Low or Medium Communication Performance	*High Communication Performance*
Dual-Channel Capability: Yes	0	5
Dual-Channel Capability: No	0	3

Interview statements may help further understand the interaction of intrinsic cognitive load and dual-channel capability in the data. For instance, the vendor engineer in case 1 was provided with an overview document at the beginning of the transition (no dual-channel capability). The following dialog took place in the fifth month of the transition. It sheds light on how the performance of the communication of knowledge through this document changed over time:

> *"How was this document helpful to you?" (First author) – "(Laughing) ... Ok, I went through the document and first I did not understand anything. Most of the things I could not understand. Whichever is easy to understand, I did not do that. I did check with [SME 2] on some parts. Other parts, when I worked on it, I realized that this is how it fits into the picture." (Vendor engineer, case 1) – "So later on, you had again a look at the presentation and said 'Ok, now I understand'?" – "Yes, it fits into the picture." (Vendor engineer, case 1) – "Then it was also helpful to look again at this presentation?" (First author) – "Yes, because it helps. At first, you do not have any knowledge. It is like a layman. But next time you*

see, you know what this part does and what this part does and then you know what the over-all picture is. It really helps." (Vendor engineer, case 1)

The statement suggests that he was initially not able to construct a meaningful mental model from the information provided in the document (low communication performance). He perceived himself as a "layman" in the domain of the content of this document, which indicates low expertise and thus high intrinsic cognitive load. This perspective is consistent with the content analysis of the document and with the evolution of the experience he made in the domains of the document during transition. The document described the data flow between components and the structure of some database tables of the custom-built data warehouse. Although the vendor engineer had considerable previous experience in data warehousing technologies, the components, their relationships, and tables of the data warehouse were specific to this software application. Thus, much of the information provided in the document could not be related to prior experience, indicating low expertise and thus high intrinsic cognitive load (see also study 1). Consistent with the directions of the hypotheses H1a and H1b implied in the CTML, the use of a medium without dual-channel capability combined with high intrinsic cognitive load was associated with low communication performance.

However, the communication of knowledge through this document seems to have improved over time. After he had worked on tasks, he took again a look at the document (another data point). Some of the tasks on which he had worked on in the meantime concerned the domains of the information displayed in the document. When reading the document again after the work on these tasks, he perceived the information provided as helpful (high communication performance). The expertise literature strongly suggests that expertise develops mainly through practice (Ericsson et al. 1993). His practice through the work on tasks is therefore likely to have resulted in an increase of expertise and thus in a reduction of intrinsic cognitive load. Under this relieved intrinsic cognitive load, he was obviously able to build a mental model from the information in the document even though it lacked dual-channel capability. The CTML offers the following explanation for the difference in communication performance. The benefits from dual-channel capability diminished because the capacity of the visual channel was sufficient to mentally organize the information in the document only after expertise had increased and intrinsic cognitive load had decreased as a consequence.

While these two related data points are indicative of a relationship between intrinsic cognitive load and communication performance, more evidence is necessary to illuminate the interaction of intrinsic cognitive load and dual-channel capability in the qualitative data. It is therefore interesting to consider other conveyance processes that occurred at the beginning of the transition in case 1, involved similar content, but media with dual-channel capability. The knowledge elicitation sessions (see section 3.1 of this study) mainly dealt the same content as the document (i.e. the data flow between the data warehousing components). They were also conducted at the beginning of the transition so that expertise can be assumed to have been rather low at that time. The set of media used in this session had dual-channel capability because the auditory explanations given by the SME were visualized at the same time by the coach using Microsoft Visio. Here is how the vendor engineer perceived the communication in the knowledge elicitation sessions:

> *"I think the sessions were really good because I got a bigger picture of the project. If I hadn't had these sessions, I would just have jumped into my task without knowing what the bigger picture is. Now, as I had the sessions, I know what the different parts within the project are. When I'm working on a task, I can kind of relate to what I've already learned in the sessions. I kind of realize. It makes things easier to understand."(Vendor engineer, case 1)*

Although the content of the communication in the knowledge elicitation was similar to the content in the document and although he suffered from the same low expertise level at that point, he seems to have been able to build a mental model of the communication in the sessions (high communication performance). The difference in communication performance cannot be explained by MST, which would suggest that the document is more effective for conveyance than the face-to-face session. Conversely, the explanations proposed by the CTML are consistent with the data: At the early stage of the transition, face-to-face presentations allowed to shift some of the high intrinsic cognitive load from the visual to the auditory channel. This helped avoid overload. At a later stage, offloading to the auditory channel was not required because the lower intrinsic cognitive load did not exceed the processing limits of one channel.

This pattern is not idiosyncratic to case 1. A set of related conveyance processes in case 8 also corroborate the association between dual-channel capability and communication performance. In this case, the vendor team replaced simple phone conferences by conferences that allowed screen-sharing. Here is the reasoning that the vendor manager, who was involved in these meetings, gave for this change:

"There is a gap between verbal communication to visual and face-to-face communication. There is a lot of body language missing during phone calls and there is a lot of visual perception that a person cannot pass over the phone. This is one reason. Another reason is that while talking on the phone you are limited to anticipating what the other person wants to pass on to you. I can ring you and talk to you. You can visualize and conceptualize the things. If the conceptualization does not seem very clear to you it usually helps to see the object. You need to see or even feel it. This is another reason why we use videoconferences." (Vendor manager, cases 6 to 8)

The statement parallels the theoretical arguments made in the CTML in that it stresses that complex content is easier to grasp when visualization aids accompany spoken words. Again, MST cannot predict the increase of communication performance in conveyance processes by shifting to a medium that has higher synchronicity.

Cross-case analysis further corroborates this pattern. Documents (i.e. media that lack dual-channel capability) were perceived to be effective during the self-study by the vendor engineers in the cases 4 and 5. In these cases, the vendor engineers could benefit from previous experience with the same software package (see also study 1). Unlike the cases 1 to 3, in which the software architecture was specific to the project and thus novel to the vendor engineers, the cases 4 and 5 confronted the vendor engineers with domains with which they were largely familiar. The higher start levels of expertise in the cases 4 and 5 were therefore associated with lower intrinsic cognitive load, a condition under which media without dual channel capability may be effective according to the CTML. The cases 6 and 7 also replicate the pattern. In case 6, the vendor engineer had little prior experience with the type of architectural decisions he was supposed to make. The communication through emails, documents, and phone lacked dual channel capability. As anticipated by CTML, communication was ineffective as indicated by the failure of the engineer to complete the task and by the following statement of the SME:

"I had to realize that he had no idea of what I was talking about." (SME 1, case 6)

Conversely, conveyance was effective in case 7 which involved the same task as case 6, but another vendor engineer with prior related experience, and which was conducted on-site. This pattern is again consistent with the prediction of CTML.

The association between dual-channel capability and communication performance are corroborated by further interview statements on causal linkages that are consistent with the findings of pattern matching. These statements are given in Appendix II-6. Taken together, we see strong evidence for an interaction of dual-channel capability and intrinsic cognitive load that follows the causal reasoning of the CTML.

4.2 Reprocessability and Intrinsic Cognitive Load

MST and the CTML suggest that communication performance in conveyance processes is higher when media with high reprocessability are used. CTML adds that this effect is stronger under high intrinsic cognitive load. Table II-25 shows the number of data points by reprocessability. Table II-26 shows the number of data points by intrinsic cognitive load and reprocessability.

Table II-25: Reprocessability and Communication Performance

	Low or Medium Communication Performance	*High Communication Performance*
Low Reprocessability	0	2
Medium Reprocessability	2	9
High Reprocessability	4	3

Table II-26: Interaction of Reprocessability and Intrinsic Cognitive Load on Communication Performance

Number of Data Points with High Intrinsic Cognitive Load		
	Low or Medium Communication Performance	*High Communication Performance*
Low Reprocessability	0	2
Medium Reprocessability	2	4
High Reprocessability	4	0
Number of Data Points with Low or Medium Intrinsic Cognitive Load		
	Low or Medium Communication Performance	*High Communication Performance*
Low Reprocessability	0	0
Medium Reprocessability	0	5
High Reprocessability	0	3

The tables do not show the pattern that would be expected based on H2a and H2b. In particular, while H2a and H2b would predict low communication performance for low reprocessability and high cognitive load and high communication performance for high reprocessability under high cognitive load, we have not obtained any such data point. Pattern matching does therefore not support H2a and H2b. Yet, the associations between reprocessability and communication performance are not as pronounced as the associations between dual-channel capability and communication performance. Medium amounts of reprocessability were associated with both high and low communication performance. We are not aware of a theoretical rationale for a negative relationship between reprocessability and communication performance nor do we find suggestions for causal explanations for a negative relationship in our data. The pattern may be spurious and caused by the correlations between dual-channel capability and reprocessability. If we code *dual-channel capability: yes* to 1 and *dual-channel capability: no* to 0 and if we code low, medium, and high reprocessability to 1, 2, and 3, we obtain r = -.77, which indicates a very strong correlation between the two media capabilities. We conclude therefore that our pattern matching results do not permit any inference on the associations between reprocessability and communication performance. More elaborate statistical tests such as regression, which would control for the influence of dual-channel capability, are not admissible given the nature of our data.

Although pattern matching does not permit inference on reprocessability, we find some interview statements that follow the causal arguments made by MST and by the CTML about the benefits of reprocessability (see also Appendix II-6). For instance, the dialog reproduced at the beginning of the section 4.1 of this study is also indicative of the benefits of reprocessability. The vendor engineer was able to create a meaningful mental model only when he read the document again after he had worked on related tasks. The effective conveyance of information when he read the document again was therefore contingent on the reprocessability of the document, which allowed him to process the message again at a later stage. In case 8, the SME emphasizes the benefits of the ability looking up guidelines during the work, which may imply aspects of reprocessability of these guidelines:

> *"I think it was the most important input from our side because they were new to [the client] and therefore they did not know how we usually proceed in our projects. It was certainly helpful that they could look up the guidelines." (SME 1, case 8)*

Taken together, our analysis indicates that dual-channel capability may have been the dominant media capability antecedent to communication performance in conveyance processes when intrinsic cognitive load was high. While the arguments made by MST and CTML on the benefits of reprocessability seem conclusive and are slightly supported in some reasoning of our study participants, our research method did not permit inference on the relationships between reprocessability and communication performance.

It may increase the internal validity of our analysis to discuss alternative explanations for our data on communication performance. An alternative explanation may be that the three omitted media capabilities transmission velocity, rehearsability, and parallelism may account for our findings. Two arguments may be put forward against this explanation. First, assuming the propositions in MST, these dimensions do not predict the pattern of our results. According to MST, the five media capabilities are strongly correlated, which allows the theory to subsume their collective influence in one sole mediating construct of media synchronicity. If we included the three remaining media capabilities parallelism, transmission velocity, and rehearsability in our model, the predictions of MST would not substantially change and would remain inconsistent with our data. Second, we did not find any interview statements that stress the roles of parallelism, transmission velocity, and rehearsability.

Another potential explanation is that certain media may have an intrinsic fit with certain types of information and that this intrinsic fit caused communication performance in our data. For instance, the data flow in a software system may be more easily grasped from a pictorial than from a verbal form because the information is of a spatial nature. However, this would not explain why vendor engineers struggled with information provided in documents under high intrinsic cognitive load. Instead, the interaction of dual-channel capability and intrinsic cognitive load was able to fully explain communication performance.

5 DISCUSSION

Prior work indicates that vendor engineers may effectively learn software-maintenance tasks by engaging in authentic learning tasks that impose neither too high nor too low cognitive load. Although supportive information may be one strategy to reduce cognitive load at the beginning of transitions, study 1 of this dissertation showed a rather weak negative relationship between supportive information and cognitive load. A possible interpretation of the weak relationship was that supportive information may be differentially effective and that media choice may influence the effectiveness of supportive information. The study at hand aimed to shed light on this interpretation. We examined whether media choice may have impact on communication performance in supportive-information activities. We drew on MST and the CTML to demonstrate that the theories make conflicting predictions on the effects of dual-channel capability or symbol sets and consistent predictions on the effects of reprocessability. We presented the results of a multiple-case study intended to test the predictions made by the theories.

In our data, we saw strong support for the assertion of the CTML according to which media with dual-channel capability are associated with higher communication performance than media without dual-channel capability when intrinsic cognitive load is high. This interaction effect not only perfectly explained our data; it was also corroborated by the explanations given by our study participants for communication outcomes. This suggests that media should be chosen based on the intrinsic cognitive load that the material or the related task imposes on the engineer. When material is expected to be complex for the engineer, media should be preferred that permit to simultaneously transfer both visual and auditory information. This calls for the use of face-to-face meetings with visualization aids such as in the knowledge transfer method described by Ackermann (2011), of video conferences, of phone conferences assisted by screen sharing, and of recordings that simultaneously show coherent visual information such as the software and auditory explanations. When the use of media with dual-channel capability imposes additional costs, project managers may decide based on the expected intrinsic cognitive load. Projects that involve at least moderately specific software will frequently confront vendor employees with unfamiliar domains and thus involve high intrinsic cognitive load (see also study 1). In these projects, managers may be well advised to incur the costs to enable the use of dual-channel media.

Our results also imply that documents may not be effective means to convey supportive information in domains that involve high intrinsic cognitive load such as in many application offshoring contexts (Chua and Pan 2008; Dibbern et al. 2008; Krancher and Dibbern 2012). These recommendations are consistent not only with the results of our study, but also with the theory developed through a series of controlled experimental research on the conveyance of complex information in multi-media learning environments (Mayer and Moreno 2003). Documents may have an unfortunate characteristic if used to provide supportive information: They may be least effective when they are most needed. When vendor engineers are confronted with at least moderately complex tasks in largely unfamiliar domains, their need for cognitive load reduction strategies will be particularly high (see also study 1). It is in these situations that documents may be least appropriate to effectively convey the supportive information to the vendor engineers. They may thus fail to fulfill the need for cognitive load reduction. This finding is noteworthy because vendors advertise their codification strategies as means to convey knowledge to offshore teams as a short informal search of transition approaches on vendor homepages reveals. While the codification of knowledge may seem appealing in order to retain knowledge despite high personnel turn-over in vendor teams, our results suggest that this strategy may pay off only in settings in which tasks are rather simple, e.g. because they follow sequential logics, and in which the vendor engineers can draw on their prior experience because the software is not highly specific (see also study 1). Beyond these realms, optimistic views on the usefulness of documents for the purpose of knowledge transfer do not seem justified.

The findings offer additional explanations for the weak the relationship of supportive information and cognitive load observed in study 1. When task complexity is relatively high, when prior related experience is limited, and when documents are the most easily available source of supportive information, vendor engineers may spend particularly long time for studying documents, struggling to make sense of the information provided. Initially high cognitive loads may thus result in particularly strong increases of supportive information (i.e. of additional time dedicated to study the documents). Complex interactions and feedback processes may thus operate between media choice, task complexity, expertise, cognitive load, and the time dedicated to supportive information. Future work with higher sample sizes may examine these interactions.

The study also may also have theoretical implications outside the realm of SMOO transitions. First, our research may help sharpen the boundary conditions of the CTML and of MST. The CTML has originally been developed to inform the design of multimedia self-learning environments. Our results indicate that the theoretical arguments of the CTML are worth of being tested in work contexts in which information with high intrinsic cognitive load needs to be conveyed through appropriate media. For instance, the CTML may have implications for the design of knowledge management systems. MST aims at predicting the impact of media on the effectiveness of organizational communication processes. The results of this study indicate that the predictions of the current form MST may extend only to a limited degree to contexts in which information that imposes high intrinsic cognitive needs to be learned. Hence, our study does not reject MST, which originates from research on collaborative problem-solving rather than individual learning, but it may help qualify the boundary conditions of MST in its current form. Nevertheless, our work may also provide opportunities for the further development of MST. For instance, given that Dennis et al. (2008) refer to cognitive load several times in their article, they may consider including intrinsic cognitive load as a moderator in conveyance processes. In light of the strong empirical support for the benefits of dual-channel capability for conveyance processes in CTML research, it may also be worth to reconsider how large symbol sets (Dennis et al. 2008) impair conveyance. The CTML may be a valuable source for further developing the arguments in this regard.

Our study has several limitations. First, our findings rely exclusively on qualitative data gathered in the natural environment of the phenomenon of interest. This implies drawbacks for the precision of the measurement of our variables and for the control of environmental conditions. Yet, the unambiguous results for dual-channel capability are unlikely to be explained by measurement error. We also involved two coders to increase the reliability of our data analysis. In addition, we partially relied on longitudinal data and followed best practices anchored in case-study research to increase the internal validity of the study such as triangulation of data by using multiple sources of evidence and theoretical triangulation by using two rival theories. Having ruled out some alternative explanations may make it less likely that our results are explained by variables for which we did not control. The corroboration of our results with related controlled experimental research (Mayer and Moreno 2003) may also the trust into validity of our analysis. Second, our study does not permit statistical generalization. In addition, we did not statistically reject hypotheses. This is because of the low number of data points and the non-random selection of data points. Yet, pattern matching may support theoretical generalization (Lee and Baskerville 2003; Yin 2009) because theory was able to explain our results across cases. Third, we did not theorize on how cultural distance and language barriers impact the predictions of communication performance because our data lacked variance in these dimensions. Future work may include these constructs into theorizing.

Future research may build on the findings of this study. The study may be replicated in other client organizations. Mixed-methods research procedures (Tashakkori and Teddlie 1998) may also heal some of the weaknesses of this study. For instance, qualitative data collection may be complemented by quantitative, structured measurement of the constructs of the study through survey items. Future research may examine cost-benefit trade-offs in media choice in SMOO transitions. Another avenue is experimental research on conveyance processes in which selected media capabilities and intrinsic task complexity are manipulated. Such research could also shed further light on the interactions between media choice, task complexity, expertise, cognitive load, and the time dedicated to supportive information.

APPENDIX II-4: CASES OVERVIEW

Case	*Task*	*Vendor engineer*	*Geographical dispersion*
1	Software enhancements and defect corrections in the large, highly custom-built data warehousing application 1	Experienced in data warehousing technologies, no experience with the client	Co-located (Switzerland)
2	Software enhancements and defect corrections in the medium-sized, moderately custom-built data warehousing application 2	Experienced in data warehousing technologies, no experience with the client	Co-located (Switzerland)
3	Software enhancements in the large, moderately custom-built data warehousing application 3	Experienced in data warehousing technologies, no experience with the client	Co-located (Switzerland)
4	Software enhancements and defect corrections of a medium-sized, standard software package for the control of banking transactions	Experienced with the software package, previous experience with the client	Co-located (Switzerland)
5		Some experience with the software package, some previous experience with the client	Co-located (Switzerland)
6	Reengineering the data warehousing application 4 by migrating existing reports to a new technology	Little experience with this type of architectural decisions, no previous experience with the client	Distributed (Switzerland, India)
7		Experienced with this type of architectural decisions, no previous experience with the client	Co-located (Switzerland)
8[4]	Design and implementation within the reengineering of the data warehousing application 4 (see cases 6 and 7)	Experience with data warehousing technologies, no previous experience with client	Distributed (Switzerland, India)

[4] Although two vendor engineers were involved, we condensed the data to one case because the recipients were similar in experience, they were mainly involved in the same communication processes, and these processes yielded similar outcomes for both recipients.

APPENDIX II-5: CODING RULES AND SAMPLE QUOTES

Construct	*Category*	*Rule*	*Sample Quotes*
Dual-Channel Capability	Yes	The set of media allows the sender to simultaneously transfer both auditory and visual information (i.e. power point presentation held face-to-face by the SME, knowledge elicitation session with drawing tool (Microsoft Visio), knowledge elicitation session with development tool, desktop sharing conference)	• *“We had one session this Tuesday, it was a meeting […].”* • *“I did have discussions with [the SME] for this.”* • *“[In this meeting, the SME] gave an overview of what happens in the project”*
	No	Else (i.e. study of PowerPoint presentations, technical documentation, code, server lists, etc.; phone calls; emails)	• *“[…], I sent this document to the offshore team by email.”* • *”Yes, we only were in touch over the phone and with emails.”*
Reprocessability	High	The recipient has full discretion of the timing of information presentation and can reprocess the information presented at a later time.	• *“There are documentations.”* • *“Then I just go to the Excel sheet and see [the definition], just as an example.”*
	Medium	The recipient does not have full discretion of the timing of information presentation, but can reprocess some of the information at a later time without having to recall it from memory.	• *“I did have discussions with the SME for this. They [the discussions] were quite long because I was not sure of what needs to be done. He has a document to explain what needs to happen and what needs to be implemented. So that was pretty helpful.”* • *“[…], there were ppt presentations and you can raise questions and they'll answer you.”*
	Low	The recipient does not have full discretion of the timing of information presentation and does not have the possibility to reprocess the information at a later time without recalling it from memory.	• *„He takes time and explains what actually happens. He gives an overview first and then explains what actually happens.“*

Construct	*Category*	*Rule*	*Sample Quotes*
Intrinsic Cognitive Load	High	The recipient has no previous related experience or the recipient has some previous experience but the task or material is complex (Interdependencies of information elements or acts are salient.).	• *"I think that he lacked the knowledge to carry out such a project."* • *"If someone has gained 15 years of experience with this technology it should not be necessary to talk about basics."*
	Medium	Else	• *"[…], I also had knowledge on those things [the application]. […]. It's [the presentation] only about the project structure, how the project activities are happening. This was totally new for me, I didn't know anything about it."*
	Low	The recipient has considerable previous related experience and the task or material is simple (interdependencies are not salient).	• *"He knew what he was doing and he possessed the necessary knowledge."*
Communication Performance	High	The recipient understood the content of the message or perceived communication as effective or was able to apply the information presented and there is no indication for low communication performance	• *"He conveyed us how and what could be improved by a better design. Through this, we gained a clearer understanding."*
	Medium	There is mixed evidence, i.e. evidence of both high and low communication performance.	• *"Initially, the analysis was a little bit difficult since we did not have access to the client's data."; "Once we understood it, it was easy to complete this task."*
	Low	The recipient did not understand the message or perceived the communication as ineffective	• *"I had to realize that he had no idea what I was talking about."*

APPENDIX II-6: INTERVIEW STATEMENTS ON CAUSAL RELATIONSHIPS

Dual-Channel Capability → (+) Communication Performance (H1a; CTML)	
Statement	*Rationale*
“Sometimes if you have to explain something it is necessary to show it to the person you are talking to on a screen”	In the opinion of the interviewee, the use of both channels improved understanding.
“Well, an image sometimes says more than 1000 words. If I cannot explain it verbally I take the developer [a tool that visualizes entity relationship diagrams] and show them how the different components depend on each other.“	
Reprocessability → Communication Performance (H2a; CTML and MST)	
Statement	*Rationale*
“Yes, because it helps. At first [when you look at the documents], you do not have any knowledge. It is like a layman. But next time you see [the documents], you know what this part does and what this part does and then you know what the overall picture is.”	The opportunity to reprocess information at a later time enabled the recipient to understand the documents.
„We communicated via email and afterwards discussed it during videoconferences where we showed them screen shots, etc. These guidelines definitely facilitated their work.”	The possibility to look up formal guidelines was perceived as helpful to complete the task.
Intrinsic Cognitive Load → Communication Performance (CTML)	
Statement	*Rationale*
“The reason why the work package did not succeed was that the vendor was too optimistic about their own skills. They took on too much. They said that they are capable of doing it and that they had conducted similar projects several times.”	Lack of experiences is perceived to be cause of unsuccessful communication.
“For example the same column names were used in [technology 1] and in [technology 2]. This made it easy to understand”	A simple one-to-one relationship between components indicates low task complexity and thus low intrinsic cognitive load, which eased understanding according to the statement.

CHAPTER III

GOVERNING THE LEARNING OF SOFTWARE-MAINTENANCE TASKS

STUDY 3

GOVERNING INDIVIDUAL LEARNING IN THE TRANSITION PHASE OF SOFTWARE-MAINTENANCE OFFSHORING: A DYNAMIC PERSPECTIVE

ABSTRACT

Prior studies suggest that clients need to actively govern knowledge transfer to vendor staff in offshore outsourcing. In this paper, we analyze longitudinal data from four software-maintenance offshore outsourcing projects to explore how governance and the individual learning of vendor engineers interact over time. Our results suggest that self-control is central to learning, but may be hampered by low levels of trust and expertise at the outset of projects. For these foundations to develop, clients may initially need to exert high amounts of formal and clan controls to enforce learning activities against barriers to knowledge sharing. Once learning activities occur, trust and expertise can increase and control portfolios may show greater emphases on self-control.

1 INTRODUCTION

Companies continue to outsource software-maintenance work to offshore vendors given scarce domestic personnel and labor cost advantages. Yet, many offshore outsourced projects do not meet the clients' expectations. Tedious knowledge transfer (the process through which vendor staff acquires the knowledge to perform the maintenance tasks) is a frequent source of failure (Chua and Pan 2008; Dibbern et al. 2008; Oshri et al. 2011; Westner and Strahringer 2010). Ineffective knowledge transfer may result in tasks not taken over by offshore teams (Chua and Pan 2008) and in extra costs for knowledge transfer, specification, coordination, and control that offset the savings through labor cost advantages (Dibbern et al. 2008).

Software maintenance may be particularly prone to problematic knowledge transfer. Software maintenance is a cognitively demanding task, in which engineers heavily rely on their tacit knowledge to identify where maintenance actions need to be made and to conceive solutions (Pennington 1987; Von Mayrhauser and Vans 1995). Their performance is primarily driven by their knowledge of the particular software application system (Boh et al. 2007). In software-maintenance offshore outsourcing (SMOO) projects, this knowledge needs to be transferred during the transition phase at the outset of projects. During transition, vendor employees are typically present at the client site to acquire the knowledge through extensive interaction with subject matter experts (SME) such as former delivery personnel (Chua and Pan 2008; Tiwari 2009). Yet, the coexistence of SME and vendor staff makes the transition phase a particularly costly phase. This poses a dilemma to management. Whereas tacit knowledge is difficult to transfer (D'Eredita and Barreto 2006) and central to project success, long transitions may jeopardize the business case of offshoring. Strategies for effectively managing knowledge transfer during transition may help mitigate this dilemma.

Prior studies suggest that client management needs to actively govern knowledge transfer in offshore outsourcing to facilitate effective knowledge transfer (Dibbern et al. 2008; Gregory et al. 2009). Consistent with the information systems (IS) outsourcing literature, we use the term governance to refer to structures and actions that guide behavior towards desired objectives (Goo et al. 2009; Huber et al. 2011). Governance may be needed because knowledge transfer in offshore outsourcing faces barriers specific to the nature of offshore outsourcing such as cultural and semantic distances (Dibbern et al. 2008), frequently low prior related experience (Dibbern et al. 2008), and interorganizational conflict. For instance, client management may need to establish formalized communication structures and to define knowledge transfer procedures to enforce effective interactions against low motivation of client staff (Gregory et al. 2009).

Although effective knowledge transfer therefore seems to depend on governance, we know little on how existing governance and control theories apply to the governance of knowledge transfer. The relationship between governance and knowledge transfer may be far from simple. Both governance portfolios (Choudhury and Sabherwal 2003; Kirsch 2004) and knowledge evolve over time. Knowledge may be both an antecedent to and an outcome of governance (Sydow and Windeler 2003). In SMOO transitions, this raises questions such as: Does knowledge transfer governance need to be adapted over time to accommodate changes in the expertise of vendor engineers?

In this research, we strive to explore the interaction of governance and knowledge transfer in SMOO. Although knowledge may be transferred at various levels, we focus on the individual learning of vendor engineers because of the central role of individual maintainers' knowledge. This paper thus addresses the following research question:

> *How do governance and individual learning interact over time during SMOO transitions?*

The paper is organized as follows. In section 2, we present our conceptual framework. In section 3, we describe how we conducted a longitudinal multiple-case study to explore the relationship between governance and individual learning. In Section 4, we describe and discuss the interactions of governance and learning in the cases and build a process theory. In section 5, we discuss implications of our study.

2 CONCEPTUAL FRAMEWORK

Consistent with Eisenhardt's (1989b) recommendations for theory building, we entered the field with an a-priori selection of constructs taken from the literature, but without hypotheses. Figure III-1 shows the constructs along with the theoretical lenses that suggest their relevance. Next, we provide rationales for the choice of the theoretical lenses and relate the constructs to prior literature. Definitions of the constructs included in the developed theory are given in Appendix III-1.

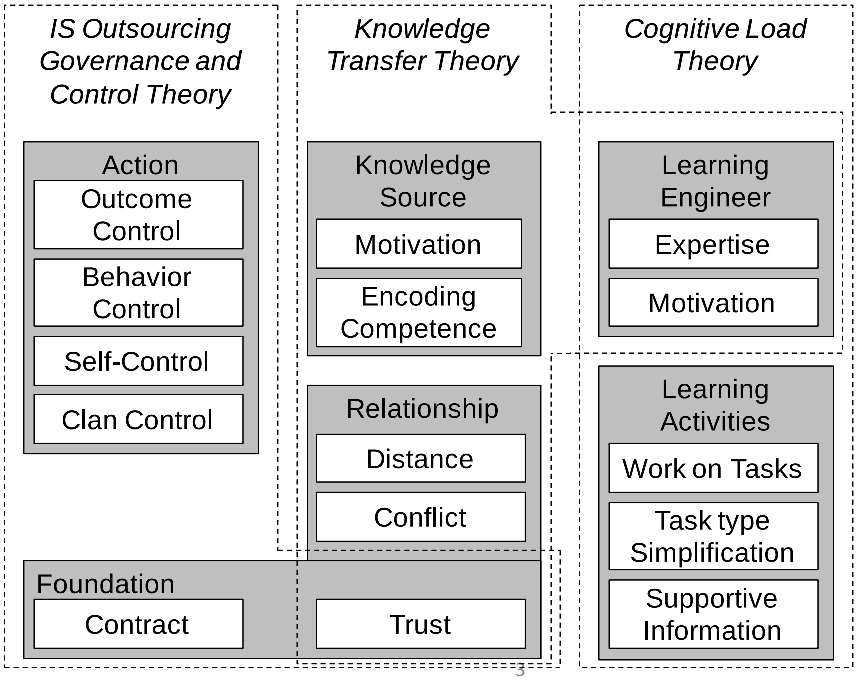

Figure III-1: Conceptual Framework

2.1 Outsourcing Governance and Control

The IS outsourcing governance literature aims to explain how clients influence behavior in outsourcing projects through governance (Goo et al. 2009). A refined conceptualization (Huber et al. 2011) suggests that governance comprises the foundations and actions that guide behavior towards desired objectives. Prominent foundations of governance include the contract and relationship attributes such as trust (Huber et al. 2011). These foundations enable actions of governance, to which we refer as control (Huber et al. 2011). Control denotes actions intended to align individual behavior with organizational objectives (Kirsch 1996). For instance, a contract may prescribe that vendor personnel must be able to independently solve particular software-maintenance problems at the end of the transition phase (foundation). This enables client management to measure, evaluate, and reward or sanction vendor performance (control).

Control theory has been established in the IS literature as a framework to describe and predict control (Kirsch 1996). Control theory distinguishes four modes of control: outcome control, behavior control, clan control, and self-control (see Appendix III-1 for our adapted definitions). Whereas outcome control and behavior control are instances of formal controls, clan control and self-control have been referred to as informal controls. Control may not only be described with regard to its mode, but also with regard to its amount. The amount of control has been defined as the variety of mechanisms and the extent to which each of the mechanisms is used (Rustagi et al. 2008). Findings from control theory include that control portfolios are chosen based on task characteristics, the controller’s knowledge, and relationship characteristics (Choudhury and Sabherwal 2003; Kirsch 1996; Kirsch 2004; Rustagi et al. 2008) and that control portfolios may change when these contextual factors change (Choudhury and Sabherwal 2003; Kirsch 2004).

While control theory has been used to study how control influences software delivery, we adopt control theory to study how control influences learning. In this context, outcome control includes actions targeted at enforcing specific learning outcomes such as levels of understanding or task performance standards. Behavior control may refer to the enforcement of knowledge transfer procedures such as the compulsory use of replay sessions (Gregory et al. 2009). Similarly, clan control and self-control can be used to informally steer behavior in a way that aligns it with the learning goals desired by the client.

2.2 Cognitive Load Theory

Cognitive load theory (CLT) (Sweller et al. 1998; Van Merriënboer and Sweller 2005) is currently one of the most influential theories in educational psychology (Ozcinar 2009; Schnotz and Kürschner 2007). It is positioned as a theory to explain the learning of rather complex tasks (Van Merriënboer and Sweller 2005), such as software maintenance (Banker et al. 1998; Von Mayrhauser and Vans 1995), in settings that impose heavy cognitive load on the learner, such as in the transition phase of offshoring projects (Chua and Pan 2008). Recent research suggests that CLT may well explain learning outcomes in SMOO (Krancher and Dibbern 2012, see also study 1). In the context of this study, CLT can therefore be a useful lens to identify those behaviors that can be expected to result in learning. This may subsequently help explore how governance may enforce these behaviors.

According to CLT, learning is an increase in expertise through the acquisition or automation of schemas (Van Merriënboer and Sweller 2005). Cognitive load theorists concur that meaningful learning occurs when the learner engages in authentic tasks as long as the cognitive load imposed by the task is manageable for the learner and the learner is motivated to engage in schema construction (Schnotz and Kürschner 2007; Van Merriënboer et al. 2003). When the complexity of a task exceeds the processing capacity indicated by the learner’s expertise, strategies to reduce cognitive load are necessary. These strategies include supportive information and task type simplification (Van Merriënboer et al. 2003).

Supportive-information activities such as informal explanations, formal presentations, and document study provide blue-prints for schemas on non-recurrent aspects of the task domain. Task type simplification reduces the problem-solving search processes, e.g. by partially or fully indicating the solution to a problem. For instance, an SME may compile the design of a modification request and leave only its implementation to the vendor engineer. These strategies have been not only suggested by literature in education psychology, but also reported to be used in SMOO projects (Krancher and Dibbern 2012).

The discussion of CLT suggests that the combination of work on tasks, supportive information, and/or task type simplification results in meaningful learning. Given the objective of this paper, it will be desirable to understand how governance influences the occurrence of these activities and what impact the resulting expertise has on governance.

2.3 Knowledge Transfer Theory

Adding to the perspective provided by CLT, the knowledge transfer literature explains under which conditions knowledge is communicated from a source so that it is learned and applied by a target (Ko et al. 2005). Although this conceptualization does not grasp learning that occurs outside of communication processes, such as during the work on tasks, the knowledge transfer literature has enhanced our understanding of antecedents to knowledge transfer, in particular those that are specific to cross-border and interorganizational knowledge transfer.

The antecedent conditions to knowledge transfer include characteristics of the SME, of the learner, and of their relationship. Relevant characteristics of the SME are their motivation (Ko et al. 2005; Szulanski 1996) and their encoding competence (Ko et al. 2005). Characteristics of the relationship include trust (Chowdhury 2005), distance (Chen and McQueen 2010), and conflict (Hinds and Bailey 2003). The characteristics of the learners include their motivation and their absorptive capacity (their ability to assimilate and apply outside knowledge) (Cohen and Levinthal 1990; Ko et al. 2005; Szulanski 1996). We did not incorporate absorptive capacity as it is strongly related to the construct of expertise in that both denote the ability to relate novel information to former experience.

Taken together, the existing literature provides us with constructs to describe activities that are likely to result in learning (learning activities), conditions under which these activities are likely to occur (antecedents established in the knowledge transfer literature), and foundations and actions through which these activities may be enforced (governance). In this study, we aim to explore how governance and learning activities influence each other over time in the transition phase of SMOO projects. Next, we describe the methods that we applied to this end.

3 METHODS

We chose a longitudinal multiple-case study approach to develop theory (Eisenhardt 1989b). We deemed the case-study approach suitable because we addressed a *how* question in a context in which the boundaries are not clear and of which researchers have no control (Yin 2009).

We adopted an embedded-case study design. A transition of a software-maintenance role to one vendor employee represented one case. We only included cases in which the vendor team consisted of initially one staff. Within a case, we used temporal bracketing (Langley 1999) to divide transitions into phases that show consistent configurations of the constructs of our conceptual framework. Any significant discontinuity of a construct demarked the start of a new phase. Hence, the phases were used as embedded units of analysis "for the exploration and replication of theoretical ideas" (Langley 1999, p. 703). Bracketing has already been used as an analytic technique to explore the evolution of control (Kirsch 2004).

We gathered data on four transition projects. The transitions were conducted on site in Switzerland on the premises of a Swiss bank, which represented the client in all four projects. In all projects, the vendor engineers were planned to take over the roles of on-site coordinators. Hence, our data focused on the transitions from SME to on-site coordinators, whereas subsequent knowledge transfers to offshore teams were outside the scope of our analysis. We believe that this focus is well-grounded because transitions are typically conducted on site (Dibbern et al. 2008), because the learning of on-site coordinators has been reported to be influential for the success of offshore projects (Gregory et al. 2009), and because this setting still reflects many of the knowledge transfer barriers specific to offshore outsourcing such as cultural and semantic distances and conflict. We decided to investigate the knowledge transfer to individuals who independently take over maintenance tasks to distill the role of individual learning. All projects were based on time-and-materials contracts (Gopal et al. 2003). This may be an important boundary condition of our findings because vendor incentives related to knowledge transfer may differ in time-and-materials versus in fixed-price projects. For instance, clients may pay for the time dedicated to initial knowledge transfer in time-and-material settings, whereas knowledge transfer efforts may be included in the lump sum of fixed-price contracts. That is, vendor may frequently be paid additional efforts for knowledge transfer in time-and-material settings, while they will not in fixed-price settings. All projects were considered successful by the stakeholders. Our data do thus not allow us theorizing how governance may relate to success. Instead, it allows us exploring how governance and learning interact with each other in successful transitions.

Data were collected through semi-structured interviews, observation of knowledge transfer sessions, and document analysis based on a case-study protocol (Yin 2009, p. 79). Table III-1 gives an overview of the data sources. The data were collected real-time (Langley 1999) in the cases 1 to 3 and retrospectively in the case 4. To increase the validity of longitudinal data, several interviews with the vendor engineers were conducted at different points in time. All interviews except for one were tape-recorded and transcribed. In addition, the first author observed knowledge transfer sessions. The documents analyzed included contract extracts, knowledge transfer plans, the knowledge transfer methodology used (Ackermann 2011), and software specifications.

The data analysis process is displayed in Figure III-2 and described next. Data were coded to nodes and relationship nodes in NVivo 9. Because the existing literature did not provide rich descriptions of the governance of learning in SMOO projects, data were initially coded to nodes at the construct level. This first coding run was done by two researchers and was guided by definitions of the a-priori constructs. The coders then discussed the disagreements that emerged in five interviews and reasons for disagreement were documented. The reasons for disagreement and the results of the first coding run allowed inductively defining most of the coding rules, while some rules were inspired by the literature (see Appendix III-1). This approach was consistent with Eisenhardt's recommendations for positivist theory-building according to which "measures often emerge from the analysis process itself, rather than being specified a priori." (Eisenhardt 1989b, p. 542)

Table III-1: Data Sources

<table>
<tr><th rowspan="2">Case</th><th colspan="4">Interviews: number of interviews/ number of interviewees</th><th rowspan="2">No. of Observed Sessions</th><th rowspan="2">No. of Documents</th></tr>
<tr><th>Vendor engineers</th><th>SME</th><th colspan="2">Client Managers</th></tr>
<tr><td>1</td><td>5/1</td><td>2/2</td><td>2/1</td><td rowspan="3">1/1</td><td>4</td><td>21</td></tr>
<tr><td>2</td><td>2/1</td><td>2/2</td><td rowspan="2">3/3</td><td>2</td><td>9</td></tr>
<tr><td>3</td><td>3/1</td><td>2/2</td><td>2</td><td>16</td></tr>
<tr><td>4</td><td>2/1</td><td>2/2</td><td colspan="2">1/1</td><td>2</td><td>2</td></tr>
</table>

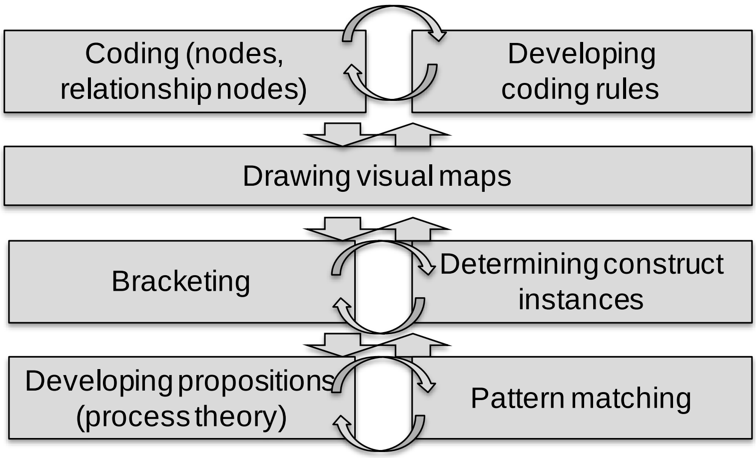

Figure III-2: Data Analysis Process

The first author also coded interview statements that indicated a causal link between the constructs to relationship nodes. Relationship nodes indicate relationships between two nodes. For instance, the following statement was interpreted as suggesting causal links between expertise and self-control and between self-control and supportive information:

> *"He gathered a lot information independently from others. Because he had very vast knowledge on the tool, he knew where to look up what information." (Client manager, case 4)*

Next, the first author drew one visual map (Langley 1999) per case, in which he arranged the events that had been coded to be control or learning activities on a timeline. The resulting displays provided first illustrations of the evolution of control and learning activities over time. He then marked events that suggested changes in all constructs beyond control and learning activities.

Subsequently, the first author determined construct instances such as high, medium, low, while looking for discontinuities in the constructs to divide the transition into phases (bracketing). This iterative process resulted in 15 phases. Out of these 15 phases, three were eliminated due to lack of triangulation. Construct instances were determined by the first author through the coding rules. Examples are provided in the next section.

In a last step, propositions were developed and examined by pattern matching. The propositions have been intended to form a process theory that explains the interaction of governance and learning. Process theories depict events that are assumed necessary for an outcome to occur (Mohr 1982). To this end, we looked for causal interview statements that were able to explain changes in learning activities or governance from one phase to another. Next, we attempted to match the causal argument with the changes that could be observed within cases and with differences between the cases. We terminated data analysis once the process theory could explain the evolution of governance and learning in all cases. Explanation development reduced the a-priori set of constructs to the subset that proved relevant in data analysis (Eisenhardt 1989b). While we developed propositions to explain the occurrence of learning activities, we controlled for a decreasing need of supportive information and task type simplification with increasing expertise (Krancher and Dibbern 2012; Van Merriënboer et al. 2003).

4 CASE ANALYSIS

This section presents the results of the data analysis. First, we illustrate the analysis by providing a narration of case 1 and explaining how we determined some of the construct instances in case 1. Second, we describe and analyze the interactions of governance and learning in all four cases.

4.1 Illustration of Construct Evaluation

Phase 1 of case 1 started when the vendor engineer arrived at the client site. Management commanded that she[5] should participate in knowledge elicitation sessions that were accompanied by a knowledge transfer coach of the client firm (behavior control). This was based on the knowledge transfer methodology (Ackermann 2011) adopted by the client. Phase 2 started in week 3, when no supportive information activities were commanded by client management any more. Instead, the coach assigned the responsibility for arranging further sessions to the vendor engineer (self-control). No formal sessions took place in phase 2, while informal discussions served as the main mechanism for obtaining supportive information. In addition, the vendor engineer was assigned (behavior control) first tasks in phase 2. She spent considerable effort working on these tasks (work on tasks). In phases 1 and 2, the SME experienced high workloads on duties other than knowledge transfer (conflict). In phase 3, the fourth month of her on-site stay, the vendor engineer was not assigned any tasks by management (low behavior control). In her search for learning op-

[5] The reported genders of the study participants may deviate from their actual genders to protect their identities.

portunities, she approached one SME to work on a task together with him (self-control). This helped the SME to deliver the task (low conflict).

Next, we illustrate how we obtained a subset of the construct instances in case 1. Appendix III-1 shows how we applied the coding rules to obtain all construct instances in case 1. Outcome control was evaluated low across phases. Target learning outcomes were not explicit and not related to a timeline:

> *"It will surely take some time […]. My expectation is not that any specific status is reached after a month. It is very difficult to measure this." (Client manager, case 1)*

Behavior control of learning was evaluated medium in phases 1 and 2. In phase 1, management prescribed approaches for providing supportive information, but did not assign any tasks to the vendor engineer. In phase 2, the opposite situation was observed. We consider both prescribing forms of supportive information and assigning tasks as behavior control because managers specify how the learning process is to be organized.

Clan control was considered medium during phases 1 till 3, because the vendor engineer's learning was not discussed in any meetings, but seems to have been influenced by values of the team members:

> *"[Learning-by-doing] is of great importance. […] I think our team shares this perspective." (Client manager, case 1)*

Self-control was considered to increase from medium in phases 1 and 2 to high in phase 3. In phase 1, the vendor engineer asked a moderate amount of questions and replayed her own understandings to a moderate extent during the observed knowledge elicitation sessions, as indicated by the count of statements. The counts were lower in the first of the two sessions. These observations indicate that she took moderate action when goals such as understanding were not met. Furthermore, she reported to engage for three hours per day in code study, although she later conceded that this activity was not effective for learning:

> *"As such, just code study is pretty tough. Especially here where implementation is quite complex. It's tough." (Vendor engineer, case 1)*

This suggests that the engineer did not trigger significant improvements of learning strategies at that stage. Taken together, the observations suggested medium self-control. In phase 2, the vendor engineer frequently engaged in informal discussions to clarify queries.

However, although the vendor engineer was assigned the responsibility of arranging formal sessions and although she indicated a need for them to the SME, they did not take place in phase 2. We therefore considered self-control still medium in phase 2. In contrast, self-control was high in phase 3, when the vendor engineer independently approached a SME to become involved into a task:

> *"Actually, it is not my task, but I was with [SME 2] when he was developing one of his tasks. – Was that to learn or to help him? – It was for learning and a little bit of help." (Vendor engineer, case 1)*

Finally, we illustrate how we evaluated conflict. Conflict was considered medium in phases 1 and 2, and low in phase 3. In phases 1 and 2, SME 1 reported high work load through tasks other than knowledge transfer:

> *"Has the overall-picture session already taken place? – No, not yet, I am currently a little bit under stress, but that will improve." (SME 1, case 1)*

On the other hand, the SME reported benefits from the vendor engineer's learning, which overall suggests a medium level of conflict:

> *"It was good that [the vendor engineer] freed me of ties by working on these small things." (SME 2, case 1)*

Conflict was less in phase 3, when the involvement of the vendor engineer in the task of SME 2 resulted both in learning for the vendor engineer and in help for the SME to get his task accomplished.

Construct evaluation was paralleled by the subdivision of the transitions into phases based on discontinuities between the phases (bracketing). From phase 1 to phase 2, we observed changes in the mechanisms of behavior control, in self-control, and in work on task. Conversely, phase 3 is distinguished from phase 2 by a decrease in behavior control, because the vendor engineer was assigned no tasks, by a decrease in conflict, because the SME now benefited from providing the vendor engineer with learning opportunities, and by an increase in self-control.

4.2 Results and Discussion of Cases

Table III-2 shows the construct instances over phases. For reasons of parsimony, behavior and outcome control have been grouped to formal control. Because the table provides an aggregated overview of the cases, it does not reveal the multifaceted details of the qualitative data, of which we made use during data analysis.

Next, we describe and analyze the evolution of governance and learning over time in the four cases. Because all cases indicated similar configurations of control towards the end of the transition, we start our discussion with the end states and the events that may have led to these end states.

Across cases, we find substantial levels of *self-control* towards the ends of the transitions, while there is more variation in the initial levels of self-control. Causal interview statements suggest that trust and expertise may explain the increases of self-control. For instance, the following statement indicates the impact of trust on self-control from the perspective of the vendor engineer:

> *"I always do ask a lot of questions. But it is always better to share a good rapport with the other person. After that, I feel much more comfortable in asking my questions, and have the confidence that my doubt will be clarified. Also, I am aware that the question will not be turned down." (Vendor engineer, case 1)*

Another statement may illustrate how trust and expertise enable self-control from the perspective of client management:

> *"He is […] very strong in asking questions. He is active and he has a very good knowledge. This is why we did not have to pay a lot of attention to his knowledge acquisition. We had the confidence that he will be able to do it." (Client manager, case 1)*

Different levels of expertise in the cases 3 and 4 seemed to enable different amounts of self-control. In case 4, high expertise from former projects involving the software package facilitated self-control:

> *"[The vendor engineer] gathered a lot of information independently from others. Because he had very vast knowledge on the tool, he knew where to look up what information." (Client manager, case 4)*

Table III-2: Construct Instances

Case			Case 1			Case 2			Case 3		Case 4			
Phase			1	2	3	1	2	3	1	2	1	2	3	4
Case context		Application	Data warehousing system 1			Data warehousing system 2			Data warehousing system 3		Financial transactions control system			
		SME	Client staff			Other vendor, client staff			Other vendor, client. staff		Client staff			
Learning activities		Work on tasks	- -	++	++	0	0	++	++	++	0	0	+	++
		Supp. Inform.	++	0	0	+	-	-	+	0	++	-	0	0
		Specification	M	+	0	+	+	0	+	0	0	0	0	0
(1) Control		Formal control	-	-	- -	0	-	-	+	++	-	- -	-	-
		Clan control	0	0	0	++	++	0	++	++	- -	- -	-	-
		Self-control	0	0	++	+	+	+	+	++	+	+	++	++
(2) Found. of self-control		Expertise	-	0	+	0	+	+	0	0	+	++	++	++
(2) Found. of self-control	*(3) Barriers to knowledge sharing*	Trust	- -	0	++	M	+	+	0	+	-		0	++
	(3) Barriers to knowledge sharing	SME Motivation	0			-		M	+		-		0	
		Distance	0			- -		0	- -		+			
		Conflict	0	0	- -	+	+	0	0	-	+	++	++	0
(1) Overall control			-	-	-	+	0	0	+	++	-	-	0	0
(2) Overall foundations of self-control			- -	0	++	0	+	+	0	+	0	+	+	++
(3) Overall barriers to knowledge sharing			+	0	-	0	0	0	-	-	+	+	+	-

(- -: low, -: medium-low, 0: medium, +: medium-high, ++: high, M: missing information)

Conversely, the relatively low expertise of the vendor engineer in case 3 constrained his ability to effectively self-control learning activities:

> *"You go to this document. Immediately, you do not understand anything. […] Then I asked [SME 1]: 'What is this'? Then he said: 'You do not understand this now, you need to read first these 3 documents.' It could have been easier if he had said: 'You just read this first.'" (Vendor engineer, case 3)*

A refined conceptualization of governance (Huber et al. 2011) has proposed the term *foundation* to describe the basis of governance that enables control. Our analysis suggests that trust and expertise may constitute foundations of self-control of individual learning. Once these foundations are established, vendor engineers may engage in high amounts of self-control:

P1: High amounts of self-control require significant expertise and trust (foundations of self-control).

The central role of self-control towards the ends of transitions raises the question of how the foundations of self-control—trust and expertise—developed. Our data suggest that learning activities were the main antecedents to changes in these foundations. Many interview statements suggest a causal link between learning activities and increases in expertise (see Krancher and Dibbern 2012 for more examples) such as

> *"He is now able to do what he has previously done. That is what he is able to do." (SME 2, case 1)*

Likewise, we find statements explaining how learning activities increase trust such as:

> *"You slowly noticed that you can give him tasks after you saw that he well accomplished the first task in the project with [SME 2] and that it did not consume much of his time. Suddenly you noticed, there is someone to whom you can assign responsibility." (Client manager, case 4)*

We therefore propose:

P2: Expertise and trust develop through learning activities.

If learning activities play a central role for the development of trust and expertise, there is interest in understanding the antecedents to learning activities. While we observed relatively high amounts of learning activities towards the ends of all transitions, there was more variation at their beginnings. Our evidence indicates that high amounts of learning activities initially only manifested under rather high levels of formal and clan controls. In case 1, significant supportive information was given only through the knowledge elicitation sessions in phase 1, when the process was under the control of the knowledge transfer coach (behavior control). In contrast, supportive information activities were less in phase 2, when the responsibility for arranging formal sessions was assigned to the vendor engineer (low behavior control regarding supportive information). Not only supportive information, but also work on tasks seemed to depend on formal control in case 1. Work on tasks started only in phase 2, when the vendor engineer was formally assigned a change request (behavior control). Likewise, the vendor engineer of case 4 only had the opportunity to work on new tasks once management increased the behavior control on the SME to delegate tasks. All these discontinuities are increases of formal control that are associated with higher amounts of learning activities. In case 3, formal and clan controls were relatively high from the beginning and so was the amount of learning activities, confirming thereby a strong link between control and the amount of learning activities.

Yet, the amount of control needed for significant learning activities varied within and across cases. Our data suggest conflict, distance, lack of motivation of SME, and lack of trust as barriers that needed to be overcome by control. The higher these barriers were, the less learning activities took place under unaltered control. For instance, knowledge elicitation sessions in case 3 only took place once delivery activities for a release (a source of conflict) were completed. The decrease in conflict led obviously to an increase in learning activities while control remained unchanged. In case 4, similar amounts of control led to more learning activities in phase 4 than in phase 3 after trust increased, while cultural distance was still present:

> *"I appreciate him a lot, as a colleague and as a human. He showed me that the wall between the cultures is not that big. The wall is broken down." (SME 1, case 4)*

> *"Suddenly we noticed that we could give certain things to [the vendor engineer]: 'Can you write this script for me?'" (SME 1, case 4)*

The preceding discussion suggests that high amounts of learning activities only take place under significant amount of control. The required amount of control seems to be a function of the barriers to knowledge sharing. We propose therefore:

P3: High amounts of learning activities manifest only when control outweighs the barriers to knowledge sharing (conflict, distance, lack of motivation of SME, lack of trust)

Taken together, the propositions form a process model that explains the evolution of governance and learning over time. The model is depicted in Figure III-3.

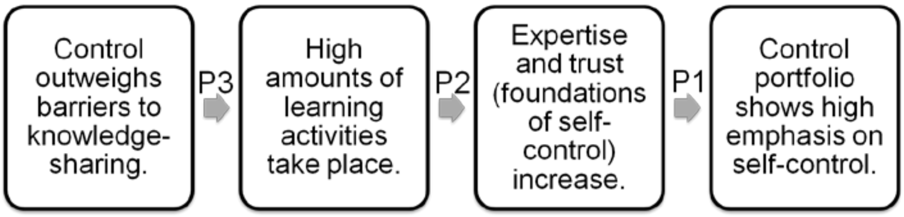

Figure III-3: Process Model of the Interaction of Governance and Learning

5 IMPLICATIONS

Although the existing literature suggests that knowledge transfer and its governance are central to the success of SMOO projects, we knew little on how knowledge transfer and governance interact. In this paper, we focused particularly on the interaction of governance and individual learning during the transition phase of SMOO projects. Based on in-depth data from four transitions, we developed three propositions of a process model of the interaction of governance and learning in SMOO transitions.

Our paper contributes to the existing literature in various ways. First, we suggest explanations for how and why governance portfolios may change in SMOO transitions. In particular, we propose that shifts in the amounts of self-control may be a major driving force in the evolution of control. In our data, control portfolios were initially characterized by relatively weak self-control components. Our analysis indicates two reasons for initially weak self-control: a lack of the vendor engineer’s expertise and a lack of trust.

Vendor engineers may frequently have low prior experience in the task domain at the beginning of transitions (see also study 1), which translates into low expertise. Low expertise seemed to constrain their abilities to self-control their learning. Under low expertise, vendor engineers struggled to ask the right questions, to identify appropriate knowledge sources, and to diagnose and remedy weaknesses in their learning approaches. An explanation for this finding may be sought within the cognitive-load framework. Self-control demands cognitive resources (Baumeister et al. 1998). Self-control thus competes for mental resources with those processes that mentally organize novel information. Given the frequently low expertise of vendor engineers, mentally organizing information consumes a substantial amount of resources or may not even by possible with the available mental resources (see also the studies 1 and 2). As a consequence, few resources may remain for self-control. Low-expertise engineers thus face a dilemma. They are least able to self-control their learning when they would benefit most from actions that could improve their learning process such as triggering task type simplification or identifying supportive information. Educational research on the impact of domain knowledge on self-regulated learning (Moos and Azevedo 2008) also lends support for the proposition that a lack of expertise constrains self-regulation.

Not only a lack of expertise, but also a lack of trust seems to have constrained self-control at the outset of projects. Vendor engineers may not dare to address their questions to the SME or to open themselves learning opportunities by taking over work assigned to the SME. Likewise, the SME may be hesitant to delegate work to vendor engineers when they lack the trust that the vendor engineer will deliver the work. Attributional theory (Lepine and Van Dyne 2001; Weiner 1985) may help explain this. It suggests that coworkers (such as SME) will make decisions on helping behavior based on ability attributions. When they face trust in the ability of a low-performing coworker, they will engage in helping behavior because they anticipate that their help may soon result in stronger performance. Conversely, when they lack trust in the ability of their coworker, they cease helping because they do not expect any change in performance. The SME may thus initially be hesitant to engage in substantial helping when they lack evidence of the ability of the vendor engineer. Vendor engineers may anticipate these judgments. They may refrain from help-seeking when they fear that their questions may result in negative ability attributions. Conversely, once trustful relationships have been established, the vendor engineers may be aware of the trust in their abilities and may thus seek help or proactively take over tasks.

Because expertise and trust constrain self-control by the vendor engineers at the outsets of projects, client management may initially need to engage in formal and clan controls that outweigh barriers to knowledge sharing. The barriers may include conflict, distance, lack of motivation of the SME, and lack of trust. The higher these barriers are, the more control may be needed. Only when control exceeds the level imposed by the barriers will the vendor engineers engage in high amounts of learning activities. Over time, these learning activities will result in increased expertise and trust and thus allow more self-control. As a result, formal and clan controls may be reduced over time and governance portfolios may have strong emphases on self-control towards the end of successful transitions.

Second, to the best of our knowledge, our study is the first to apply control theory to learning. Our results suggest that control theory may be used to explain the occurrence of learning activities in organizational settings. The coding rules developed in this study to grasp the control of learning may be a connecting point for future research.

Third, our paper confirms that conceptualizing governance as foundations of governance and control helps understand how governance evolves over time. Our paper extends this perspective by proposing expertise and trust as foundations of self-control. The results thereby also suggest an extension to control theory, at least if the task to be controlled is knowledge transfer. Control theory posits that the controller's knowledge enables behavior control (Kirsch 1996). This argument may similarly apply to self-control. Because the controllee is controller at the same time in the case of self-control, she/he may require knowledge to effectively exert control of his own behavior.

Fourth, the prior literature (Chowdhury 2005; Ko et al. 2005; Szulanski 1996) has shown a strong link between trust and knowledge sharing. Our study suggests that these findings do apply to learning in SMOO projects. Moreover, our paper explains that trust may be essential for knowledge transfer because it plays a dual role. On the one hand, lack of trust is a barrier to knowledge sharing by SME. On the other hand, trust seems to foster self-control.

Our study has several limitations. First, although two coders were involved during early data analysis, the final coding was done by one coder only. We attempted to mitigate negative effects of this by specifying detailed rules for the evaluation of constructs. Second, we investigated only rather successful transitions. It is possible that control and learning interact in ways other than anticipated here in less successful projects. Third, our study is limited to individual learning by vendor on-site coordinators. While this scenario is typical for transitions in SMOO projects, our findings may not necessarily apply to transitions to teams that are placed offshore. Fourth, all projects were based on time-and-materials contracts. Governance may play different roles in fixed-price contracts, in which the vendor may not be paid for knowledge transfer activities. Fifth, all projects referred to the same client and were of small scale. Sixth, we did not report rival explanations.

Future research may connect to this study. Researchers may repeat the study under different conditions such as in fixed-price or large-scale projects involving offshore teams. Other extensions of our study are to build a variance theory or to test the process theory. Finally, our study has highlighted that self-control may play an influential role in individual learning. This suggests that theories of self-regulated learning (Pintrich 2004) may serve as a powerful lens to understand learning in the transition phase of SMOO projects. Evidence from school settings suggests that national cultures may explain differences in strategies of self-regulated learning (Purdie and Hattie 1996). Future research may test whether such differences also manifest in the self-regulated learning of software engineers.

APPENDIX III-1: CONSTRUCT DEFINITIONS, CODING RULES, AND CONSTRUCT INSTANCES IN CASE 1

Definition	Measures	*Phases of Case 1* 1	2	3
Work on Tasks		- -	++	++
The extent to which the vendor engineer was engaged in a realistic software-maintenance task (Van Merriënboer et al. 2003)	The vendor engineer was engaged in realistic software-maintenance tasks.	- -	+ +	+ +
Supportive Information		++	*0*	*0*
The extent to which the learner consults information on non-recurrent aspects of the task (Van Merriënboer et al. 2003)	The vendor engineer was strongly engaged in supportive-information activities such as informal discussions, document study, formal presentations, knowledge elicitation sessions, Google search.	++	0	0
Task Type Simplification		*M*	+	*0*
The extent to which the solutions to the tasks were given to the engineer or to which the goal state of the tasks were relaxed (adapted from Van Merriënboer et al. 2003)	Code to ++ when all tasks were worked examples, to + when some of the tasks were worked examples, to 0 when all tasks were completion tasks, imitation tasks, or goal-free tasks, to - when some of the tasks were conventional tasks, to - - when all tasks were conventional tasks (see section 2 in study 1 for an overview of task types)		+	0
Outcome Control		- -	- -	- -
The amount of measurement, evaluation, and rewarding or sanctioning of behavior against articulated learning outcomes (adapted from Kirsch 1996; Kirsch 2004)	The vendor engineer reported to be aware of a timeline for the completion of knowledge transfer.	- -	- -	- -
	Client management indicated target learning outcomes to be reached at the end of transition.	- -	- -	- -
	Client management indicated intermediate learning outcomes.	- -	- -	- -
	The software products created by the vendor engineer were tested or reviewed.	- -	++	++
	Learning outcomes were measured regularly beyond the results of software tests or reviews.	- -	- -	- -
	Client management regularly evaluated learning outcomes against targets.	- -	- -	- -
	The vendor was rewarded or sanctioned depending on learning outcomes.	- -	- -	- -

Behavior Control		*0*	*0*	- -
The amount of measurement, evaluation, and rewarding or sanctioning of behavior against articulated procedures of behavior that results in learning (adapted from Kirsch 1996; Kirsch 2004)	Client management prescribed that supportive-information activities such as formal presentations, informal discussions and document study should take place.	++	- -	- -
	Client management influenced how supportive-information activities were conducted.	++	- -	- -
	Client management prescribed that the vendor engineer should work on tasks.	- -	++	- -
	Client management or SME considered task complexity and the vendor engineer's expertise when assigning tasks.	- -	++	- -
	Client management prescribed the degree of task type simplification (e.g. design taken over by SME).	- -	- -	- -
Clan Control		*0*	*0*	*0*
The amount of social mechanisms to control the behavior of individuals that results in learning (adapted from Kirsch 1996)	The vendor engineer was part of regular meetings in which learning was informally discussed.	- -	- -	- -
	The SME reported values that indicate the importance of the vendor engineer's learning to them.	++	++	++
	Client management reported their appreciation towards any approach for conducting knowledge transfer.	++	++	++
Self-control		*0*	*0*	++
The amount of actions in which the vendor engineer sets own learning goals, monitors learning, and rewards or sanctions herself/himself accordingly (adapted from Kirsch 1996) (Some measures were taken from the literature on self-regulated learning (Pintrich 2004).)	Client management expressed that they expected self-control.	++	++	++
	Client management or SME reported that the vendor engineer took control of his learning process.		++	++
	The vendor engineer reported target learning outcomes.	++	++	++
	The vendor engineer actively approached the SME whenever she/he had doubts that could not be answered based on documents.	0	0	++
	The vendor engineer effectively consulted documents.			
	The vendor engineer insisted to take over tasks.	- -	- -	++
	The vendor engineer reported reflections on the effectiveness of own learning strategies.			++
	The vendor engineer triggered improvements based on reflections of learning effectiveness.	- -	- -	++
Expertise		-	*0*	+
The power of the vendor engineer's schemas, i.e. memory structures that categorize information in a manner specific to perform a particular task (Sweller et al. 1998) (measured based on the categories of the body of knowledge (Iivari et al. 2004))	The vendor engineer demonstrated strong expertise in task-specific application knowledge.	-	-	0
	The vendor engineer demonstrated strong expertise in task-specific application domain knowledge.	-	-	0
	The vendor engineer demonstrated strong expertise in task-specific IS development process knowledge.	-	-	0
	The vendor engineer demonstrated strong expertise in task-specific technical knowledge.	+	+	++
	The vendor engineer demonstrated strong expertise in task-specific organizational knowledge.	-	0	+

Trust		- -	*0*	+ +
The reciprocal willingness of the vendor engineer and the SME to depend on another person's actions that involve opportunism (Chowdhury 2005; Mayer et al. 1995)	The vendor engineer reported to expect friendly answers when approaching the SME with questions.		0	++
	The SME expressed appreciation of the personal traits of the vendor engineer.			++
	The relationship between the vendor engineer and the SME is cordial.	- -		++
	The SME reported that the vendor engineer possessed strong skills from prior experience.			++
	The vendor engineer reported that (s)he perceived the SME as competent.			++
SME Motivation		*0*		
The willingness of the SME(s) to provide the vendor engineer with learning opportunities	Client management reported that the SME was willing to create learning opportunities for the vendor engineer.			
	The vendor engineer reported that the SME was willing to create learning opportunities for the vendor engineer.	0		
	The SME reported to enjoy rather than to detest sharing knowledge.	0		
Distance		*0*		
The cultural, semantic, and geographic distances between SME and learner (Dibbern et al. 2008)	The SME and the vendor engineer come from countries with different national cultures.	++		
	The SME reported that cultural differences strongly affected the interactions.	- -		
	The SME reported that language barriers strongly affected the interactions.	0		
	The vendor engineer reported that language barriers strongly affected the interactions with the vendor engineer.	0		
	The observation notes indicate that the vendor engineer or the SME struggle with English.	0		
	The vendor engineer and the SME worked in the same room. (inverted)	++		
Conflict		*0*	*0*	- -
The degree of incompatibility of activities, resource share, and goals between partners (Lee and Kim 1999)	The SME had high work load of tasks other than knowledge transfer.	++	++	
	Providing the vendor engineer with learning opportunities helped the SME get own duties done. (inverted)	- -	- -	++
	Increasing the productivity of the vendor engineer lies in the SME's interest. (inverted)	++	++	++

(Empty cell: no basis, - -: low, -: medium-low, 0: medium, +: medium-high, ++: high)

STUDY 4

MANAGEMENT DECISIONS IN SOFTWARE-MAINTENANCE OFFSHORING TRANISITIONS: INSIGHTS FROM A DYNAMIC MODEL

ABSTRACT

The existing literature suggests that transitions in software-maintenance offshore outsourcing projects are prone to knowledge transfer blockades, i.e. situations in which the activities that would yield effective knowledge transfer do not occur, and that client management involvement is central to overcome them. However, the theoretical understanding of the knowledge transfer blockade is limited, and the reactive management behavior reported in case studies suggests that practitioners may frequently be astonished by the dynamics that may give rise to the blockade. Drawing on recent research from offshore sourcing and reference theories, this study proposes a system dynamics framework that may explain why knowledge transfer blockades emerge and how and why client management can overcome the blockade. The results from Monte Carlo simulations based on the system dynamics framework suggest that blockades may emerge from a vicious circle of weak learning due to cognitive overload of vendor staff and negative ability attributions that result in decreased helping behavior and thus aggravate cognitive load. Client management may avoid these vicious circles by selecting vendor staff with strong prior related experience. Longer phases of coexistence of vendor staff and subject matter experts may only help to the extent that they are accompanied by initially high formal and clan controls.

1 INTRODUCTION

Businesses continue to outsource software-maintenance work to offshore vendors given scarce domestic personnel and labor cost advantages in countries such as India (Oshri et al. 2011). Yet, many software-maintenance offshore outsourcing (SMOO) endeavors do not meet the initial expectations (Booth 2013; Dibbern et al. 2008; Gregory et al. 2009; Wende and Philip 2011). Ineffective knowledge transfer to vendor staff is a frequent source of failure, in particular when vast amounts of client-specific knowledge need to be transferred (Chua and Pan 2008; Dibbern et al. 2008; Oshri et al. 2011; Westner and Strahringer 2010). Knowledge transfer may be defined as the process (Szulanski 2000) through which vendor engineers acquire the task knowledge (i.e. the knowledge required to perform the software-maintenance tasks).

Knowledge transfer is particularly salient and critical during the transition phase (Chua and Pan 2008; Dibbern et al. 2008). This phase succeeds the contract signing and ends when the vendor team is able to take over delivery (Tiwari 2009). Vendors frequently send engineers to the client site during transition to help them acquire knowledge on the client's software applications, business processes, organizational structure, software-maintenance processes, and client-specific technologies (Chua and Pan 2008; Dibbern et al. 2008). They acquire knowledge by interacting with the client's subject matter experts (SME), by studying documents and software, and by working on software-maintenance tasks (Gregory et al. 2009; Krancher and Dibbern 2012; Nicholson and Sahay 2004). Yet, this learning process is frequently problematic. Case studies report that the engineers of the offshore unit were cognitively overloaded by the amount of client-specific information to be learned (Chua and Pan 2008; Krancher and Dibbern 2012). As a result of problematic knowledge transfer, the offshore teams may not be able to fully take over the tasks at the planned end of transition (Chua and Pan 2008), which may yield extra costs for knowledge transfer, specification, coordination, and control that offset the savings through labor cost advantages (Dibbern et al. 2008). Software-maintenance tasks may be particularly prone to problematic knowledge transfer because of the high cognitive demands that maintenance imposes on the individual engineer (Pennington 1987; Von Mayrhauser and Vans 1995) and because of the central role of domain-specific experience for maintenance productivity (Boh et al. 2007). Effectively managing knowledge transfer during transition may thus greatly contribute to the success of SMOO projects.

Recent research has improved our theoretical understanding of the mechanisms that may operate during knowledge transfer in SMOO transitions. Drawing on in-depth longitudinal data from five SMOO projects, Krancher and Dibbern (2012) found that the knowledge acquisition by vendor engineers could be well predicted by cognitive load theory (CLT) (Sweller et al. 1998; Van Merriënboer and Sweller 2005). In this perspective, the high cognitive demands that are frequently imposed on the engineers of the offshore unit (Chua and Pan 2008; Dibbern et al. 2008) impair not only their task performance, but also their learning, i.e. their improvement of task performance over time. Stably high cognitive load (i.e. cognitive demands that the tasks impose on vendor engineers) may thus result in continuously low task performance unless remedies against high cognitive are provided. Two broad strategies may be effective remedies to this end: (1) help by the SME and (2) simple-to-complex sequencing of the tasks assigned to the vendor engineer (Krancher and Dibbern 2012). While these strategies may mitigate the cognitive loads on vendor engineers and thereby improve their learning, they may not materialize without management involvement given the high barriers imposed by cultural differences, language barriers, low familiarity with the SME, little prior related experience, and conflict (Dibbern et al. 2008; Gregory et al. 2009; Krancher and Slaughter 2013). Under these circumstances, vendor engineers may not be able to and may not dare to consult appropriate help or enforce simple-to-complex sequencing at the outset of projects when their expertise in the domains of the task is low and their relationships with SME lack trust (Krancher and Slaughter 2013). These findings suggest that a complex set of dynamic interactions operates. Improvements of knowledge may depend on the use of strategies that, in turn, may be contingent on knowledge and knowledge-related outcomes such as trust (Mayer et al. 1995). Client management has been suggested to play a important role in breaking knowledge transfer blockades, i.e. situations in which the activities that would yield effective knowledge transfer do not occur (Gregory et al. 2009; Krancher and Slaughter 2013).

Although it may lie in the hands of client managers to overcome adverse dynamics in SMOO projects, it is not clear how their involvement may shape these dynamics. Client management may make at least three decisions that can affect the dynamics during transition. First, client management may or may not engage in vendor personnel selection to control the prior related experience of the staff assigned to the project (Dibbern et al. 2008; Gregory et al. 2009). Second, managers choose the amount of organizational controls related to knowledge transfer (Gregory et al. 2009; Krancher and Slaughter 2013). Evidence suggests that management may frequently fail to anticipate the need for controls, trusting in the self-control by the vendor (Dibbern et al. 2008; Gregory et al. 2009). Third, client management may decide on the duration of coexistence, the period during which the SME are assigned to the project to provide help to the vendor engineers. Whereas short coexistence periods risk prematurely interrupting knowledge transfer, long coexistence durations may jeopardize the business case of offshoring given the high cost rates of experts. Client management may frequently underestimate the need for coexistence and face unexpected costs for the SME involvement as a consequence (Dibbern et al. 2008). The reactive mode in which client management made and revised these decisions in the cases reported in the literature suggests that there is a practical need to understand how management decisions affect the dynamics in transitions. The study at hand aims at fulfilling this need by addressing the following question:

> *How do the client management decisions (1) on the involvement in staff selection, (2) on organizational controls, and (3) on coexistence duration impact transition outcomes in SMOO?*

The question is addressed adopting the system dynamics paradigm (Forrester 1961). This paradigm helps explore how the components in complex systems interact over time (Forrester 1961). The paper takes a two-steps approach to this end. In section 2, a dynamic model of learning and support during SMOO transitions is developed based on research on SMOO and reference theories. This model is subsequently used to explore how client management decisions may impact the outcomes from the dynamics in the system by means of a Monte Carlo simulation. Section 3 describes the simulation methods before the results are provided and discussed in the sections 4 and 5.

2 A DYNAMIC MODEL OF LEARNING AND SUPPORT IN SMOO TRANSITIONS

In this section, a dynamic model of learning and support during SMOO transitions is developed from the literature. After a brief introduction into the system dynamics paradigm, three increasingly complex systems are modeled to explore and illustrate the dynamics in SMOO transitions.

2.1 The System Dynamics Paradigm

The system dynamics paradigm helps explore the dynamics that operate in complex systems by means of formal simulation (Forrester 1961). It assumes that the behavior of complex systems is frequently difficult to grasp not because exogenous forces are unknown, but because the interactions of the causal mechanisms endogenous to the system are too complex to be accessible to human intuition or cross-sectional data analysis (Forrester 1987; see also Van de Ven and Poole 1995 for similar claims, which, however, are not related to the system dynamics paradigm). In this view, interventions to a system by management or other stakeholders may often not result in the expected outcomes because the feedback mechanisms in complex systems are not taken into account (Forrester 1987). The system dynamics paradigm therefore has a twofold purpose (Forrester 1987; Sawicka 2008). First, it is a tool for analysis. It helps determine how a complex system of related components behaves under given assumptions. Second, it is a tool for communication. It supports visualizing system dynamics by stock-and-flow models and by diagrams of the evolution of system components.

In this study, the system dynamics paradigm is used to explore and illustrate the dynamics of learning and support in SMOO transitions. The system dynamics paradigm is deemed useful to this end because the existing literature has suggested that a complex set of dynamic interactions operates during transition and that management may frequently fail to anticipate these interactions. Prior studies have applied the paradigm to study dynamics in software projects (Abdel-Hamid and Madnick 1989) and in cognitive-load-based learning (Sawicka 2008).

In the following sections, three increasingly complex models of cognitive-load-based learning are developed. The models aim at predicting the learning process of a vendor engineer who is confronted with a series of learning tasks over a certain time frame (Krancher and Dibbern 2012; Van Merriënboer et al. 2003) and who may or may not benefit from support by the SME. This implies a focus on individual learning, which may be a foundation to group learning processes that may emerge at later stages of the project (Kim 1993; Nonaka 1994), but remain outside the scope of this study. The models thus simplify reality in assuming one vendor engineer and one SME.

2.2 Model 1: A Simple Model of Cognitive-Load-Based Learning

Figure III-4 depicts a simple model of cognitive-load-based learning using the stock-and-flow notation. Table III-3 lists the definitions of the variables in this and the subsequent models. The theoretical arguments behind the model and the stock-and-flow notation are explained next.

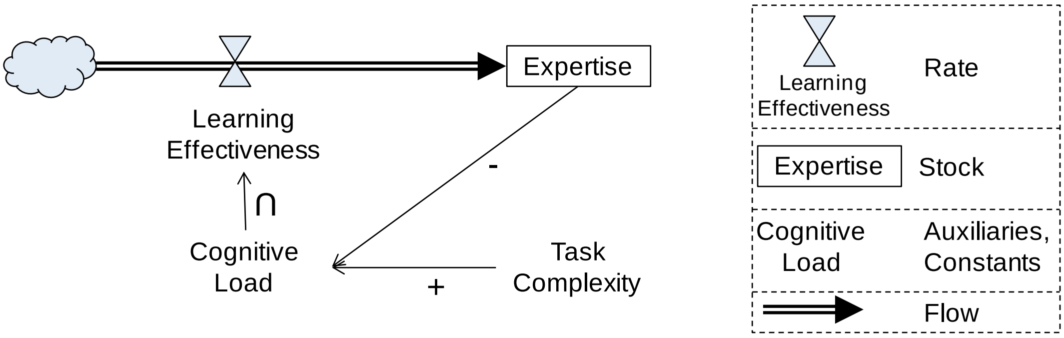

Figure III-4: Model 1: A Simple Model of Cognitive-Load-Based Learning

Table III-3: Definitions of Key Terms

Ability trust: The SME's beliefs in the ability of the vendor engineer (adapted from Mayer et al. 1995; McAllister 1995)
Coexistence: The phase during which both the SME and the vendor engineers are assigned to the project
Cognitive load: The cognitive demands that a maintenance task imposes on the cognitive system of the vendor engineer (Sweller et al. 1998)
Control: The amount and intensity of actions to align the behavior of the SME and the vendor engineer with the client's knowledge transfer goals (adapted from Kirsch 1996)
Expertise: The power of schemas in the vendor engineer's long-term memory that are related to the maintenance task (Sweller et al. 1998)
Help: The amount of social help provided by the SME to the vendor engineer. This includes supportive information and the simplification of task types such as by the use of completion tasks or worked examples (Krancher and Dibbern 2012; Van Merriënboer et al. 2003).
Learning effectiveness: The increase in expertise (Sweller et al. 1998)
Self-control: The extent to which the vendor engineers engages in control related to her/his own learning
Support: Help and simple-to-complex sequencing of tasks
Task complexity: The component, coordinative, and dynamic complexity (Wood 1986) associated with a particular maintenance task such as a change request or a software defect
Transition duration: The duration of the period after which the vendor engineer is able to deliver according to service levels

CLT theory assumes that humans learn by acquiring and refining schemas in their long-term memory (Kalyuga et al. 2003; Sweller et al. 1998; Van Merriënboer and Sweller 2005). Experts hold powerful schemas that enable them to make sense of the world within the domains of their expertise and to solve problems. Conversely, novices lack such schemas and are thus frequently overstrained by novel information and by problem-solving tasks. Learning is thus an increase of expertise. CLT holds that the effectiveness of learning (i.e. the degree to which a setting allows learners to acquire or refine schemas) depends on the cognitive load on the learner. Learning is ineffective when tasks impose too high or too low cognitive loads on learners (Sweller et al. 1998; Van Merriënboer and Sweller

2005). When tasks impose high cognitive loads, the learners are overstrained by the complexity of the task and lack working-memory capacity to generalize from the concrete experience to schemas (Sweller et al. 1998; Van Merriënboer and Sweller 2005). Tasks that impose low cognitive loads do not bear significant learning opportunities because the learners may be largely familiar with the domain of the task (Schnotz and Kürschner 2007). Tasks that impose moderate cognitive load are therefore held to be optimal for learning. The expertise of the learner and the complexity of the task are two main antecedents to cognitive load (Sweller et al. 1998; Van Merriënboer and Sweller 2005). These laws have been found useful to predict how vendor engineers acquire expertise when they work on a series of maintenance tasks during transition (Krancher and Dibbern 2012).

Figure III-4 shows how these predictions can be formalized in a stock-and-flow diagram. The diagram shows expertise as a stock, learning effectiveness as a rate, and cognitive load and task complexity as auxiliaries and constants. Stocks are variables that have a memory (Forrester 1961). Their values may increase or decrease over time based on flows. Conversely, the values of rates and auxiliaries are not a function of their values in prior periods. In model 1, expertise is a stock that may increase over time. Learning effectiveness is the rate that determines how strongly expertise increases. The flow that reflects the increases of expertise departs from a cloud symbol, indicating that there is a potentially endless source of expertise to be acquired as long as the inflow rate *learning effectiveness* permits it. In each period, learning effectiveness depends on cognitive load in an inverted-U-shaped relationship, i.e. moderate cognitive load yields the highest learning effectiveness while high and low loads result in weaker learning. Likewise, the cognitive load at a particular time depends on the expertise at that time and task complexity. Consistent with prior system dynamics research, we assume a scale of 0 (very low) to 1 (very high) for rates, auxiliaries, constants, and the initial values of stocks. Some additional assumptions are needed to mathematically explore the behavior of the dynamic system. These assumptions are described in Appendix III-2.

Figure III-5 shows the behavior of this model if a moderate constant value of task complexity (.5) and a low value of initial expertise (.15) are chosen. The low initial expertise leads to high cognitive load and, consequently, to low learning effectiveness. Due to low learning effectiveness, expertise initially increases only marginally over a sustained period of time. The intuition behind this is that the vendor engineer is overstrained by the demands of the tasks and lacks free mental resources to generalize from his experience. Only after a considerable time of weak learning, expertise reaches levels that provide a significant relief in cognitive load, resulting in higher learning effectiveness. After the maximum of learning effectiveness is reached, further increases in expertise yield cognitive load below the optimal moderate level and, consequently, decreasing learning effectiveness. The resulting expertise curve is S-shaped in contrast to a concave curvilinear relationship frequently found in learning curve research (e.g. Kim et al. 2012; Morrison and Brantner 1992). The S shape emerges because of the period of initially weak learning due to high cognitive load under absence of help. Figure III-6 shows how the system behavior changes if a slightly higher value of initial expertise is chosen (.3). The period of effective learning now starts substantially earlier. In essence, the curves of Figure III-5 are moved to the left, eliminating the tedious initial phase of weak learning. Selecting vendor engineers with prior experience in the domains of the task is a strategy that may take advantage of this effect.

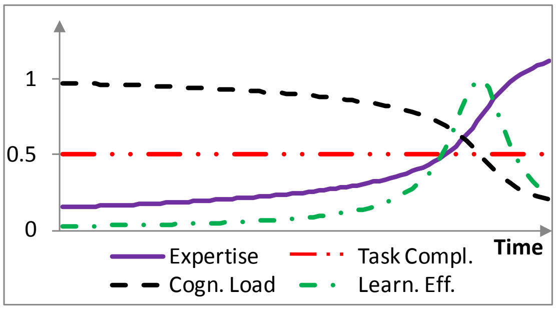

Figure III-5: Model 1 Behavior (initial expertise = .15, task complexity = .5)

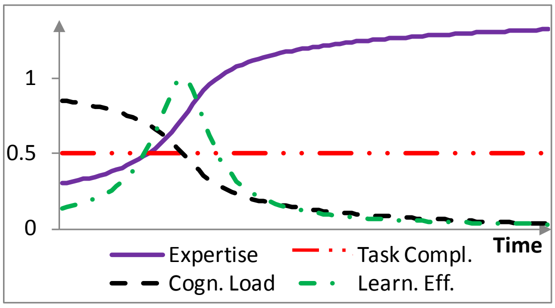

Figure III-6: Model 1 Behavior (initial expertise = .3, task complexity = .5)

2.3 Model 2: A Model of Cognitive-Load-Based Learning and Static Support

Client management may want to avoid the situation illustrated in Figure III-5. Delivery outcomes will be poor when vendor engineers are constantly overloaded by the demands of the tasks, and business departments may not be willing to wait until cognitive load decreases after a long period of weak learning. This may be one major reason for the coexistence of vendor engineers and the SME during initial transitions. The SME may support the learning process of vendor engineers by two broad strategies: simple-to-complex sequencing and help (Krancher and Dibbern 2012; Van Merriënboer et al. 2003). Figure III-7 shows model 2, which extends model 1 by including static support[6] through simple-to-complex sequencing and help. The model is explained next.

Model 2 assumes that task complexity is not constant, but is adjusted based on the simple-to-complex sequencing principle (Van Merriënboer et al. 2003). Client management may purposefully assign simpler maintenance requests at the beginning of transitions or more complex tasks towards the end of the transition to the vendor engineer in order to avoid too high or too low cognitive loads (Krancher and Dibbern 2012). Such behavior is reflected in model 2. A client manager may anticipate the cognitive load before simple-to-complex sequencing (i.e. the cognitive load in absence of simple-to-complex sequencing) based on the task complexity before simple-to-complex sequencing (i.e. the task complexity in absence of simple-to-complex sequencing) and expertise. He may therefore adjust the task complexity accordingly (task complexity after simple-to-complex sequencing) by assigning a different task.

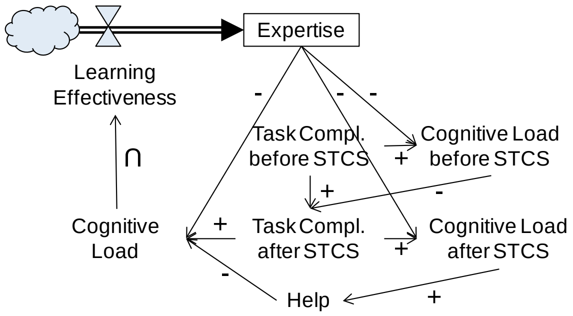

Figure III-7: Model 2: A Model of Cognitive-Load-Based Learning and Static Support

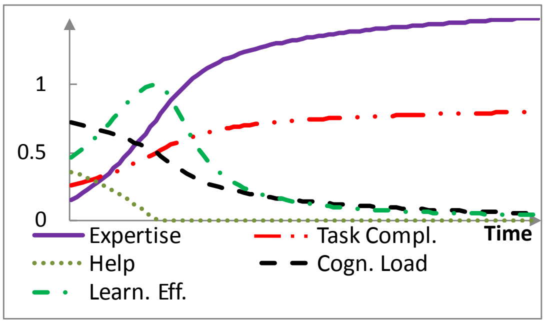

Figure III-8: Model 2 Behavior (initial expertise = .15, task complexity before simple-to-complex sequencing = .5)

Help is a second strategy to reduce cognitive load beyond simple-to-complex sequencing. Help may take the form of supportive information or simplified task types (Krancher and Dibbern 2012; Van Merriënboer et al. 2003). Examples of supportive information include formal presentations and informal conversations[7] on domain-related concepts such as the

---

[6] The qualifier *static* is used because the foundations that determine whether anticipated cognitive load results in support do not change over time.

[7] Documents may be a further source of supportive information. However, the availability of documents may be to a lesser extent the result of dynamic processes in transitions. For reasons of parsimony, this paper therefore focuses on social help, leaving the influence of documents subject to future research.

software architecture, application domain concepts, or the client's maintenance processes (Chua and Pan 2008). Supportive information can decrease cognitive load when it allows learners to link task-related information to higher-order concepts. Help may also be provided by simplifying task types. Task types are simplified when parts of the solution to a problem are provided or goal conditions are relaxed (Van Merriënboer et al. 2002). For instance, the SME may provide parts of the solution by creating a detailed design document (Dibbern et al. 2008). Help may result both from the proactive anticipation of the need for help based on cognitive-load concerns and from the reactive help seeking behavior of vendor engineers as a result of cognitive overload (Krancher and Dibbern 2012). Model 2 reflects both scenarios by assuming that help is a function of the cognitive load that would materialize after simple-to-complex sequencing if no help were provided. The more this cognitive load exceeds the optimal moderate level, the more help will be provided. Cognitive load is then a function of three antecedents: expertise, task complexity after simple-to-complex sequencing, and help. The additional assumptions made to explore the behavior of model 2 are given in Appendix III-2.

Figure III-8 shows the behavior of model 2 assuming the same initial expertise value of .15 and the same task complexity before simple-to-complex sequencing of .5 as in Figure III-5. It is insightful to compare the paths in Figure III-8 (model 2) to the paths in Figure III-5 (model 1). As a consequence of the simple-to-complex sequencing strategy, task complexity is initially lower and later on higher in model 2 than in model 1. Moreover, help is provided in function of cognitive load. As a result, cognitive load is initially lower, resulting in higher learning effectiveness from the beginning. Also, the curtosis of the learning effectiveness curve is lower, which indicates longer periods of effective learning and yields higher expertise values. Taken together, the results are indicative of the benefits from the use of cognitive load regulation strategies during coexistence.

2.4 Model 3: A Model of Cognitive-Load-Based Learning and Dynamic Support

Although support (i.e. simple-to-complex sequencing and help) may positively affect the vendor engineer's learning outcomes, it may be contingent on the willingness of the SME to provide the support and on the ability and willingness of the vendor engineer to make use of the support (Krancher and Slaughter 2013). Many context factors in SMOO may hinder constructive interaction between the vendor engineer and the SME, such as cultural distance, language barriers, conflict, low expertise of the vendor engineer, and low familiarity (Dibbern et al. 2008; Gregory et al. 2009; Krancher and Slaughter 2013). These barriers may be harmful for the vendor engineer's learning outcomes because they may impede the social interactions that would result in the required support (Gregory et al. 2009; Krancher and Slaughter 2013). This observation points the attention to the mechanisms that may determine to what extent support is given.

While it is beyond the scope of this article to exhaustively theorize on barriers to knowledge sharing, it is within the scope to discuss how client management decisions and forces endogenous to the model impact support. Figure III-9 shows model 3, which includes these antecedents to support. Consistent with the literature (Gregory et al. 2009; Krancher and Slaughter 2013), it suggests that control (Kirsch 1996) may moderate whether too high (or too low) cognitive load results in simple-to-complex sequencing or in help. When formal and clan controls (FCC) (Kirsch 1996) are absent and when vendor engineers are not able or willing to self-control their learning process, support will be lacking even if needed (Krancher and Slaughter 2013). Conversely, when substantial mechanisms of formal control (such as formal procedures for conducting knowledge transfer) and clan control (such as norms related to knowledge transfer that are enforced in social interaction) are present and when the vendor engineers have the stronger foundations to self-control their learning, chances high that high cognitive loads may be mitigated by appropriate support (Krancher and Slaughter 2013). Model 3 therefore includes control as a moderator in the relationships that determine support, and it includes FCC and self-control antecedents to the overall amount of control. Moreover, the model 3 reflects findings of recent research on how self-control may change over time.

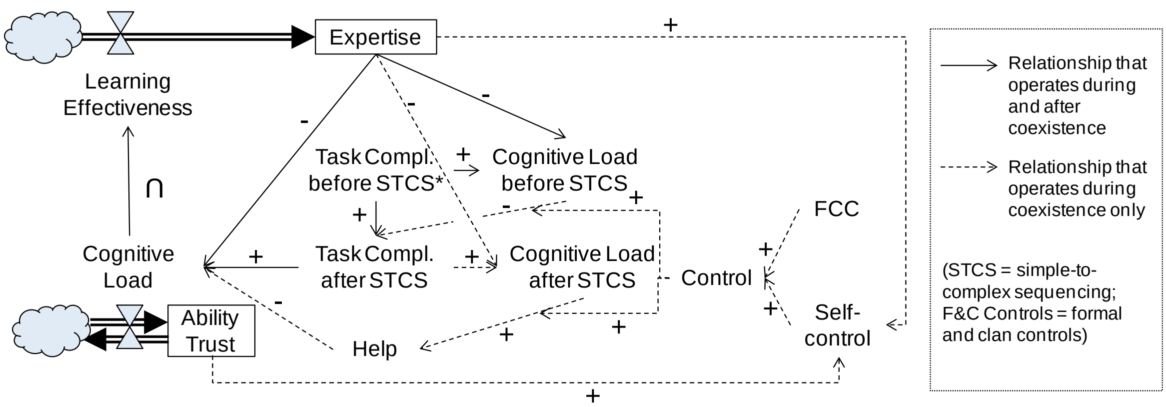

Figure III-9: Model 3: A Model of Cognitive-Load-Based Learning and Dynamic Support

Self-control may be a function of expertise and of ability trust (Krancher and Slaughter 2013; see also study 3). Expertise enables self-control because experts will have more free mental resources to reflect about the learning process and trigger appropriate corrective actions (Baumeister et al. 1998; Krancher and Slaughter 2013; Moos and Azevedo 2008). Ability trust may influence self-control for two reasons. First, when employees have high trust in the ability of a coworker, they are expected to help overloaded coworkers because they anticipate that their low performance may improve soon due to their high ability (Lepine and Van Dyne 2001; Weiner 1985). Conversely, when employees lack trust in the ability of a coworker, they are expected to cease supportive behavior because they do not expect any change in the performance of the coworker (Lepine and Van Dyne 2001; Weiner 1985). Second, vendor engineers may expect these judgments and refrain from seeking support when the relationship lacks trust (Krancher and Slaughter 2013). These theoretical arguments are reflected in model 3 as follows. Ability trust is included as a stock. It may thus increase or decrease over time. Cognitive load is the rate that determines inflows and outflows to ability trust. High cognitive load implies low task performance (Paas et al. 2003). Under repeatedly high cognitive loads and thus low task performance, the trust in the ability of the vendor engineer may decrease (Mayer et al. 1995). In a similar vein, stably low cognitive load will be associated with strong task performance and will yield increases in ability trust.

Making some assumptions (see Appendix III-2) allows again graphically exploring the behavior of the model. Figure III-10 and Figure III-11 assume a slightly lower start value of expertise (.10) than in the previous examples and medium (.5) task complexity before simple-to-complex sequencing. Both graphs assume medium (.5) initial ability trust[8]. The only difference between the two figures lies in the amounts of FCC. Whereas no FCC (value of 0) are assumed in Figure III-10, very high FCC (value of 1) are in place in Figure III-11. It is insightful to observe how the differences in control lead to different paths in the two examples. In Figure III-10, little help and little simple-to-complex sequencing (not shown in the graph) are initially provided. This is because FCC are absent and self-control is weak because of rather low trust and expertise. A very long period of weak learning can be observed. Also, ability trust decreases because the SME observes the outcomes of the vendor engineer's cognitive overload over a substantial period of time. Figure III-11 shows

[8] Trust may (or may not) initially be at a medium level when trust in the vendor organization cascades into trust in the individual engineer or when subject matter experts were involved into the selection of vendor personnel.

a different picture. Here, high FCC can partially compensate for the initially weak self-control. As a result, more help and more simple-to-complex sequencing (not shown in the graph) are provided. This results in higher values of learning effectiveness from the beginning. Interestingly, improved learning outcomes also translate on the relational level. Because the SME observe higher task performance as a result of lower cognitive load (Paas et al. 2003), ability trust initially remains at medium levels and increases far earlier than in Figure III-10.

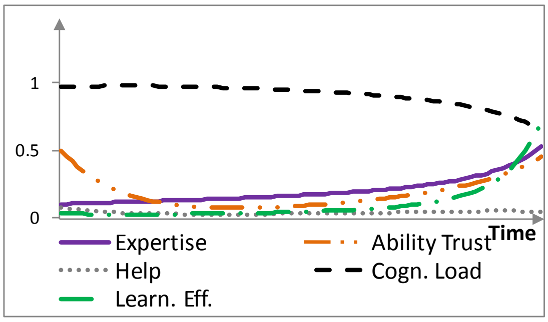

Figure III-10: Model 3 Behavior (FCC = 0)

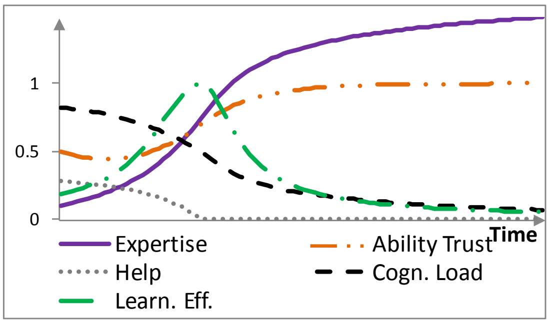

Figure III-11: Model 3 Behavior (FCC = 1)

These observations indicate vicious and virtuous circles that may operate in SMOO transitions. When initial expertise is low and organizational controls are sparse (such as in Figure III-10), one may observe a vicious circle of low support, low learning effectiveness, decreasing ability trust, decreasing self-control (in response to lower trust), less support (in response to less self-control), and thus continuously weak learning. The negative effects of cognitive overload for learning outcomes (Sweller et al. 1998) and for ability attribution (Lepine and Van Dyne 2001; Weiner 1985) reinforce each other. Conversely, when high amounts of FCC are in place, support will be provided despite low self-control. This soon gives rise to a virtuous circle in which effective learning (due to moderate cognitive load) and positive ability attribution reinforce each other.

Model 3 is also useful to explore how the duration of the coexistence phase impacts learning. Simple-to-complex sequencing and help are only accessible during coexistence because only then will the SME be available as a source of support. This is reflected in Figure III-9. Whereas the relationships drawn by dashed arrows operate during coexistence only, the relationships indicated by the solid arrows operate both during and after coexistence. Figure III-12 and Figure III-13 show two otherwise identical transitions (initial expertise = .1, task complexity = .5, FCC = 1) that only differ in the duration of the coexistence phase. Management planned the end of coexistence after 10 out of 100 periods in

Figure III-12 (see the dashed line for the end of the coexistence phase). More time was given to the vendor engineer in Figure III-13, where coexistence ended after 30 periods. The differences between the two transitions are remarkable. In Figure III-12, the vendor engineer's learning process is disrupted at a stage during which she/he highly depends on simple-to-complex sequencing and help. Learning is only marginal after the end of coexistence due to very high cognitive load. Conversely, the vendor engineer in Figure III-13 had already gained substantial expertise when coexistence ended. She/he is therefore able to effectively learn after period 30 even though no support is provided any more. The observations suggest that coexistence durations may not be linearly related to transition outcomes. The Monte Carlo simulation that is described in the next paragraph will shed further light on this.

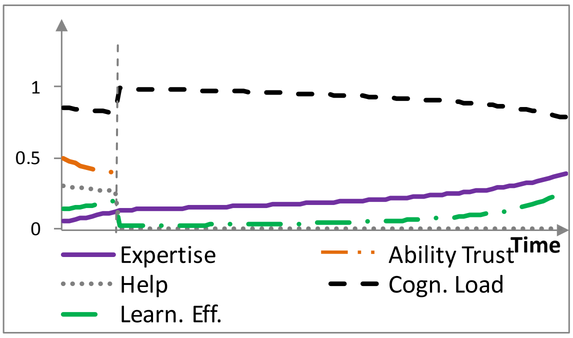

Figure III-12: Model 3 Behavior (coexistence ends after 10 out of 100 periods (see dashed line))

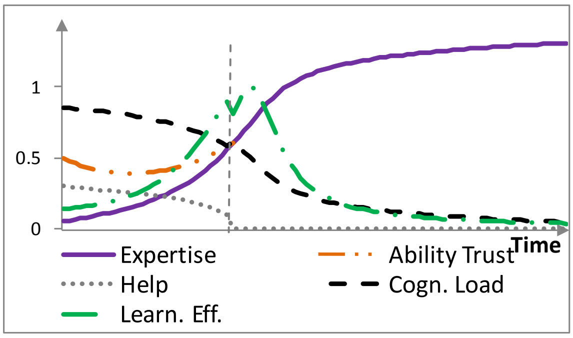

Figure III-13: Model 3 Behavior (coexistence ends after 30 out of 100 periods (see dashed line))

3 A MONTE-CARLO SIMULATION

The discussion in the previous section may have been helpful to explore how dynamic forces operate in SMOO transitions. It may also have been useful to illustrate how management decisions such as the amounts of FCC or coexistence durations may impact the dynamics in the model. However, the discussion up this point has suffered from some limitations. First, few particular cases of combinations of model variable values have been examined in an illustrative manner. This allows limited inference on, for instance, how client management decisions may collectively impact dynamics in transitions. Second, client management decisions have not yet been linked to transition outcomes. Third, the choices of parameters in Appendix III-2 may raise the question to what extent the findings are robust to alternative choices. Fourth, the dynamics in real transitions may be less predictable than suggested in the illustrations of the previous section. For instance, task complexity may vary randomly in function of the modification requests raised by users. Stochastic variation in task complexity may impact the dynamics in the model, e.g. when unusually simple tasks bear valuable learning opportunities for otherwise overloaded vendor engineers.

A Monte Carlo simulation has been conducted to mitigate these limitations. The goal of the simulation was to explore how the three client management decisions (the engagement in personnel selection to control the vendor engineer's initial expertise, the amounts of FCC, and the duration of the coexistence phase) impact transition outcomes. A central outcome of transitions is the transition duration, which may be defined as duration of the phase after which the vendor engineer is able to independently perform the software-maintenance task. It was thus explored how the three management decisions impact transition durations.

The simulation was designed as follows. Simulation runs were conducted for each permutation of the following model parameter values: initial expertise from 0 to .6 (step size .1), task complexity mean from 0 to 1 (step size .2), FCC from .0 to 1 (step size .25) and coexistence duration from 0 to 100 (step size 1). For each permutation, 1000 model runs were calculated, where each run comprised a simulation of 100 periods. The same calculation procedure was used as in the models reported in the previous section with the only difference that task complexity was considered a normally distributed random variable with a standard deviation of .25. The calculation procedures calculated the values of all model variables for 100 periods based on the assumptions given in Appendix III-2. This calculation procedure was then repeated 1000 times (i.e. in 1000 model runs) because each model run may unfold an idiosyncratic dynamic due to stochastic task complexity. After 1000 model runs, the transition duration was determined in the following manner. It was assumed that a vendor engineer is able to solve a maintenance request if the cognitive load is less than or equal to .5. Furthermore, a target service level of 90% was assumed. A period was therefore considered the end of transition when it was the first period in which the vendor engineer was able to solve the maintenance requests in at least 900 out of the 1000 simulation runs (corresponding to the service level of 90%). That is, the period was the first period in which cognitive load was less than or equal to .5 in at least 90% of the simulation runs. The 1000 simulation runs were iterated for each combination of initial expertise, task complexity mean, FCC, and coexistence duration. The simulation thus yielded one transition duration value for each combination of these input parameters.

The simulation was run in Matlab R2012b. The code is displayed in Appendix III-3.

4 RESULTS OF THE MONTE-CARLO SIMULATION

The results of the Monte Carlo simulation provide insights into how coexistence duration, initial expertise, and FCC impact transition durations. Figure III-14 helps understand how initial expertise and coexistence durations may impact transition durations. The figure shows the relationship of coexistence duration and transition duration for different values of initial expertise, assuming moderately high task complexity (.6) and medium amounts of FCC (.5). Under moderately high initial expertise (see the graph for expertise = .6), coexistence durations have no or virtually no impact on transition durations as indicated by the nearly horizontal graph. This is because the moderately high initial expertise yields moderate cognitive load levels from the beginning, which eliminates the need for support. In these settings, expenses for long coexistence phases may not pay off. In other words, recruiting vendor staff with strong related prior experience may save substantial costs for coexistence. However, this may not always be possible. When the maintenance of custom-built or strongly customized applications is outsourced, many domains of the task may be new to otherwise experienced vendor engineers. This will translate into initially lower expertise levels (see also study 1). The graph for expertise = .2 in Figure III-14 illustrates how coexistence durations may affect transition outcomes in such a case. Whereas transitions can be expected to take approximately 60 periods under the conditions of Figure III-14 and if there is no coexistence of SME and vendor engineers, the transition can be shortened to 40 periods if the SME and vendor engineers coexist during transition. Longer coexistence durations may thus result in better transition outcomes. When initial expertise is set to 0, transitions never terminate under the conditions of Figure III-14[9].A comparison of the three graphs in Figure III-14 and the unsuccessful transition in case of initial expertise = 0 indicates that initial expertise may not be linearly related to transition outcomes. Improvements from moderate to high expertise may slightly improve transition outcomes. The stronger the initial expertise negatively deviates from the demands imposed by task complexity, the more dramatic may be the negative effects for transition outcomes (as seen by the fundamental difference between the models with initial expertise = 0 and initial expertise = .2).

[9] This is why no curve for initial expertise = 0 appears in Figure III-14.

Figure III-15 helps explore the impact of FCC on transition outcomes. It shows the relationships of coexistence duration and transition duration for different values of FCC, assuming task complexity of .6 and low initial expertise of .2 (i.e. varying the upper graph in Figure III-14 for different values of FCC). The graphs indicate that the benefits from longer coexistence phases depend on FCC. When FCC are absent, longer coexistence may only slightly shorten transition durations. This is suggested by the low slope of the upper graph (FCC = 0) in Figure III-15. Conversely, when strong FCC are in place, longer coexistence durations may result in considerably shorter transitions up to a threshold (see the graph for FCC = 1). This finding may be explained by the vicious and virtuous circles described in the previous section. Under low FCC, a vicious circle of weak learning, high cognitive load, negative ability attribution and decreasing support may manifest. Longer coexistence phases will then bring only minor benefits because constructive interaction between the SME and vendor engineers will be scarce. High FCC may break this vicious circle and give rise to a situation in which strong learning due to moderate cognitive load and positive ability attributions reinforce each other. This suggests that managers should combine moderately long coexistence periods and high FCC when initial expertise is moderately low.

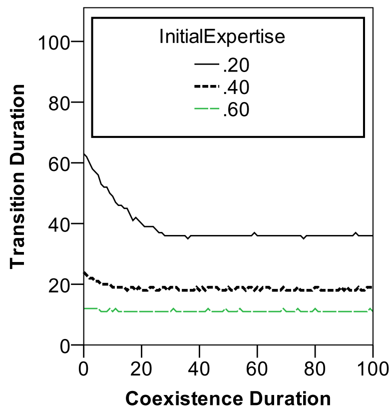

Figure III-14: Transition Duration in Function of Coexistence Duration and Initial Expertise (Task Complexity = .6, FCC = .5)

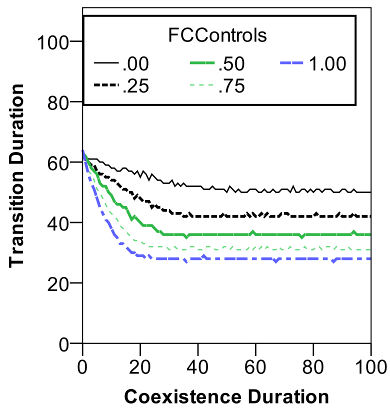

Figure III-15: Transition Duration in Function of Coexistence Duration and FCC (Task Complexity = .6, Initial Expertise = .2)

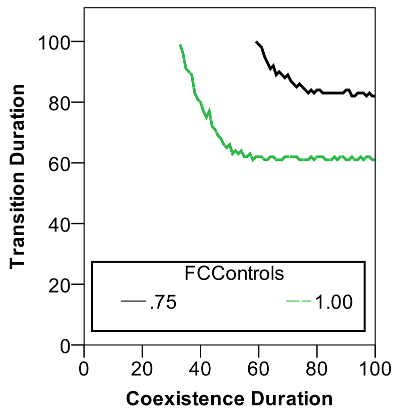

Figure III-16: Transition Duration in Function of Coexistence Duration and FCC (Task Complexity = .6, Initial Expertise = 0)

Figure III-16 illustrates how coexistence duration and FCC interact when the initial expertise is very low (0), still assuming task complexity of .6. The diagram shows that, under these conditions, transitions terminate successfully only under high FCC. Even then, long transition phases are expected, which may jeopardize the business case of offshoring. If very low expertise values are combined higher task complexity such as .8, the transitions never terminate in the simulation model, irrespective of the amount of FCC (not shown in the figures).

These results may be summarized in the following propositions:

P1: *Transition outcomes strongly depend on the ratio of initial expertise and task complexity. (a) When this ratio is very small, transitions fail irrespective of coexistence duration and FCC. (b) When ratio is very large, transitions are successful irrespective of coexistence durations and FCC. (c) Between these thresholds, higher ratios are associated with shorter transition durations.*

P2: *(a) When the ratios of expertise and task complexity are neither very small nor very large, higher coexistence durations are associated with lower transition durations up to a threshold of coexistence durations. (b) These benefits (i.e. the extent of the decrease of transition durations) are moderated by the amounts of FCC in such way that more FCC yield higher benefits from longer coexistence.*

Sensitivity analyses were run to examine whether the results are robust to alternative assumptions of the model parameters given in Appendix III-2. The results are shown in Table III-4. While alternative parameter choices reinforced or attenuated some of the effects, the propositions stated above found support in all alternative model specifications. The results do therefore not seem to be artifacts of the choices of these model parameters.

Table III-4: Results of Sensitivity Analyses

Parameter Values	*Meaning of Parameter*	*Consistent Results*
$a \in \{8, 16, 32\}$	Sensitivity to high or low cognitive loads	Yes
$b \in \{.01, .05, .1\}$	Learning rate	Yes
$c \in \{.01, .05, .1\}$	Task complexity change by simple-to-complex sequencing	Yes
$d \in \{.05, .1, .15\}$	Help rate	Yes
$f \in \{.2, .5, .8\}$	Relative emphasis of self-control	Yes
$g \in \{.05, .1, .2\}$	Ability trust change rate	Yes

5 DISCUSSION

The goal of this paper was to explore how three client management decisions (the engagement in vendor staff selection to control their prior experience, the amount of FCC, and the duration of the coexistence phase) affect dynamics in learning and helping behavior in

SMOO transitions. Applying the system dynamics paradigm to findings from previous research on SMOO and reference theories, a dynamic model of cognitive-load-based learning and support was developed and explored. A Monte Carlo simulation helped examine how the three client management decisions impact the model dynamics and thus transition outcomes.

The results indicate that recruiting vendor engineers with strong prior related experience may have the strongest effect on transition outcomes. These transitions benefit from a virtuous circle of effective learning under moderate cognitive load and appropriate support due to positive ability attribution. Recruiting high-expertise staff also obviates the need for long coexistence phases and for significant FCC. Expenses for staff selection may thus frequently pay off. Yet, client management may not always be able to recruit staff with high initial expertise, e.g. when the maintenance of custom-built software applications is outsourced (see also study 1). Such cases begin with lower expertise levels and are thus prone to vicious circles of weak learning due to high cognitive load and weak ability attribution that results in decreasing support to vendor engineers. Under such circumstances, longer phases of coexistence of the SME and vendor engineers may only yield substantial benefits if they are accompanied by FCC. FCC may break the vicious circle because they enforce constructive interaction between the vendor engineers and the SME.

This paper makes several contributions. It contributes to the literature on offshore outsourcing by proposing how transition-related managerial decisions impact the success of SMOO projects. Although the existing literature suggests that client management involvement is central for transition outcomes (Dibbern et al. 2008; Gregory et al. 2009), this paper is the first to systematically theorize on how these decisions affect transitions. It offers new explanations for “knowledge transfer blockades” (Gregory et al. 2009, p. 1) that are grounded in the interaction of dynamic mechanisms over time. It also suggests new explanations for how and why client management may break knowledge transfer blockades. Awareness of the dynamics explored and illustrated in this paper could thus help practitioners avoid blockades and better manage SMOO transitions.

Although this was not within the primary focus of the paper, the results may also have implications for the debate in the outsourcing literature on whether formal and informal controls are complements or substitutes (Choudhury and Sabherwal 2003; Goo et al. 2009; Gulati 1995; Poppo and Zenger 2002; Woolthuis et al. 2005). Our model adds to this literature by describing a new mechanism that may give rise to a complementary relationship.

FCC directed to knowledge transfer may breed trust because they foster support and thus impede negative ability attribution due to cognitive overload (see Figure III-10 and Figure III-11). Increasing trust may then allow greater self-control. More FCC will thus be associated with more self-control.

Several limitations of the study may be acknowledged. First, the study does not make immediate use of empirical data, which may limit the external validity of the findings. Second, the weights of the antecedents to cognitive load are based on results from nested case study data, which may be limited in their statistical generalizability. Future empirical research may help ascertain the effect sizes and thus provide a stronger basis for future simulations. Third, the study assumed that tasks are handed over from one SME to one vendor engineer. Although it may not be uncommon that one individual takes over the maintenance of a software application (Krancher and Dibbern 2012), group processes in vendor teams may impact the dynamics assumed in the model of this paper. These team processes are not taken into account in the model. Yet, a solid understanding of individual learning process may provide a fruitful base for theorizing on group learning. Fourth, the theorizing on the effects of FCC assumed that client manager may exert control into directions that support the learning of vendor engineers. This may only materialize when client managers are knowledgeable in the transformation process (Kirsch 1996), i.e. when they have knowledge on how people learn.

Future studies may test the propositions of this paper in empirical settings. Future work may also expand the system dynamics model in various was. The model may be extended to include the effects of barriers to knowledge-sharing such as conflict, cultural differences, and language barriers (Gregory et al. 2009; Krancher and Slaughter 2013). Moreover, task delegation behavior may be included. The SME may alter not only their support, but also their behavior with respect to the delegation of tasks in function of ability attribution (Gregory et al. 2009; Krancher and Slaughter 2013; Lepine and Van Dyne 2001). This is not reflected in the current model and could unfold additional dynamics because vendor engineers may lack learning opportunities when tasks are not delegated. The model may also be extended by including documents as a source of supportive information. While their availability is not limited to the coexistence phase, their effectiveness may differ based on the ratio of task complexity and expertise (see study 2). The dependency on expertise may give rise to further interesting dynamics. The model may also be extended to explain and predict offshoring extra costs for knowledge transfer, specification, and control

(Dibbern et al. 2008). Finally, future research may also seek closed-form solutions to the mathematical problems discussed in this study.

APPENDIX III-2: MODEL ASSUMPTIONS

Assumptions in Model 1

The following assumptions are made in model 1:

A1: The values of all rates, auxiliaries, and constants and the start values of stock variables are within a range of 0 (very low) to 1 (very high).

A2: The regression coefficients reported in Krancher and Dibbern (2012) reflect the strengths of the relationships of cognitive load with its antecedents.

A3: The inverted-U-shaped relationship between cognitive load and learning effectiveness obeys the following functional form, where *a* is a parameter indicating the sensitivity to high or low cognitive loads and 0.5 is assumed to be the optimal level of cognitive load:

$$learningeffectiveness = e^{-a \cdot (cognitiveload - 0.5)^2}$$

A4: The following integral function describes the evolution of expertise (ex) in function of time t, where *b* is a parameter for adjusting the scale between learning effectiveness and expertise (learning rate base value):

$$ex(t) = \int_{x=0}^{t} b \cdot learningeffectiveness(x) dt$$

A5: The following values have been chosen for the parameters: a = 16; b = 0.05.

Additional Assumptions in Model 2

Model 2 makes the following assumptions in addition to the assumptions made in model 1:

A6: Simple-to-complex sequencing adjusts task complexity so that cognitive load is closer to a medium level (0.5). Task complexity after simple-to-complex sequencing is therefore calculated as follows, where *c* indicates the magnitude of simple-to-complex sequencing:

$$taskcomplexityafterSTCS = taskcomplexitybeforeSTCS + c \cdot (.5 - cognitiveloadwithoutSTCS)$$

A7: No help is provided if the cognitive load after simple-to-complex sequencing is below a medium level (0.5); else, help is calculated as follows, where *d* indicates the base line magnitude of help provided (help rate):

$$help = d \cdot (cognitiveloadafterSTCS - .5)$$

A8: The following values have been chosen for the parameters c and d: c = .5; d = 1.

Additional Assumptions in Model 3

Model 3 makes the following assumptions in addition to the assumptions made in model 2:

A9: Control is the weighted sum of FCC and self-control, where *f* denotes the weight of self-control:

$$control = f \cdot selfcontrol + (1 - f) \cdot FCC$$

A10: Self-control is equally determined by expertise (ex) and ability trust:

$$selfcontrol = \tfrac{1}{2}(ex + abiltiytrust)$$

A11: Ability trust increases or decreases in function of cognitive load, where *g* denotes the sensitivity to latest cognitive load levels:

$$abilitytrust(t) = (1 - g) \cdot abilitytrust(t-1) + g \cdot \max(0, \min(1, 2 - 2 \cdot cognitiveload))$$

A12: The following values have been chosen for the parameters f and g: f = .5; g = .1.

APPENDIX III-3: SIMULATION CODE[10]

parameters.m

```
taskcomplexitystandarddeviation = 0.25;
initialtrust = 0.25;
cognitiveloadsensitivity = 16;
servicelevelsteadystate = 0.9;
learningrate = 0.05;
trustchangerate = 0.1;
helprate = 2;
```

---

[10] I thank Khôi Tran for his support in implementing the simulation procedure in Matlab.

```
simpletocomplexsequencingrate = 0.5;
selfcontrolweight = 0.5;
numberofruns = 1000;
betataskcomplexity = 0.49;
betahelp = -0.39;
betaexpertise = -0.8;
intercept = 0.85;
OUTPUTDIR = 'c:/matlab_output/';

initialexpertisevector = 0.0:0.1:0.6;
taskcomplexitymeanvector = 0.0:0.2:1;
fccontrolsvector = 0.0:.25:1;
coexistencevector = 0:1:100;
```

Testdata.m

```
classdef Testdata

    properties
        initialexpertise
        taskcomplexitymean
        fccontrols
        coexistence
        numberofruns

        % Matrix for generated data (NumberOfRuns x 100)
        Mat
        filename

        dataGenerated
    end

    methods
          function obj = Testdata( initialexpertise, taskcomplexitymean,
               fccontrols, coexistence, numberofruns )
            obj.dataGenerated = false;
            parameters
            obj.filename = [OUTPUTDIR, 'Expertise_',
                num2str(initialexpertise),
                '_TaskComplexityMean_', num2str(taskcomplexitymean),
                '_FCControls_', num2str(fccontrols),
                '_Coexistence_', num2str(coexistence),
                '_NumberOfRuns_', num2str(numberofruns),
                '.dat' ];
            if exist(obj.filename, 'file')
                return
            end

            obj.dataGenerated = true;

            obj.Mat = zeros(numberofruns, 100);
            for r=1:numberofruns
                % Reset values
                expertise = initialexpertise;
                trust = initialtrust;

                % Preallocate array
                Data = zeros(100, 1);

                for period=1:100
                    randomnumber = max( rand(), 0.000001 );
                    taskcomplexity = taskcomplexitymean +
```

```
                        norminv(randomnumber) *
                        taskcomplexitystandarddeviation;
                    CLwithoutLR = intercept + betaexpertise * expertise +
                        betataskcomplexity * taskcomplexity;
                    if period <= coexistence
                        selfcontrol = (expertise + trust) / 2;
                        control = fccontrols * (1 - selfcontrolweight) +
                            selfcontrol * selfcontrolweight;
                        taskcomplexitychange =
                            simpletocomplexsequencingrate *  ( 0.5 –
                            CLwithoutLR) * control;
                        CLafterSTCS = CLwithoutLR + taskcomplexitychange
                            * betataskcomplexity;
                         if CLafterSTCS > 0.5
                            help = (CLafterSTCS - 0.5)* control *
                               helprate;
                        else
                            help = 0;
                        end
                        cognitiveload = CLafterSTCS + help * betahelp;
                    else
                        cognitiveload = CLwithoutLR;
                    end
                    Data(period) = cognitiveload;
                    learningeffectiveness = exp(-cognitiveloadsensitivity
                         * (cognitiveload - 0.5) ^ 2 );
                    expertise = expertise + learningrate *
                         learningeffectiveness;
                    if period <= coexistence
                        trust = trust * ( 1-trustchangerate ) + max(0,
                            min(1, 2 - 2 * cognitiveload)) *
                            trustchangerate;
                    end
                end
                % set the generated data in the matrix
                obj.Mat(r,:) = Data;
            end

        end

    end

end
```

run.m

```
clear
parameters

% permutate through the parameters
for initialexpertise=initialexpertisevector
    initialexpertise
    for taskcomplexitymean=taskcomplexitymeanvector
        taskcomplexitymean
        for fccontrols=fccontrolsvector
            fccontrols
            parfor coexistence=coexistencevector
                coexistence
                td = Testdata(initialexpertise,taskcomplexitymean,
                     fccontrols, coexistence,200);
                if td.dataGenerated
                    parallelSave(td.filename, td);
```

```
                end

            end
        end

    end
end

% Results matrix size: all permutations * 18 columns (for the attributes)
[a, size_e] = size(initialexpertisevector);
[a, size_tc] = size(taskcomplexitymeanvector);
[a, size_c] = size(fccontrolsvector);
[a, size_co] = size(coexistencevector);

permutations =  size_e * size_tc * size_c * size_co;

Results = zeros( permutations, 18 );
i = 1;

% permutate through the parameters
for initialexpertise=initialexpertisevector
    for taskcomplexitymean=taskcomplexitymeanvector
        for fccontrols=fccontrolsvector
            for coexistence=coexistencevector
                % calculate service level
                sl =   ServiceLevel(initialexpertise,taskcomplexitymean,
                       fccontrols,coexistence,100);

                % store the results in the excel sheet
                Results(i, 1) = initialexpertise;
                Results(i, 2) = taskcomplexitymean;
                Results(i, 3) = fccontrols;
                Results(i, 4) = coexistence;
                Results(i, 5) = sl.timetillsteadystate;
              i = i + 1;
            end
        end

    end
end

%save('results','-ascii','Results');
xlswrite('results.xls', Results);
```

ServiceLevel.m

```
classdef ServiceLevel
    %SERVICELEVEL Calculates the service level vector for a TestData ob-
ject

    properties
        filename
        slevel
        timetillsteadystate
    end

    methods
        function this=ServiceLevel( initialexpertise, taskcomplexitymean,
               fccontrols, coexistence, numberofruns )
            % load parameters from parameters.m
            parameters
            % load data
```

```
            datafile = = [OUTPUTDIR, 'Expertise_',
                num2str(initialexpertise),
                '_TaskComplexityMean_', num2str(taskcomplexitymean),
                '_FCControls_', num2str(fccontrols),
                '_Coexistence_', num2str(coexistence),
                '_NumberOfRuns_', num2str(numberofruns),
                '.dat' ];
            dataraw = load(datafile,'-mat', 'data');

            % Return -1 if steady state is not reached
            this.timetillsteadystate = -1;
            % testdata
            td = dataraw.data;

            % preinitialize servicelevel vector for performance
            this.slevel = zeros(100);

            % evaluate Service Level for each timeperiod
            for timeperiod=1:100
                % conditional average (only sum if cognitive load is <=
                    0.5)
                this.slevel(timeperiod) = sum( td.Mat(:,timeperiod) <=
                    0.5 ) / numberofruns;
                % set TimeTillSteadyState to the first time period where
                    the servicelevel exceeds ServiceLevelSteadyState
                    (from parameters.m)
                if this.timetillsteadystate == -1
                    if this.slevel(timeperiod) >= servicelevelsteadystate
                        this.timetillsteadystate = timeperiod;
                    end
                end
            end
        end
    end

end
```

parallelSave.m

```
function [] = parallelSave( filename, data )
    save( filename, 'data' );
end
```

CHAPTER IV CONCLUSION

Although knowledge transfer is central to the success of software-maintenance offshore outsourcing (SMOO) projects, the existing literature provided scant guidance on how to govern and design effective knowledge transfer to vendor staff during transition. The four studies of the dissertation were intended to fill this gap. This concluding chapter summarizes the implications from these four studies for research and for practice.

IMPLICATIONS FOR RESEARCH

The primary objective of this dissertation was to contribute to a theory of the governance and design of effective knowledge transfer in SMOO transitions. Figure IV-1 shows the theory that has been developed through the four studies. The theory is grounded on data from case studies of knowledge transfers to vendor on-site coordinators, and on three reference theories: cognitive load theory, control theory, and attributional theory. The theory is summarized next.

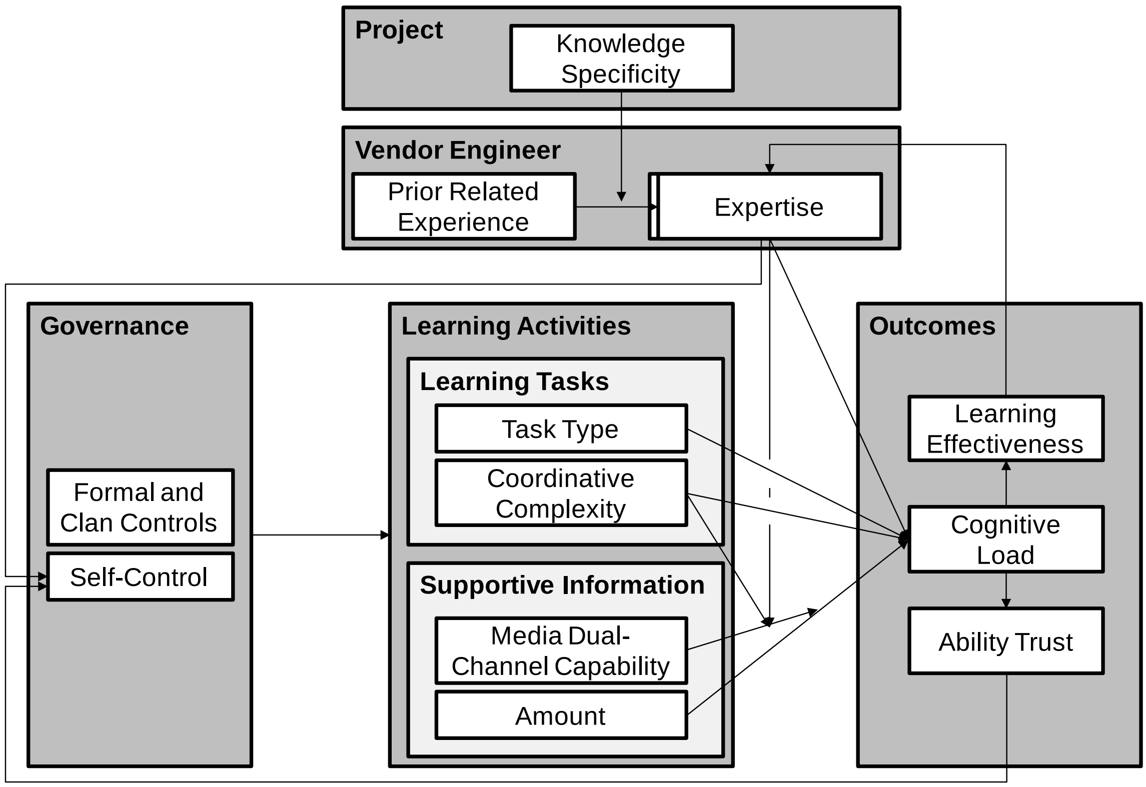

Figure IV-1: A Theory of the Governance and Design of Effective Knowledge Transfer in SMOO

Learning activities are at the heart of the developed theory. The dissertation adopted a view of knowledge transfer as a sequence of learning tasks and related supportive information (Van Merriënboer et al. 2003) that collectively help vendor engineers acquire the task knowledge (the knowledge required to perform the software-maintenance task). Processing learning tasks and supportive information are thus assumed to be the central learning activities. This view is consistent with many modern learning theories (Collins et al. 1991; Jonassen 1997; Merrill 2002; Van Merriënboer et al. 2002) and distinct from understanding knowledge transfer as information delivery or communication of knowledge, which reduces knowledge transfer to providing supportive information. Learning tasks help learners acquire the constituent skills that are required for complex tasks (Van Merriënboer et al. 2003) such as software maintenance. However, learning tasks risk overloading the learner, which makes strategies for managing cognitive load necessary (Van Merriënboer et al. 2003; Van Merriënboer and Sweller 2005). In this view, designing and governing effective knowledge transfer involves ensuring that vendor engineers are provided with the right learning tasks, which implies that appropriate measures to manage cognitive load have been taken. The communication of knowledge (e.g. by making codified knowledge available through repositories or by setting up formal presentations by experts to vendor engineers) may be a mean to the end of load reduction, but no end in itself.

Cognitive load is held to be a pivotal outcome of learning activities. It is the key presumption of cognitive load theory that learning tasks yield effective learning when they impose neither too high nor too low cognitive loads on learners (Schnotz and Kürschner 2007; Van Merriënboer and Sweller 2005). The relationship between cognitive load and learning effectiveness is therefore assumed to follow an inverted-U-shaped form. Although this association was not tested in this study, it is backed by substantial empirical support within a broad array of domains that risk imposing high cognitive loads (Schnotz and Kürschner 2007; Van Merriënboer and Sweller 2005). Study 1 of this dissertation tested the antecedents to cognitive load. It produced evidence consistent with CLT. Expertise had a very strong negative relationship with cognitive load. This makes transitions with novice vendor engineers a difficult endeavor because there is a substantial risk of high cognitive load, weak learning, and, as a consequence, stably low performance. However, study 1 also suggested that there are effective strategies to manage the cognitive load associated with learning tasks so that learning tasks may yield moderate cognitive load even when expertise is relatively low. These strategies include simple-to-complex sequencing of tasks based on their coordinative complexity, the use of simplified task types (such as job-shadowing, providing detailed designs for modification requests or solutions to similar modification requests), and, possibly, supportive information (such as formal presentations, documents, and informal help). Our data indicated that assigning tasks with lower complexity and using simplified tasks types may be associated with greater decreases of cognitive load than providing supportive information. Low-expertise engineers will thus benefit from the strong use of, in particular, the first two strategies so that high cognitive loads are avoided.

Study 2 helped understand why *supportive information* may frequently not substantially relieve cognitive load. The effectiveness of supportive information may be subject to an interaction of *media* attributes, task complexity, and expertise. Media without dual-channel capability (i.e. media that use either an auditory or a visual channel, but not both at the same time) may be severely constrained to convey supportive information when the ratio of task complexity and expertise is large. This is because they do not effectively use the capacities of the two separate channels of working memory in order to cope with critically high cognitive loads. For instance, documents may be ineffective sources of supportive information under such circumstances because they bundle the high cognitive load in the visual channel. The weak relationships of supportive information and cognitive load may thus be partially explained by ineffective media choice.

If it is central to align cognitive load reduction strategies and media choice with *expertise*, it may be appealing to explain and predict expertise levels at the beginning of transitions. For instance, should vendor engineers with five years of experience in data warehousing projects be considered as experts when they are about to take over the maintenance of a custom-developed data warehouse because they can draw on their prior experience? Or are they novices because they are not familiar with the source code of the custom-built software? Should they hence enjoy substantial cognitive load reduction strategies to avoid high cognitive loads or will less scaffolding be equally or even more effective because high expertise will yield moderate load levels even in absence of help? The exploratory analysis of study 1 helped address these questions. The findings included that prior related experience may have benefits for the initial expertise levels, but that the magnitude of these benefits greatly depends on *knowledge specificity* (i.e. the degree to which the required knowledge is specific to the client). The more specific the required knowledge is, the smaller will be the overlap in the knowledge domains of the prior experience and of the task at hand. Prior experience in highly specific software maintenance tasks will thus yield small benefits for initial expertise. Conversely, engineers who take over less specific tasks (such as the maintenance of a moderately customized software package) will begin at higher expertise levels. Knowledge specificity has therefore substantial implications for designing effective knowledge transfer. The higher knowledge specificity, the more will the vendor engineer's learning benefit from load reduction strategies because high cognitive loads will be avoided.

Although load reduction strategies such as simple-to-complex sequencing, task type simplification, and supportive information may frequently be beneficial for the vendor engineers' learning, they may not fully materialize without management involvement. Language barriers, cultural distance, and low familiarity may be some of the forces that may cause knowledge transfer blockades (Gregory et al. 2009), i.e. situations in which the activities that would yield effective knowledge transfer do not occur. This calls for a better understanding of the antecedents to the use of load reduction strategies. The governance of knowledge transfer by client management is one candidate. Study 3 of this dissertation aimed thus at understanding how governance may impact the learning process.

The theory building effort of study 3 suggested that various actions of *governance* such as formal control, clan control, and self-control may foster the assignment of effective learning tasks. However, these modes of control may differ in their foundations and they may thus be differentially salient at different stages during transitions. Whereas client management may decide on the amounts of outcome, behavior, and clan controls, the amount of *self-control* may be highly endogenous to the dynamics of the learning process. Our results suggested that self-control of the learning process by vendor engineers is contingent on their expertise and on the trust of the subject matter experts (SME) in the abilities of the vendor engineers. Self-control may depend on expertise because self-control demands cognitive resources (Baumeister et al. 1998), which are scarce when inexperienced engineers take over tasks. Education psychology has also produced initial evidence that domain knowledge fosters the self-regulation of learning (Moos and Azevedo 2008). Self-control may further depend on ability trust because SME may want to engage in helping behavior when they have high trust in the abilities of the vendor engineers because only then will they expect performance improvements from help. When vendor engineers repeatedly demand help because of cognitive load, ability trust will crowd out and helping behavior will cease because the SME attribute repeatedly high cognitive load to low ability. Vendor engineers may anticipate such attributions and thus refrain from help-seeking before trustful relationships have been established. Attributional theory (Lepine and Van Dyne 2001;

Weiner 1985) may thus help explain why self-control is contingent on ability trust. These findings give rise to unfavorable dynamics in projects that begin at low expertise levels. In these projects, vendor engineers may frequently not be able to self-control their learning process. Extensive formal and clan controls will then be required to initially enforce task delegation and the needed amount of load reduction. These controls compensate for initially low self-control and may be faded out as the foundations for self-control—ability trust and expertise—develop. Conversely, projects that begin at relatively high expertise levels may see relatively strong self-controls from the beginning. The need to complement self-control by formal and clan controls will then be substantially lower.

The findings from the studies 1, 2, and 3 suggest that a complex set of dynamic interactions operates during SMOO transitions (see also the feedback loops present in Figure IV-1). Study 4 explored how client management may favorably shape these interactions through three management decisions: engaging in vendor staff selection to manipulate initial expertise levels, the amounts of formal and clan controls, and the duration of the coexistence of vendor staff and SME. Drawing on the results of the three studies and adopting a system dynamics perspective (Forrester 1961), study 4 suggested that the three decisions may impact the dynamics in different ways and degrees. Engaging in staff selection to select high-expertise vendor engineers may be the most effective strategy because it eliminates vicious circles of weak learning due to high cognitive load and negative ability attribution. However, such strategies may not always be feasible, for instance when custom-built software is to be maintained (i.e. when knowledge specificity is high). Under these situations, formal and clan controls and longer coexistence durations may mitigate vicious circles. A Monte-Carlo simulation suggests that these two strategies, however, may have weak independent effects, but a strong interaction effect. Longer coexistence durations are particularly effective when they are accompanied by higher amounts of formal and clan controls. Conversely, long coexistence durations may be a blunt instrument when client management relies on the self-control by vendor staff because vicious circles of weak learning and negative ability attribution will prevent the intended effect from coexistence (i.e. an increase of expertise).

The developed theory may thus improve our understanding of the dynamics of knowledge transfer during transition. This may be an important contribution to the IS outsourcing literature outsourcing research, which suffered from a “dearth of research on how IS outsourcing arrangements change, and what the factors influencing these changes are" (Dibbern et al. 2004, p. 88).

While the theory has implications for the design and governance of knowledge transfer, it may also offer a complimentary theoretical perspective to explain the highly relevant issue of extra costs in offshoring. Dibbern et al. (2008) found that high-specificity projects faced particularly high unexpected costs for knowledge transfer, specification, control, and coordination and that, consistent with the knowledge-based view of the firm, problematic knowledge transfer lay at the heart of many of these costs. The theory developed in this dissertation suggests that high-specificity projects will frequently begin with low expertise levels. The low initial expertise levels have two immediate consequences. First, they entail a substantial need for cognitive load reduction, e.g. by simplifying task types and by providing supportive information. Second, they entail a need for formal and clan controls because overloaded vendor engineers will lack the cognitive resources required for self-control. This is highly consistent with the findings of Dibbern et al. (2008). Extra costs for specification may be regarded as unplanned efforts to simplify task types. Extra costs for knowledge transfer may subsume unexpected need for supportive information. The weak relationship of supportive information and cognitive load may lead to the unfortunate outcome that the same supportive information is repeatedly provided, which may further increase extra costs. Extra costs for control denote the efforts that may be needed because client management initially overestimated the ability of vendor staff to self-control their knowledge acquisition. The arguments made by the theory of this dissertation are consistent with the arguments of the knowledge-based view of the firm in that they explain the negative outcomes with knowledge asymmetries resulting from low expertise or low absorptive capacity. The theory developed in this dissertation may also extend the framework of the knowledge-based view in that it takes a dynamic perspective and specifies mechanisms to overcome the adverse dynamics from knowledge asymmetries by incorporating theory on how people learn.

Although the theory may thus increase our understanding of the governance and design of effective knowledge transfer and related issues in SMOO, some limitations need to be acknowledged. First and foremost, there is a substantial need for testing and replication that remains unfulfilled at this point. The developed theoretical framework may thus be best seen as an emergent theory which awaits further replication and refinement before it can be considered as an established theory. Several limitations apply at the current stage.

This dissertation has drawn on in-depth qualitative data to suggest how a set of causal mechanisms may operate during SMOO transitions based. The interplay of these mechanisms has been investigated through analytical modeling techniques. Only the relationships of cognitive load with its antecedents have been tested, but this test has been confined to a setting that may be affected by peculiarities such as the focus on on-site coordinators, the involvement of only one client organization, the high success of the projects, the high degree of social embeddedness of vendor staff in the SME teams, and high motivations of client and vendor staff. These settings have been favorable to make somewhat controlled inferences on cognitive skill acquisition issues under similar contextual conditions. Tests of the framework in a range of different settings will increase trust in the generalizability of the findings and would help to further develop the theory.

A particularly central avenue for future research in this realm is the role of barriers. While study 3 suggested that barriers such as language barriers, cultural distance, and low SME motivation may increase the thresholds of required controls so that effective learning tasks are used, there seems to be much room for future theorizing. A better understanding of these barriers may also help sharpen how further offshore-specific factors enter into the theory. Do language barriers only impede cognitive load reduction because the SME may want to avoid straining conversations in a foreign language (Krancher and Dibbern 2012)? Or do language barriers also impose additional cognitive loads on vendor engineers and thus increase the demand for load reduction? Do cultural differences initially reduce social interaction because the SME may initially perceive the vendor engineers as alien (Krancher and Dibbern 2012)? Or may there be more subtle interactions? For instance, does high power distance (Hofstede 1986) boost vicious effects from ability attribution because vendor engineers from high-power-distance cultures will be particularly fearful of being attributed negative ability? Does the national culture of the vendor engineer influence the strategies used for the self-control of learning (Purdie and Hattie 1996)? Does geographic distance only reduce the amount of social interaction as suggested in study 3? Or may it alter the nature of software-maintenance problems because the given states of problems remain obscure when offshore teams are not allowed to access production data to reproduce the problem described in defect descriptions? These issues clearly merit future attention. The framework proposed in this study does not answer them, but may provide a useful point of departure.

Moreover, the theory would benefit from taking the motivation of the vendor engineer and the SME into greater account. Adopting the perspective of CLT, low learner motivation could alter the predictions of the theory because motivation is required to engage in schema acquisition (i.e. in learning). This dissertation only included research cases in which the vendor engineers were described as highly motivated. While it could then be assumed that learning occurred when cognitive loads permitted it, this need not be the case when vendor engineers are not motivated to learn. Motivation could thus moderate the link from cognitive load to learning effectiveness. Future research could pick this issue up. It may also be insightful to test or expand the theory in domains in which the SME have low motivation to share their knowledge. Can constructive interaction between unmotivated SME and high-expertise vendor engineers change their motivation to share knowledge and thus allow sufficient self-control towards the ends of transitions as suggested in study 3?

Another avenue for development of this theory is to connect it with team-level issues such as coordination or group learning. Qualitative research on larger vendor teams than the teams in this study may be one way to identify connecting points between the theory of this dissertation and team-level theories. For example, it may be worth exploring whether transactive memory systems (Oshri et al. 2008) foster task type simplification because they ease the identification of experts who may provide direction on maintenance problems. It may also be insightful to explore how individual-level learning outcomes impact team learning. Is the on-site coordinator's expertise acquisition a precondition for establishing team routines that increase team productivity (Bingham and Eisenhardt 2006)? Do strong team routines lower the learning demands for individuals and/or reduce the need for cognitive load reduction? More avenues to further develop a theory of knowledge transfer in SMOO have been suggested in the individual studies and may not need to be repeated at this point.

The dissertation may also have implications for research beyond the boundary condition of SMOO transitions. The reference theories used in this dissertation have a particularly strong fit with the context of SMOO transitions given the frequent accounts of high cognitive loads and their negative consequences (Chua and Pan 2008; Dibbern et al. 2008; Gregory et al. 2009). Some caution may therefore be advised in generalizing the theory to broader or different domains such as software-maintenance learning or knowledge transfer related to tasks different from software maintenance. Future research would therefore be needed to test to what extent the theory developed in this dissertation may apply to such settings.

IMPLICATIONS FOR PRACTICE

Practitioners find little guidance on how to design and govern effective knowledge transfer in SMOO projects. The results of this dissertation have tentative normative implications to this end. They are tentative because future replication may increase the trust in the generalizability of the findings.

Practitioners should tailor knowledge transfer approaches to specificity and coordinative complexity. The more specific software is to a particular client (such as custom-built software) and the higher the average coordinative complexity of the tasks is (such as in tightly coupled data processing systems), the more and the longer are cognitive load reduction strategies demanded. When the demands for load reduction are high, care should be taken to initially assign simpler-than-average tasks to vendor engineers and to choose simplified task types. For instance, vendor engineers may initially study worked examples by explaining the SME the solution to past maintenance requests or by job-shadowing. They may then work on imitation tasks or completion tasks over a substantial period of time. For instance, the SME may specify design documents and hand over only the implementation to vendor staff. Conventional problem solving tasks should be assigned to vendor staff only after they studied worked examples and successfully processed imitation or completion tasks. The beneficial effects of worked examples should not be underestimated (Renkl and Atkinson 2003) given the strong correlations in our data. They may not only be effective to reduce cognitive load, but also result in more efficient learning because they avoid long-lasting problem solving search processes. Transition teams may institutionalize the beneficial effects from worked examples by setting up regular meetings in which they present to their fellow engineers how tasks have been solved. Beyond simple-to-complex sequencing and task type simplification, supportive information (e.g. formal presentations on certain concepts, informal explanations by SME, documents) may also have beneficial effects, but they are unlikely to substitute for the effects of the other two strategies. Supportive information may thus be best thought as complementing element to a cognitive-load-consistent task assignment strategy. It should not be understood as the central element of knowledge transfer. In settings that have high demands for load reduction, some media may be more effective to convey supportive information than others. The results of this dissertation suggest that face-to-face presentations, phone conferences assisted by screen-sharing, and commented screen recordings may be more effective than documents or simple phone conferences.

When specificity and coordinative complexity are low and vendor engineers with substantial prior related experience are recruited, knowledge transfer approaches will have weak or even no demands for load reduction. In such settings, documents may be an effective source of explicit knowledge and vendor engineers may benefit most from working on conventional tasks from the beginning. Investments in scaffolding strategies may not pay off under these circumstances.

It seems advisable to tailor not only learning task design, but also the amount of organizational controls to the demands for load reduction. Under high demands for load reduction, expectations of substantial self-control by vendor engineers do not appear realistic. Client management should implement substantial formal and clan controls with regards to knowledge transfer in such settings. For instance, client management may specify the tasks on which vendor engineers should work. They may also specify the task type, which implies the amount of direction that is to be provided by the SME. Client management may implement controls on the amounts of sessions intended to elicit knowledge and exert clan controls by establishing expectations on delegation and helping behavior consistent with the demands for load reduction. A further form of organizational controls may be work samples that specify what class of problems the vendor engineers will be expected to solve at what stage during transition. These organizational controls may help substitute for the weak self-control by vendor engineers caused by low expertise and initial lack of trust.

Conversely, when the demands for load reduction are low, the benefits from formal and clan controls can be expected to be lower. When the demands for load reduction are low, the vendor engineers will be able to engage in more self-control. Still, some initial formal and clan controls may be advisable before trustful relationships are established. However, the amounts of formal and clan controls may be substantially lower than in settings that yield high demands for load reduction.

Client management may also make decisions on the engagement in vendor staff selection and on the duration of the coexistence of vendor staff and SME based on specificity and task complexity. Engaging in vendor staff selection to control the prior experience of vendor staff may be generally advisable given the very strong correlations between expertise and cognitive load. However, the benefits from such efforts will be higher when specificity is lower. Elaborated staff selection tests such as a set of work samples may thus have tremendous business cases when low-specificity tasks such as the maintenance of software packages are outsourced. The duration of the coexistence of SME and vendor engineers should also be aligned with load reduction demands. High load reduction demands will frequently entail the need for long coexistence phases, which may exceed six months. Clients may consider implementing a staff augmentation strategy rather than delegating the full responsibility to vendor staff under these settings. Staff augmentation helps keep the SME available to the project and thus fulfill a sustained need for load reduction. However, in settings that demand for high load reduction, long coexistence phases are only expected to be effective to the extent that they are accompanied by strong formal and clan controls.

Although issues of contractual governance were not central in the cases studied in this dissertation, some implications may be drawn. Client management may intend to prevent knowledge losses by contractually obliging vendors to codify knowledge in extensive documentations. The results of the dissertation suggest a pessimistic expectation on the effectiveness of this strategy. Supportive information in the form of documents is unlikely to compensate for the expertise of humans in at least moderately complex problem-solving domains. Contractually specifying knowledge codification may then rather accelerate than reduce knowledge losses because it risks legitimizing documents as a substitute for personal knowledge. Client managers may rather be advised to implement service level agreements that reduce the loss of personal knowledge. Service levels on turnover in the vendor teams may be an effective strategy to this end. Work samples and other strategies to measure personal knowledge may also be worth being considered in contract design.

ACKNOWLEDGEMENTS

This dissertation has been financially supported by the Swiss National Science Foundation (SNSF) (Grant No. 100018_140407 / 1), by the Mittelbauvereinigung der Universität Bern (MVUB), and by the Berne University Research Foundation. My thanks go to these institutions.

I thank the study participants (the employees of the Swiss bank and of the vendors) for their generous support during data collection. Their willingness to openly share facts and perspectives has greatly helped theory development.

I also thank my co-authors and the following people for their comments and other contributions to this work (in alphabetical order): Julien Chevalley, Thomas Fischer, Inga Glogger, Thomas Huber, Jasmin Kaiser, Dorothy Leidner, Paul Meyer, Simon Milligan, Ilan Oshri, Kiron Ravindran, Daphne Rich, Rajiv Sabherwal, Rolf Schwonke, Kai Spohrer, Khôi Tran, Astrid Wichmann, Cindy Zapata, Han Zhang, and the anonymous reviewers of my papers at the International Conference on Information Systems 2012 and at the Hawaii International Conference on System Sciences 2013.

LITERATURE

Abdel-Hamid, T.K., and Madnick, S.E. 1989. "Lessons Learned from Modeling the Dynamics of Software Development," *Communications of the ACM* (32:12), pp. 1426-1438.

Ackermann, B. 2011. "Structured Knowledge Transfer for a Continuous Organisational Development: Methodology in Theory and Praxis," in: *Proceedings of I-KNOW ’11 and I-SEMANTICS ’11*. Graz.

Alavi, M., and Leidner, D. 2001. "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues," *MIS Quarterly* (25:1), pp. 107-136.

Anderson, S.W., and Dekker, H.C. 2005. "Management Control for Market Transactions: The Relation between Transaction Characteristics, Incomplete Contract Design, and Subsequent Performance," *Management Science* (51:12), pp. 1734-1752.

ATKearney. 2011. "The A.T. Kearney Global Services Location Index," A.T. Kearney, Chicago.

Baddeley, A. 1992. "Working Memory," *Science* (255:5044), pp. 556-559.

Banker, R.D., Datar, S.M., Kemerer, C.F., and Zweig, D. 1993. "Software Complexity and Maintenance Costs," *Communications of the ACM* (36:11), pp. 81-94.

Banker, R.D., Davis, G.B., and Slaughter, S.A. 1998. "Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study," *Management Science* (44:4), pp. 433-450.

Banker, R.D., and Slaughter, S.A. 2000. "The Moderating Effects of Structure on Volatility and Complexity in Software Enhancement," *Information Systems Research* (11:3), pp. 219-240.

Baumeister, R.F., Bratslavsky, E., Muraven, M., and Tice, D.M. 1998. "Ego Depletion: Is the Active Self a Limited Resource?," *Journal of Personality and Social Psychology* (74), pp. 1252-1265.

Bergman, M. 2010. "Hermeneutic Content Analysis: Textual and Audiovisual Analyses within a Mixed Methods Framework," in *Sage Handbook of Mixed Methods in Social & Behavioural Research,* A. Tashakkori and C. Teddlie (eds.). Thousand Oaks, CA: Sage, pp. 379-396.

Bingham, C., and Eisenhardt, K. 2006. "Unveiling the Creation and Content of Strategic Processes: How and What Firms Learn from Heterogeneous Experience," in: *Proceedings of the Academy of Management*.

Blumenberg, S., Wagner, H.T., and Beimborn, D. 2009. "Knowledge Transfer Processes in It Outsourcing Relationships and Their Impact on Shared Knowledge and Outsourcing Performance," *International Journal of Information Management* (29:5), pp. 342-352.

Boh, W.F., Slaughter, S.A., and Espinosa, J.A. 2007. "Learning from Experience in Software Development: A Multilevel Analysis," *Management Science* (53:8), pp. 1315-1331.

Booth, T. 2013. "Here, There and Everywhere -- Special Report: Outsourcing and Offshoring," in: *The Economist*. The Economist Newspaper Limited, pp. 2-16.

Chase, W.G., and Simon, H.A. 1973. "Perception in Chess," *Cognitive Psychology* (4:1), pp. 55-81.

Chen, J., and McQueen, R.J. 2010. "Knowledge Transfer Processes for Different Experience Levels of Knowledge Recipients at an Offshore Technical Support Center," *Information Technology & People* (23:1), pp. 54-79.

Choudhury, V., and Sabherwal, R. 2003. "Portfolios of Control in Outsourced Software Development Projects," *Information Systems Research* (14:3), pp. 291-314.

Chowdhury, S. 2005. "The Role of Affect-and Cognition-Based Trust in Complex Knowledge Sharing," *Journal of Managerial Issues* (17:3), p. 310.

Chua, A., and Pan, S. 2008. "Knowledge Transfer and Organizational Learning in Is Offshore Sourcing," *Omega* (36:2), pp. 267-281.

Clark, R.C., and Mayer, R.E. 2011. *E-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning*. San Francisco, CA: Jossey-Bass/Pfeiffer.

Cohen, W.M., and Levinthal, D.A. 1990. "Absorptive Capacity: A New Perspective on Learning and Innovation," *Administrative Science Quarterly* (35:1), pp. 128-152.

Collins, A., Brown, J.S., and Holum, A. 1991. "Cognitive Apprenticeship: Making Thinking Visible," *American Educator* (15:3), pp. 6-11.

Conner, K., and Prahalad, C. 1996. "A Resource-Based Theory of the Firm: Knowledge Versus Opportunism," *Organization Science* (7:5), pp. 477-501.

D'Eredita, M.A., and Barreto, C. 2006. "How Does Tacit Knowledge Proliferate? An Episode-Based Perspective," *Organization Studies* (27:12), p. 1821.

Daft, R.L., and Lengel, R.H. 1986. "Organizational Information Requirements, Media Richness and Structural Design," *Management Science* (32:5), pp. 554-571.

Darcy, D.P., Kemerer, C.F., Slaughter, S.A., and Tomayko, J.E. 2005. "The Structural Complexity of Software an Experimental Test," *Software Engineering, IEEE Transactions on* (31:11), pp. 982-995.

Dennis, A., Fuller, R., and Valacich, J. 2008. "Media, Tasks, and Communication Processes: A Theory of Media Synchronicity," *MIS Quarterly* (32:3), pp. 575-600.

Dennis, A., and Valacich, J. 1999. "Rethinking Media Richness: Towards a Theory of Media Synchronicity," in: *The 32nd Hawaii International Conference on System Sciences*. Maui, HI.

Dibbern, J., Goles, T., Hirschheim, R., and Jayatilaka, B. 2004. "Information Systems Outsourcing: A Survey and Analysis of the Literature," *ACM SIGMIS Database* (35:4), pp. 6-102.

Dibbern, J., Winkler, J., and Heinzl, A. 2008. "Explaining Variations in Client Extra Costs between Software Projects Offshored to India," *MIS Quarterly* (32:2), pp. 333-366.

Eisenhardt, K. 1989a. "Agency Theory: An Assessment and Review," *Academy of Management Review* (14:1), pp. 57-74.

Eisenhardt, K. 1989b. "Building Theories from Case Study Research," *Academy of Management Review* (14:4), pp. 532-550.

Ericsson, K.A., Krampe, R.T., and Tesch-Römer, C. 1993. "The Role of Deliberate Practice in the Acquisition of Expert Performance," *Psychological Review* (100:3), p. 363.

Espinosa, J.A., Slaughter, S.A., Kraut, R.E., and Herbsleb, J.D. 2007. "Team Knowledge and Coordination in Geographically Distributed Software Development," *Journal of Management Information Systems* (24:1), pp. 135-169.

Forrester, J.W. 1961. *Industrial Dynamics*. Cambridge, MA: MIT Press

Forrester, J.W. 1987. "Lessons from System Dynamics Modeling," *System Dynamics Review* (3:2), pp. 136-149.

Goo, J., Kishore, R., Rao, H., and Nam, K. 2009. "The Role of Service Level Agreements in Relational Management of Information Technology Outsourcing: An Empirical Study," *MIS Quarterly* (33:1), p. 8.

Gopal, A., Sivaramakrishnan, K., Krishnan, M., and Mukhopadhyay, T. 2003. "Contracts in Offshore Software Development: An Empirical Analysis," *Management Science* (49:12), pp. 1671-1683.

Greene, J.C., Caracelli, V.J., and Graham, W.F. 1989. "Toward a Conceptual Framework for Mixed-Method Evaluation Designs," *Educational Evaluation and Policy Analysis* (11:3), pp. 255-274.

Gregory, R., Beck, R., and Prifling, M. 2009. "Breaching the Knowledge Transfer Blockade in It Offshore Outsourcing Projects - a Case from the Financial Services Industry," *The 43th Hawaii International Conference on Systems Sciences*, Big Island, USA.

Gulati, R. 1995. "Does Familiarity Breed Trust? The Implications of Repeated Ties for Contractual Choice in Alliances," *Academy of Management Journal* (38:1), pp. 85-112.

Hinds, P.J., and Bailey, D.E. 2003. "Out of Sight, out of Sync: Understanding Conflict in Distributed Teams," *Organization Science* (16:3), pp. 615-632.

Hofstede, G. 1986. "Cultural Differences in Teaching and Learning," *International Journal of Intercultural Relations* (10:3), pp. 301-320.

Huber, T., Fischer, T., and Dibbern, J. 2011. "Substitutes or Complements? A Framework for Investigating the Dynamic Interplay of Is Outsourcing Governance Mechanisms," in: *32nd International Conference of Information Systems*. Shanghai, China.

Iivari, J., Hirschheim, R., and Klein, H. 2004. "Towards a Distinctive Body of Knowledge for Information Systems Experts: Coding Isd Process Knowledge in Two Is Journals," *Information Systems Journal* (14:4), pp. 313-342.

Jansen, R.C. 1993. "Interval Mapping of Multiple Quantitative Trait Loci," *Genetics* (135:1), pp. 205-211.

Jonassen, D.H. 1997. "Instructional Design Models for Well-Structured and Iii-Structured Problem-Solving Learning Outcomes," *Educational Technology Research and Development* (45:1), pp. 65-94.

Kalyuga, S., Ayres, P., Chandler, P., and Sweller, J. 2003. "The Expertise Reversal Effect," *Educational Psychologist* (38:1), pp. 23-31.

Kankanhalli, A., Pee, L., Tan, G., and Chhatwal, S. 2012. "Interaction of Individual and Social Antecedents of Learning Effectiveness: A Study in the It Research Context," *Engineering Management, IEEE Transactions on* (59:1), pp. 115-128.

Kim, D.H. 1993. "The Link between Individual and Organizational Learning," *Sloan Management Review* (35:1), pp. 37-50.

Kim, Y., Krishnan, R., and Argote, L. 2012. "The Learning Curve of It Knowledge Workers in a Computing Call Center," *Information Systems Research* (23:3), pp. 887-902.

Kirsch, L. 1996. "The Management of Complex Tasks in Organizations: Controlling the Systems Development Process," *Organization Science* (7:1), pp. 1-21.

Kirsch, L. 2004. "Deploying Common Systems Globally: The Dynamics of Control," *Information Systems Research* (15:4), pp. 374-395.

Kirschner, P.A., Sweller, J., and Clark, R.E. 2006. "Why Minimal Guidance During Instruction Does Not Work: An Analysis of the Failure of Constructivist, Discovery, Problem-Based, Experiential, and Inquiry-Based Teaching," *Educational Psychologist* (41:2), pp. 75-86.

Kline, P. 1999. *The Handbook of Psychological Testing*. London: Routledge.

Ko, D., Kirsch, L., and King, W. 2005. "Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations," *MIS Quarterly* (29:1), pp. 59-85.

Kotlarsky, J., and Oshri, I. 2005. "Social Ties, Knowledge Sharing and Successful Collaboration in Globally Distributed System Development Projects," *European Journal of Information Systems* (14:1), pp. 37-48.

Krancher, O., and Dibbern, J. 2012. "Learning Software-Maintenance Tasks in Offshoring Projects: A Cognitive-Load Perspective," in: *The 33rd International Conference on Information Systems*. Orlando, USA.

Krancher, O., and Slaughter, S. 2013. "Governing Individual Learning in the Transition Phase of Software Maintenance Offshoring: A Dynamic Perspective," in: *The 46th Hawaii International Conference on System Sciences*. Maui, HI.

Lacity, M.C., Khan, S., Yan, A., and Willcocks, L.P. 2010. "A Review of the It Outsourcing Empirical Literature and Future Research Directions," *Journal of Information Technology* (25:4), pp. 395-433.

Landis, J.R., and Koch, G.G. 1977. "The Measurement of Observer Agreement for Categorical Data," *Biometrics* (33:1), pp. 159-174.

Langley, A. 1999. "Strategies for Theorizing from Process Data," *The Academy of Management Review* (24:4), pp. 691-710.

Larsen, M.M., Manning, S., and Pedersen, T. 2012. "Uncovering the Hidden Costs of Offshoring: The Interplay of Complexity, Organizational Design, and Experience," *Strategic Management Journal* (34:5), pp. 533–552.

Lee, A.S., and Baskerville, R.L. 2003. "Generalizing Generalizability in Information Systems Research," *Information Systems Research* (14:3), pp. 221-243.

Lee, J., and Kim, Y. 1999. "Effect of Partnership Quality on Is Outsourcing Success: Conceptual Framework and Empirical Validation," *Journal of Management Information Systems* (15:4), pp. 29-61.

Lepine, J.A., and Van Dyne, L. 2001. "Peer Responses to Low Performers: An Attributional Model of Helping in the Context of Groups," *Academy of Management Review* (26:1), pp. 67-84.

Mayer, R.C., Davis, J.H., and Schoorman, F.D. 1995. "An Integrative Model of Organizational Trust," *Academy of Management Review* (20:3), pp. 709-734.

Mayer, R.E. 2003. "The Promise of Multimedia Learning: Using the Same Instructional Design Methods across Different Media," *Learning and Instruction* (13:2), pp. 125-139.

Mayer, R.E., and Moreno, R. 2003. "Nine Ways to Reduce Cognitive Load in Multimedia Learning," *Educational Psychologist* (38:1), pp. 43-52.

McAllister, D.J. 1995. "Affect-and Cognition-Based Trust as Foundations for Interpersonal Cooperation in Organizations," *Academy of Management Journal* (38:1), pp. 24-59.

Merrill, M.D. 2002. "First Principles of Instruction," *Educational Technology Research and Development* (50:3), pp. 43-59.

Miles, M.B., and Huberman, A.M. 1994. *Qualitative Data Analysis: An Expanded Sourcebook*. Thousand Oaks, CA: SAGE.

Mohr, L.B. 1982. *Explaining Organizational Behavior*. San Francisco, CA: Jossey-Bass.

Moos, D.C., and Azevedo, R. 2008. "Self-Regulated Learning with Hypermedia: The Role of Prior Domain Knowledge," *Contemporary Educational Psychology* (33:2), pp. 270-298.

Morrison, R.F., and Brantner, T.M. 1992. "What Enhances or Inhibits Learning a New Job? A Basic Career Issue," *Journal of Applied Psychology* (77:6), p. 926.

Nicholson, B., and Sahay, S. 2004. "Embedded Knowledge and Offshore Software Development," *Information and Organization* (14:4), pp. 329-365.

Nonaka, I. 1994. "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science* (5:1), pp. 14-37.

Oshri, I., Kotlarsky, J., and Willcocks, L.P. 2011. *The Handbook of Global Outsourcing and Offshoring*. Houndmills, United Kingdom: Palgrave MacMillan.

Oshri, I., Van Fenema, P., and Kotlarsky, J. 2008. "Knowledge Transfer in Globally Distributed Teams: The Role of Transactive Memory," *Information Systems Journal* (18:6), pp. 593-616.

Ouchi, W.G. 1979. "A Conceptual Framework for the Design of Organizational Control Mechanisms," *Management Science*), pp. 833-848.

Ozcinar, Z. 2009. "The Topic of Instructional Design in Research Journals: A Citation Analysis for the Years 1980-2008," *The Australasian Journal of Educational Technology* (25:4), pp. 559-580.

Paas, F., Tuovinen, J.E., Tabbers, H., and Van Gerven, P.W.M. 2003. "Cognitive Load Measurement as a Means to Advance Cognitive Load Theory," *Educational Psychologist* (38:1), pp. 63-71.

Pennington, N. 1987. "Stimulus Structures and Mental Representations in Expert Comprehension of Computer Programs," *Cognitive Psychology* (19:3), pp. 295-341.

Pintrich, P.R. 2004. "A Conceptual Framework for Assessing Motivation and Self-Regulated Learning in College Students," *Educational Psychology Review* (16:4), pp. 385-407.

Plano Clark, V.L., and Badiee, M. 2010. "Research Questions in Mixed Methods Research," in *Sage Handbook of Mixed Methods in Social and Behavioral Research,* A. Tashakkori and C. Teddlie (eds.). Thousand Oaks, CA: Sage, pp. 275-304.

Plass, J.L., Moreno, R., and Brünken, R. 2010. *Cognitive Load Theory*. Cambridge, United Kingdom: Cambridge Univ Press.

Polanyi, M. 1962. *Personal Knowledge. Towards a Post-Critical Philosophy*. Chicago, IL: The University of Chicago Press.

Poppo, L., and Zenger, T. 2002. "Do Formal Contracts and Relational Governance Function as Substitutes or Complements?," *Strategic management journal* (23:8), pp. 707-725.

Purdie, N., and Hattie, J. 1996. "Cultural Differences in the Use of Strategies for Self-Regulated Learning," *American Educational Research Journal* (33:4), pp. 845-871.

Renkl, A., and Atkinson, R.K. 2003. "Structuring the Transition from Example Study to Problem Solving in Cognitive Skill Acquisition: A Cognitive Load Perspective," *Educational Psychologist* (38:1), pp. 15-22.

Renkl, A., Mandl, H., and Gruber, H. 1996. "Inert Knowledge: Analyses and Remedies," *Educational Psychologist* (31:2), pp. 115-121.

Rustagi, S., King, W.R., and Kirsch, L.J. 2008. "Predictors of Formal Control Usage in It Outsourcing Partnerships," *Information Systems Research* (19:2), pp. 126-143.

Sawicka, A. 2008. "Dynamics of Cognitive Load Theory: A Model-Based Approach," *Computers in Human Behavior* (24:3), pp. 1041-1066.

Schnotz, W., and Kürschner, C. 2007. "A Reconsideration of Cognitive Load Theory," *Educational Psychology Review* (19:4), pp. 469-508.

Singh, P.V., Tan, Y., and Youn, N. 2011. "A Hidden Markov Model of Developer Learning Dynamics in Open Source Software Projects," *Information Systems Research* (22:4), pp. 790-807.

Slaughter, S., and Kirsch, L. 2006. "The Effectiveness of Knowledge Transfer Portfolios in Software Process Improvement: A Field Study," *Information Systems Research* (17:3), p. 301.

Sweller, J. 1994. "Cognitive Load Theory, Learning Difficulty, and Instructional Design," *Learning and Instruction* (4:4), pp. 295-312.

Sweller, J., and Chandler, P. 1994. "Why Some Material Is Difficult to Learn," *Cognition and Instruction* (12:3), pp. 185-233.

Sweller, J., Van Merrienboer, J.J.G., and Paas, F.G.W.C. 1998. "Cognitive Architecture and Instructional Design," *Educational Psychology Review* (10:3), pp. 251-296.

Sydow, J., and Windeler, A. 2003. "Knowledge, Trust, and Control: Managing Tensions and Contradictions in a Regional Network of Service Firms," *International Studies of Management and Organization* (33:2), pp. 69-100.

Szulanski, G. 1996. "Exploring Internal Stickiness: Impediments to the Transfer of Best Practice within the Firm," *Strategic Management Journal* (17:1), pp. 27-43.

Szulanski, G. 2000. "The Process of Knowledge Transfer: A Diachronic Analysis of Stickiness," *Organizational Behavior and Human Decision Processes* (82:1), pp. 9-27.

Tashakkori, A., and Teddlie, C. 1998. *Mixed Methodology: Combining Qualitative and Quantitative Approaches*. Thousand Oaks, CA: Sage.

Teddlie, C., and Tashakkori, A. 2010. "Overview of Contemporary Issues in Mixed Methods Research," in *Sage Handbook of Mixed Methods in Social and Behavioral Research,* A. Tashakkori and C. Teddlie (eds.). Thousand Oaks, CA: Sage, pp. 1-41.

Tiwana, A., and Keil, M. 2007. "Does Peripheral Knowledge Complement Control? An Empirical Test in Technology Outsourcing Alliances," *Strategic Management Journal* (28:6), pp. 623-634.

Tiwana, A., and Keil, M. 2009. "Control in Internal and Outsourced Software Projects," *Journal of Management Information Systems* (26:3), pp. 9-44.

Tiwari, V. 2009. "Transition During Offshore Outsourcing: A Process Model," *The 30th International Conference of Information Systems*, Phoenix, AZ.

Van de Ven, A.H., and Poole, M.S. 1995. "Explaining Development and Change in Organizations," *Academy of Management Review* (20:3), pp. 510-540.

Van Merriënboer, J.J.G., Clark, R.E., and De Croock, M.B.M. 2002. "Blueprints for Complex Learning: The 4c/Id-Model," *Educational Technology Research and Development* (50:2), pp. 39-64.

Van Merriënboer, J.J.G., and Kirschner, P.A. 2007. *Ten Steps to Complex Learning: A Systematic Approach to Four-Component Instructional Design*. Mahwah, NJ: Lawrence Erlbaum.

Van Merriënboer, J.J.G., Kirschner, P.A., and Kester, L. 2003. "Taking the Load Off a Learner's Mind: Instructional Design for Complex Learning," *Educational Psychologist* (38:1), pp. 5-13.

Van Merriënboer, J.J.G., and Sweller, J. 2005. "Cognitive Load Theory and Complex Learning: Recent Developments and Future Directions," *Educational Psychology Review* (17:2), pp. 147-177.

Vlaar, P., van Fenema, P., and Tiwari, V. 2008. "Cocreating Understanding and Value in Distributed Work: How Members of Onsite and Offshore Vendor Teams Give, Make, Demand, and Break Sense," *MIS Quarterly* (32:2), p. 5.

Von Mayrhauser, A., and Vans, A.M. 1995. "Program Comprehension During Software Maintenance and Evolution," *Computer* (28:8), pp. 44-55.

Weiner, B. 1985. "An Attributional Theory of Achievement Motivation and Emotion," *Psychological Review* (92:4), p. 548.

Wende, E., and Philip, T. 2011. "Instant Messenger in Offshore Outsourced Software Development Projects: Experiences from a Case Study," *The 44th Hawaii International Conference on System Sciences*, Koloa, HI.

Wende, E., Schwabe, G., and Philip, T. 2010. "Exploring the Media Mix During It-Offshore Transition Phase – an Opportunity for a Better Knowledge Transfer," in: *The Fourth Global Sourcing Workshop*. Zermatt, Switzerland.

Westner, M., and Strahringer, S. 2010. "Determinants of Success in Is Offshoring Projects: Results from an Empirical Study of German Companies," *Information & Management* (47:5), pp. 291-299.

Williams, C. 2011. "Client–Vendor Knowledge Transfer in Is Offshore Outsourcing: Insights from a Survey of Indian Software Engineers," *Information Systems Journal* (21:4), pp. 335-356.

Winkler, J., Dibbern, J., and Heinzl, A. 2008. "The Impact of Cultural Differences in Offshore Outsourcing—Case Study Results from German–Indian Application Development Projects," *Information Systems Frontiers* (10:2), pp. 243-258.

Wood, R.E. 1986. "Task Complexity: Definition of the Construct," *Organizational Behavior and Human Decision Processes* (37:1), pp. 60-82.

Woolthuis, R.K., Hillebrand, B., and Nooteboom, B. 2005. "Trust, Contract and Relationship Development," *Organization Studies* (26:6), pp. 813-840.

Yin, R.K. 2009. *Case Study Research: Design and Methods*. Thousand Oaks, CA: Sage.

STATEMENT OF AUTHORSHIP

Ich erkläre hiermit, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Koautorenschaften sowie alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe o des Gesetzes vom 5. September 1996 über die Universität zum Entzug des aufgrund dieser Arbeit verliehenen Titels berechtigt ist.

Bern, 30. April 2013 Oliver Krancher