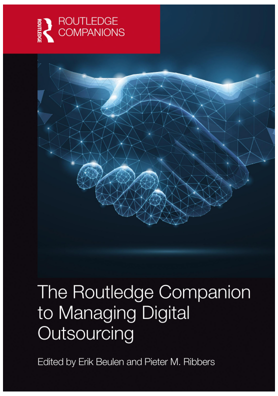

The Routledge Companion to Managing Digital Outsourcing

Erik Beulen, Pieter M. Ribbers

Knowledge transfer in the transition phase

Publication details
https://test.routledgehandbooks.com/doi/10.4324/9781351037785-13
Oliver Krancher
Published online on: 28 Jul 2020

11

KNOWLEDGE TRANSFER IN THE TRANSITION PHASE

Review of the literature and implications for digital business strategy and agility

Oliver Krancher

11.1 Introduction

Any outsourcing or offshoring project initially goes through a so-called transition phase, i.e., the phase during which the service is handed over to a new service delivery unit (SDU), which can be a vendor or a captive centre [1,2]. The key challenge in transitions is to enable the new SDU to take over the service as quickly as possible. This requires the transfer of critical resources, including hardware [3], software [4], and knowledge [5]. In many cases, the most difficult of these transfers is the transfer of knowledge, or knowledge transfer (KT). While KT was rather unproblematic in earlier domestic outsourcing arrangements in which clients transferred their IT staff, along with their knowledge, to vendors [6], staff transfer is not an option anymore in the age of offshoring. Engineers will not accept being relocated to another country while earning a fraction of their former salaries. Thus, in many contemporary sourcing arrangements, transitions involve a KT process during which the new SDU acquires knowledge from the incumbent SDU, i.e., the unit that provided the service thus far.

KT in the transition phase is one of the most critical [7] and least understood [8] aspects of sourcing arrangements. Managers often highly underestimate the efforts and challenges associated with KT in transitions, assuming that KT resembles other asset transfer processes [8,9]. It is not uncommon that managers plan transition phases of a few weeks [1,9] although managers experienced with transitions concede that it may take, in some instances, five years until the new SDU has acquired enough knowledge to display the performance levels of the incumbent SDU [9,10]. It is thus of little surprise that unanticipated costs for KT during transition are a principal reason for offshoring failures [11]. The critical nature of KT during transition is further elevated by the fact that transitions are no one-off in the life of a client. Many clients are engaging in their second- and third-generation outsourcing projects [2,12]. These organizations need to manage transitions each time they switch a vendor. The difficulty of switching vendors, rooted in the problematic nature of KT, is widely acknowledged in practice and theory. Willcocks and Lacity conclude that "the single most threatening aspect of outsourcing is the substantial switching costs" [13,4]. The most popular theoretical perspectives in outsourcing research, transaction cost economics [14], and the knowledge-based view [15] also emphasize the problematic nature of transferring knowledge (or human assets) to the new SDU. Hence, a thorough understanding of KT during transitions is essential both for practice and theory of outsourcing and offshoring.

The critical nature of KT during transitions has recently given rise to a vibrant stream of research on the topic. In light of this quickly increasing body of work, the goal of this chapter is twofold. First, it summarizes the current state of research about KT in the transition phase. Second, it theorizes how the two recent developments that are in the focus of this book, digital business strategies [16,17] and the increasingly important role of agility [18], are related to KT during transitions.

The chapter proceeds follows. It first summarizes two foundational literature streams on which research on KT during transitions draws heavily: the literatures on KT and expertise. It then provides a review of 35 published studies on KT in the transitions phase. The review focuses on four themes: (1) types of knowledge, (2) the mechanisms through which knowledge is transferred, (3) contextual factors that affect KT, and (4) the management of KT. The key findings from the review are (1) that knowledge about the software application is often the most critical knowledge, (2) that learning tasks are often the most effective KT mechanism, (3) that the knowledge recipient's expertise and distances of various types are important context factors, and (4) that these contextual factors affect the need for management efforts by the client. Building on the findings from the review, I discuss implications for KT during transition in the age of digital business strategies and the quest for agility. I present three specific recommendations: (1) cultivate embedded, concurrent-sourcing relationships, (2) organize KT around learning tasks and feedback, and (3) reduce the need for learning by hiding complexity. The discussion suggests that the effective execution and management of transitions will be vital for organizations that strive to leverage digital business strategies and agility while doing outsourcing or offshoring.

11.2 The foundations: the literatures on KT and expertise

Although research on KT in the transition phase draws on a variety of perspectives, two have been particularly influential: the broader literature on KT and the literature on expertise. This section provides brief summaries of these two literature streams.

11.2.1 The KT literature

With the increasing interest in knowledge-based perspectives on organizations since the 1990s [19,20], a substantial body of literature on KT has emerged [21,23]. Although this literature is not concerned with the specifics of the transition phase in outsourcing or offshoring projects, it has served as a foundation for much of the research on the transition phase.

Among the most influential studies are those by Garbriel Szulanski and Wenpin Tsai. Szulanski examined the transfer of best practices between different organizational units within a firm. He found that although the literature at the time emphasized motivational factors (i.e., the source's willingness to share knowledge and the recipient's willingness to accept knowledge), these factors hardly correlated with KT outcomes. Of much greater importance were knowledge-related factors, in particular the recipient's *absorptive capacity* (i.e., their ability to assimilate and apply new knowledge to commercial ends), *causal ambiguity* (i.e., uncertainty about the factors of production and their interaction), and *arduous relationships* (i.e., the extent to which the source-recipient relationship is laborious and distant) [21]. Szulanski also demonstrated that KT is a long process, rather than a one-off event, and that different factors come into play at different stages of the process [24]. The recipient's absorptive capacity was by far the most important factor during those stages (implementing, ramp-up, and integrating) in which the recipient is supposed to acquire knowledge. A key concept in Szulanski's arguments, as well as in the broader KT literature, is *tacit knowledge*, i.e., knowledge that is rooted in action and that cannot be transferred by communication [25]. Tacit knowledge contributes to causal ambiguity and it makes individual exchanges in embedded relationships important. Tsai [22] further delved into the role of relational factors. He demonstrated the importance of the *network position* that an organizational unit occupies within the organization. The more central the network position, the more does the unit benefit from KT, in particular if the absorptive capacity of the unit is high.

Subsequent studies on KT corroborated the importance of relational and knowledge-based characteristics. In a meta-analysis of research on KT, Van Wijk and Jansen [26] found that the factors with the highest correlations with KT outcomes were relational factors (trust, tie strength, a centralized network position, shared vision, cultural distance, and the number of relationships) and the knowledge-related factors of causal ambiguity and the recipient's absorptive capacity.

Although these findings provided important foundations for research on the transition phase in outsourcing and offshoring, two boundary conditions of the KT literature should be kept in mind when transferring its insights to outsourcing or offshoring settings. First, with few exceptions, the KT literature focuses on the collective knowledge that one organizational unit (or organization) possesses and that another organizational unit (or organization) attempts to acquire. Examples of *collective knowledge* include organizational routines and shared language [20,27]. Yet, without empirical examinations of transitions, it is not clear whether the most critical process in transitions is the transfer of collective knowledge or of individual knowledge, such as individual cognitive schemas of application domains and applications.

Second, the recipient in the KT literature is typically an organizational unit that, while attempting to acquire knowledge from another unit, is already competent in the execution of its task. It seeks additional knowledge in order to improve its performance, rather than in order to be able to perform its task in the first place. For instance, Szulanski examined how well a recipient unit was able to incorporate a specific best practice (a specific organizational routine) from another unit into its existing operations, not how a newly created recipient unit learned to perform its basic mission. In contrast to the KT literature, the literature on expertise shows greater interest in the individual level of analysis and in the initial acquisition of competence.

11.2.2 The expertise literature

Long before the recent interest in KT in organizational studies, the expertise literature has set out to explore the nature and the acquisition of expertise. At the heart of this research are the questions of *why* particular individuals excel in their performances of particular tasks (i.e., the nature of expertise) and *how* their superior ability has come about (i.e., the process of expertise acquisition).

Substantial insights into the organization of expertise came from the seminal work by William Chase and Hebert Simon on perception in chess [28]. They found that expert chess players did not outperform novice chess players in the task of memorizing and reproducing randomly created chess positions, while they strongly outperformed novices if the chess positions exhibited a typical pattern (i.e., a pattern that can occur in real chess plays). From this observation, Chase and Simon concluded that expertise does not result from superior general memory abilities but from *domain-specific cognitive schemas* that allow chunking information to familiar higher-order units. In the context of chess, these cognitive schemas include typical patterns of chess positions that chess masters learned over thousands of matches. Elaborating on these ideas, subsequent research has established that expertise resides in cognitive schemas that are (1) *highly organized* (i.e., many linkages between concepts), (2) *highly automated* (i.e., experts rely on the schemas in an effortless, immediate, intuitive way), and (3) *domain specific* (i.e., they enable superior performance only in a relatively narrow domain). For instance, illustrating organization, Adelson [29] found that novice programmers comprehend source code based on the syntax expressed in individual lines of codes, whereas expert programmers relate lines of codes into a hierarchy of concepts. In a similar vein, Doane et al. [30] found that UNIX experts had much greater knowledge of the relationships between UNIX structures than novices. This knowledge allowed UNIX experts to perform complex commands (composite commands), which novices failed to perform although they had been instructed on them. These examples show that experts differ from novices in the organization of their cognitive schemas: The experts' schemas include more relationships and more powerful higher-order concepts, which allows experts to aggregate information to higher-order chunks. Expert knowledge is not only well organized, but also highly automated. Chess grandmasters can devise excellent moves within few seconds, and, in simultaneous chess, they can defeat dozens of less experienced players at the same time. Illustrating the domain specificity of expertise, Voss et al. [31] found that expert chemists are essentially novices in solving political science problems, much like Chase and Simon's chess players are novices in the task of recalling randomized chess positions.

How do experts acquire their well-organized, automated cognitive schemas that enable them to quickly assimilate new information and solve complex problems within a domain? Expertise research concurs that it takes *long periods of deliberate practice* to become an expert [32,33]. Practice implies that learners need to perform the task in which they want to become experts in order to become more proficient. Deliberation means that learners need to reflect about their practice and to be coached about ways to improve it [32,33]. Long periods of practice imply that it typically takes 10 years, or 10,000 hours, of deliberate practice in a particular domain in order to become expert in it [32]. While practice, deliberation, and coaching are essential for acquiring expertise according to the expertise literature, codified information is not. As Ericsson and colleagues put it: "Knowledge management systems rarely, if ever, deal with what psychologists call knowledge" [34,3]. While most of the expertise literature has put greater emphasis on advanced skill acquisition (transition from proficient to expert level) than on early skill acquisition, research on cognitive load theory extends expertise research to the process of early skill acquisition. This research recommends scaffolded practice, i.e., learning based on simplified task tasks that match the learner's expertise level [35,36]. Scaffolded practice allows learners to engage in practice tasks while it reduces the risk of cognitive overload.

Table 11.1 compares key aspects of the KT and the expertise literatures. The literatures focus on different knowledge-acquiring (or learning) entities. While, with few exceptions [23], the KT literature focuses on organizational units that are already proficient in their task, the expertise literature focuses on individuals at various expertise levels. The literatures also differ in their conceptualizations of knowledge. While the KT literature views knowledge as pieces of explicit and tacit knowledge that entities may or may not possess, the expertise literature views knowledge as personal domain-specific cognitive schemas that vary in organization and automation. Both literature streams have contributed important knowledge. While the KT literature emphasizes context factors (e.g., absorptive capacity), the expertise literature emphasizes the mechanisms through which knowledge is acquired.

Table 11.1 Summary of the literatures on KT and expertise

	KT literature	*Expertise literature*
Entity that acquires knowledge	Organizational unit already proficient in its task	Individual at various expertise levels
Nature of knowledge	Explicit and tacit knowledge	Domain-specific cognitive schemas that vary in organization and automation
Key findings	The extent to which a unit acquires knowledge from another unit strongly depends on knowledge-related factors (e.g., absorptive capacity) and relational factors (e.g., network centrality).	Individuals acquire expertise through years of domain-specific deliberate practice, initially supplemented by scaffolding.

11.3 What we know about KT in the transition phase: a review of the literature

This section provides a review of the literature on KT in the transition phase in information systems (IS) research. Since the essence of transitions is to enable a new service delivery unit (SDU) to provide its service, the review is limited to studies that examine unidirectional KT to a new SDU (which may be a vendor or a captive centre). This excludes papers that examine KT from an SDU to the client [37,38] and papers that examine bidirectional knowledge flows [6,39,40]. Studies are included in the review if they give insights into the KT to a new SDU, even if the terms transition or KT are not used. Since the focus in this book lies on IS work, we included only studies that refer to IS tasks, such as application development, application maintenance, technical support, or infrastructure operations. Thirty-five papers met these criteria and were included into the review. All topic areas that are in the focus of this review are well represented in this literature sample, with most work focusing on KT management (23 studies), followed by KT mechanisms (19 studies), context factors (17 studies), and knowledge categories (14 studies).

11.3.1 Knowledge categories

The literature provides substantial insights into the types of knowledge that are transferred during transitions. Table 11.2 shows important knowledge categories, their definitions, and key findings related to the categories. While the first four categories refer to different types of knowledge, the subsequent categories refer to different domains of knowledge.

Individual vs. Collective Knowledge. Not surprisingly, research on transitions draws heavily on the different types of knowledge that are distinguished in the broader KT or outsourcing literature. A number of studies mention the distinction between individual and collective knowledge (see Table 11.2 for definitions). There is some agreement that both individual and collective knowledge matter in transitions [2,41,42]. For instance, at an individual level, engineers in the new SDU have to learn the structure of the existing software applications [1]. At a collective level, the new SDU builds up transactive memory systems (i.e., distributed knowledge about who knows what) [43]. Moreover, the client and the new SDU build up new collaboration processes [1]. While studies emphasize individual and collective knowledge, there is little research that systematically compares the criticality of individual with collective knowledge in transitions. Thus, it is unclear whether KT during transitions is often difficult because of challenges related to the acquisition of individual

Table 11.2 Findings related to types of knowledge

Knowledge categories	*Definitions (examples in parentheses)*	*Key findings*
Individual vs. collective	• Individual knowledge: Knowledge possessed by individuals (e.g., programming skills, an engineer's mental models about a particular software application) • Collective knowledge: Knowledge possessed by collective entities such as teams or organizations (e.g., work routines, transactive memory systems, shared knowledge) [27]	Transitions involve the acquisition of both individual knowledge (e.g., individuals learning to perform the outsourced task) and collective knowledge (e.g., units adjusting their routines to the new set-up) [2,41–43].
Explicit vs. tacit	• Explicit knowledge: Knowledge that can be communicated without losing its value (e.g., passwords) • Tacit knowledge: Knowledge that would lose its value when being communicated (e.g., programming skills, mental models about a particular software application) [25]	Transitions involve the transfer of both explicit and tacit knowledge [5,11,41–45].
Embedded vs. non-embedded knowledge	• Embedded knowledge: Knowledge that is anchored in local (e.g., societal, organizational) meanings (e.g., the British tax system) • Non-embedded knowledge: Knowledge that can be interpreted irrespective of local meanings (e.g., meaning of crash test simulation models) [46,47]	Although the transfer of embedded knowledge is often challenging [46,48,49], the transfer of non-embedded knowledge may be equally difficult [47].
Specific vs. generic knowledge	• Specific knowledge: Knowledge that loses much of its value when being redeployed to another client (e.g., knowledge about a custom-built software application) • Generic knowledge: Knowledge that can be redeployed to other clients without significant loss of value (e.g., programming skills) [14]	Although the transfer of client-specific knowledge demands substantial involvement from the client and often causes cost overruns [11,50], the transfer of generic knowledge can be equally challenging and involve significant efforts [47].
Business knowledge (or application domain knowledge)	Knowledge about the application domain for which an information system is built (e.g., knowledge about the client's business) [51]	Especially in application development projects, the transfer of business knowledge is difficult given that business knowledge is often highly embedded in the client's local context [8–10,48,52–57].

Knowledge categories	*Definitions (examples in parentheses)*	*Key findings*
Technical knowledge	Knowledge about hardware and software programming [51]	In some transitions, technical knowledge plays a relatively unimportant role because the engineers in the new SDU already possess the required technical knowledge [5,8,11,49,52,57]. In other transitions, the engineers in the new SDU lack technical knowledge [47,54].
Application knowledge	Knowledge about the software application, its structure, functionality, and behaviour [51]	Especially in software maintenance projects, application knowledge tends to be the most difficult knowledge to transfer [5,10,11,57,58].
Process knowledge	Knowledge about tools, techniques, methods, approaches, and principles used in software development [51]	Process knowledge can be critical [5], in particular if the new SDU lacks experience with business software development [49]. In other projects, process knowledge was found to be less critical [8,57]. However, the client and the new SDU may need to develop new processes for their collaboration [1].

knowledge (e.g., engineers struggle to acquire mental models of applications) or of collective knowledge (e.g., clients and vendors struggle to fine-tune their collaborative processes). In other words, it is unclear whether KT during transitions is primarily an individual learning problem or a collective learning problem.

Explicit vs. Tacit Knowledge. As Table 11.2 shows, many studies evoke the dichotomy of explicit and tacit knowledge that goes back to Polanyi [25]. Interestingly, this dichotomy is more salient in the theoretical arguments than in empirical investigations. Few studies have attempted to measure the amount of explicit versus tacit knowledge that is transferred or needs to be transferred during transitions.

Embedded vs. Non-Embedded Knowledge. Interpretive research on transitions emphasizes the challenges that arise due to the embedded nature of knowledge [46,48,49]. For instance, in one project with an English client, Indian engineers struggled to understand the business logic of a software package that was supposed to support the administration of social security benefits because the Indian engineers were not familiar with the concept of a State-supported welfare system [46]. Interestingly, however, Leonardi and Bailey [47] presented a case in which non-embedded knowledge was very difficult to transfer. They showed that requests sent from two different onshore sites (the United States and Mexico) were highly similar in the nature of knowledge they required, suggesting that the needed knowledge was not of an embedded nature. Yet offshore engineers required substantial help in order to be able to interpret the requests and work on them.

Specific vs. Generic Knowledge. Relatively few studies focus on the distinction between specific and generic knowledge. Dibbern, Winkler, and Heinzl found that transitions suffer from greater extra costs if the required knowledge was highly specific to the client [11]. In a similar vein, Deng and Mao found that the acquisition of client-specific knowledge ("learning about client" in their terms) affected project costs [50]. While specific knowledge may therefore be difficult to transfer, Leonardi and Bailey [47] showed that even the transfer of generic ("occupational" in their terms) knowledge can involve substantial costs. The relatively weak attention to specific versus generic knowledge in the literature is somewhat surprising given that specificity is a key construct of transaction cost economics [59] and of the knowledge-based view [15], theories that are popular in outsourcing and offshoring research.

Domains of Knowledge. A key question in many studies is what domains of knowledge are most salient in the KT process. While many studies build on the dichotomy of business and technical knowledge advocated in particular by Amrit Tiwana's early research [60,61], recent work has relied on more complex taxonomies that also distinguish application knowledge and process knowledge [5,51,57,58]. Research relying on the classic distinction between business and technical knowledge points to the key role of *business knowledge*, in particular in application development projects [48,52,53]. Business knowledge is often specific to the client. Engineers in the new SDU frequently lack business knowledge at the outset of transitions and, hence, struggle to understand requirements and their context [54]. In contrast, *technical knowledge* is typically generic. For example, skills in a programming language are typically applicable to many clients. Since engineers can leverage generic technical knowledge acquired in prior projects, problems related to technical knowledge are often much less salient during transitions [5,49,57], although the transfer of technical knowledge can be an issue in projects that involve many junior engineers in the new SDU [47,54].

Although the business-technical dichotomy has been influential in research on transitions, a number of studies reveal that especially in application maintenance projects, *application knowledge* is often the most difficult knowledge to transfer [5,57,58]. It is well known from software maintenance research that maintainers need to acquire highly elaborated cognitive schemas of the structure of the particular software application in order to be able to maintain the software. The process of acquiring these schemas may require years of application-specific practice [62,63]. Not surprisingly, the acquisition of application knowledge can be a key challenge during transitions, which are typically planned to take weeks or months. In line with these ideas, a multiple-case study by Krancher and Dibbern [57] showed that application knowledge was the knowledge domain that was most salient in interviews, that best predicted cognitive load on vendor engineers, and that engineers found most difficult to transfer. *Process knowledge* may also be important to transfer, in particular when the new SDU lacks experience in business application development. For example, Levina and Vaast [49] reported that teams in the Russian captive centre were initially unaware of structured deployment processes, although the teams were able to learn these processes after some time. In another study, Chua and Pan [5] interpreted that the lack of process knowledge was a reason for having to retain experienced onshore staff. Conversely, Krancher and Dibbern [57] reported that in the software maintenance outsourcing cases studied, the engineers in the new SDU found it straightforward to adjust to the clients' maintenance processes.

In sum, research on knowledge domains concurs that business knowledge and application knowledge are often difficult to transfer during transition, while technical and process knowledge are often rather unproblematic, unless in cases of inexperienced SDUs and engineers.

11.3.2 KT mechanisms

The literature gives insights into a large variety of KT mechanisms, i.e., of activities aimed at helping the new SDU to acquire knowledge from the incumbent SDU. Table 11.3 provides an overview of the mechanisms along with definitions and key findings related to each mechanism.

Table 11.3 Findings related to KT mechanisms

Context factors	*Definitions*	*Key findings*
Knowledge recipient's expertise/absorptive capacity	• Knowledge recipient's expertise: The power of domain-specific cognitive schemas in the knowledge recipients' long-term memory [67] • Knowledge recipient's absorptive capacity: The knowledge recipient's ability to utilize outside knowledge [11]	The knowledge recipient's expertise or absorptive capacity affects KT efforts [9,11,54], the choice and effectiveness of particular KT mechanisms [41,42,53,67], and the extent to which knowledge recipients can self-manage the KT process [70].
Experts' availability and motivation	• Experts' availability: The extent to which experts have free capacity to support the KT • Experts' motivation: The experts' willingness to contribute to KT	Expert's availability and their motivation to share knowledge are critical for KT [1,2,9,52,58,70,71].
Distance (cultural, geographic, semantic, temporal, status, cognitive)	• Cultural distance: Differences in shared values, norms, beliefs, and assumptions between groups (such as nations or organizations) [11] • Geographic distance: The spatial distance between experts and knowledge recipients [11] • Semantic distance: The extent of language barriers between experts and knowledge recipients [11] • Temporal distance: The time-zone difference between experts and knowledge recipients [49] • Status distance: The degree to which knowledge recipients and experts differ in deference and respect for each other according to shared beliefs • Cognitive distance: Differences in taken-for-granted beliefs, perspectives, and mental models at the individual level [8,46]	Although distance of various types often introduces boundaries and thereby hampers KT [48,49,56], distance is less problematic at high levels of knowledge recipient's expertise [11,41,42,70] and at low levels of task complexity [44].
Social capital	Resources embedded in the relationship between experts and knowledge recipients. Resources include network ties, trust, norms, identity, and knowledge [66,72].	Social capital affects the experts' ability and willingness to share knowledge and the knowledge recipients' willingness to ask questions [66].
Turnover	The frequency at which engineers leave the SDU unit	Since high turnover often requires redoing the KT [11] and reduces experts' motivation to share knowledge [66], client managers should take actions to reduce turnover, such as through contractual governance and through socialization efforts [58].
Task complexity	The extent to which a task involves many elements, to which the relationships between these elements are sophisticated, and to which changes in the world interfere with the task [67]	Higher task complexity requires higher amounts of direction in order to reduce cognitive load to a manageable level [67]. High task complexity also makes transitions riskier [9].

Formal Information Sharing. Formal information-sharing mechanisms are planned mechanisms that convey information about a knowledge domain to recipients. The most prominent formal information-sharing mechanisms are documents and presentations. There is a strong emphasis on formal information-sharing mechanisms in many transitions, in particular at early stages of transitions [1,5,44,52]. Some evidence suggests that documents and presentations can indeed help convey useful knowledge about the location of expertise [43,65] and about the client [64], in particular if they incorporate stories (events, facts, and experience in the context of specific situations) [45]. Formal information-sharing mechanisms are appealing mechanisms because they promise the efficient broadcasting of knowledge to many recipients (such as the attendants of a presentation or the readers of a document) [53]. Moreover, especially documents are appealing because they promise the ability to capture knowledge, making it available beyond the point when the personnel of the incumbent SDU leaves [68]. Despite such positive expectations, longitudinal case studies that examined at which point new SDUs were able to take over their task (the ultimate goal of transitions) shed a rather negative light on the potential offered by formal information sharing. A number of case studies show that documents and presentations rarely enable engineers to take over the tasks that clients expect them to take over [5,10,11,47,52,53,57,67]. For instance, Krancher and Dibbern made the following observation about software maintenance transitions:

> [The engineers in the new SDU] initially struggled to make sense of documents. They were able to understand many documents only after they had worked on related tasks. Put differently, documents did not enable them to solve tasks, but solving tasks enabled them to understand documents.
>
> *[57, p. 4412]*

Such a rather pessimistic perspective on the potential offered by documents and presentations is in line with the concept of tacit knowledge. A fundamental idea behind the concept of tacit knowledge is that knowledge that can be communicated (e.g., through documents and presentations) is only a relatively minor facet of the knowledge that enables competent human action [25,69].

Socialization. Many transition projects rely strongly on socialization mechanisms, including site visits, teleconferencing, and instant messaging [58,65,66]. Although the managers in some projects initially fail to anticipate the need for socialization, they often increase the use of these mechanisms after observing initial disappointing work outcomes [11,48]. Socialization mechanisms stimulate informal communication, which offers a number of potential benefits: helping develop transactive memory systems [65], allowing to rapidly clarify queries [48,66], helping the engineers in the new SDU to deepen their knowledge about the client [64], and helping the client to gain a better understanding of the needs of the new SDU [66]. In a survey study, Williams found support for the importance of socialization by showing that client embedment, a construct highly related to socialization, predicts KT outcomes [64]. Socialization efforts need some time to become effective given that engineers in the new SDU initially often hesitate to request help through informal communication channels [58,66,70].

Feedback. Some transitions rely on feedback mechanisms such as quizzes and playback [5,58]. These mechanisms are feedback mechanisms because they provide the people involved in KT with information about the outcomes of KT activities (e.g., learning). Feedback mechanisms help not only to judge the effectiveness of KT but also to discover and to align different understandings [5,58]. Moreover, feedback mechanisms stimulate communication by inviting engineers in the new SDU to ask questions [58]. Stimulating communication may be necessary in particular in offshoring projects that involve countries with high power distance, such as India, where engineers initially often hesitate to ask questions [58,67].

Learning Tasks. Among the most important mechanisms in transitions are learning tasks, i.e., authentic tasks from the domain in which knowledge shall be transferred [67]. Learning tasks include on-the-job training, where the engineers of the new SDU work on real tasks or observe engineers of the former SDU perform their task [5], and support simulation, where the engineers of the new SDU work on tasks that occurred in the past [5]. Learning tasks trigger active cognitive processing in the domain of the particular task, helping engineers to acquire powerful mental models of the task domain [67]. In many case studies, the engineers of the new SDU were able to take over work only after they had engaged in a number of learning tasks in the domain [5,10,11,47,52,53,57,67]. This is consistent with the expertise literature, which suggests that people acquire competence mostly through the active cognitive engagement that is stimulated by practice in the task domain [32].

Direction. Although learning tasks are essential during transitions, they risk cognitively overloading knowledge recipients [11,67]. An effective, and often necessary, strategy for avoiding cognitive overload is direction, i.e., providing knowledge recipients with hints on how to solve a particular learning task. Although studies have used different terms such as direction [67], specification [11], and defining requirements [47], the key commonality among these notions is the principle that experts scaffold the problem-solving process that knowledge recipients encounter during the work on learning tasks by providing hints on how the particular task can be solved. Direction is different from information-sharing mechanisms in that direction is specific to a particular tasks (e.g., the list of changes in a database that are required for a particular change request) whereas information sharing refers to information of a knowledge domain (e.g., information about the structure of the database and the meaning of fields in the database).

To summarize, although many transitions heavily rely on formal information-sharing mechanisms such as documents, case study evidence suggests that learning tasks and direction are critical to enable the SDU to take over work.

11.3.3 Context factors

The literature points to a variety of context factors that affect KT. Table 11.4 provides an overview of context factors along with definitions and key findings. Context factors make KT easier or more difficult, they affect which KT mechanisms are appropriate, or they engender greater need for client management efforts. Context factors are initially exogenous factors in the sense that they are given rather than subject to choice. Yet, over the course of transitions, context factors can change endogenously due to the emergent KT outcomes.

Knowledge Recipients' Expertise or Absorptive Capacity. A key context factor is the knowledge recipient's expertise or absorptive capacity. Some studies rely on the individual-level concept of expertise, i.e., the power of cognitive schemas in the knowledge recipients' long-term memory [28,35]. Other studies rely on the highly related concept of absorptive capacity [5,11], an organizational-level concept that was originally developed in an analogy to individual-level concept of expertise [73] but then often reapplied to the individual level of analysis. Expertise affects KT in at least three ways. First, expertise affects the choice and effectiveness of KT mechanisms. Chen and McQueen [41] observed that low expertise called for more structured KT mechanisms such as formal information-sharing mechanisms. Krancher and Dibbern [67] found that although knowledge recipients learned

Table 11.4 Findings related to context factors

Context factors	*Definitions*	*Key findings*
Knowledge recipient's expertise/ absorptive capacity	• Knowledge recipient's expertise: The power of domain-specific cognitive schemas in the knowledge recipients' long-term memory [67] • Knowledge recipient's absorptive capacity: The knowledge recipient's ability to utilize outside knowledge [11]	The knowledge recipient's expertise or absorptive capacity affects KT efforts [9,11,54], the choice and effectiveness of particular KT mechanisms [41,42,53,67], and the extent to which knowledge recipients can self-manage the KT process [70].
Experts' availability and motivation	• Experts' availability: The extent to which experts have free capacity to support the KT • Experts' motivation: The experts' willingness to contribute to KT	Experts' availability and their motivation to share knowledge are critical for KT [1,2,9,52,58,70,71].
Distance (cultural, geographic, semantic, temporal, status, cognitive)	• Cultural distance: Differences in shared values, norms, beliefs, and assumptions between groups (such as nations or organizations) [11] • Geographic distance: The spatial distance between experts and knowledge recipients [11] • Semantic distance: The extent of language barriers between experts and knowledge recipients [11] • Temporal distance: The time-zone difference between experts and knowledge recipients [49] • Status distance: The degree to which knowledge recipients and experts differ in deference and respect for each other according to shared beliefs • Cognitive distance: Differences in taken-for-granted beliefs, perspectives, and mental models at the individual level [8,46]	Although distance of various types often introduces boundaries and thereby hampers KT [48,49,56], distance is less problematic at high levels of knowledge recipient's expertise [11,41,42,70] and at low levels of task complexity [44].
Social capital	Resources embedded in the relationship between experts and knowledge recipients. Resources include network ties, trust, norms, identity, and knowledge [66,72].	Social capital affects the experts' ability and willingness to share knowledge and the knowledge recipients' willingness to ask questions [66].
Turnover	The frequency at which engineers leave the SDU unit	Since high turnover often requires redoing the KT [11] and reduces experts' motivation to share knowledge [66], client managers should take actions to reduce turnover, such as through contractual governance and through socialization efforts [58].
Task complexity	The extent to which a task involves many elements, to which the relationships between these elements are sophisticated, and to which changes in the world interfere with the task [67]	Higher task complexity requires higher amounts of direction in order to reduce cognitive load to a manageable level [67]. High task complexity also makes transitions riskier [9].

most effectively from learning tasks irrespective of their level of expertise, expertise affected how strongly the cognitive load imposed by learning tasks needed to be reduced through direction, simple-to-complex sequencing, and supportive information. Second, expertise affects the management efforts required from the client. Knowledge recipients with high expertise are in a better position to self-manage KT than are knowledge recipients at low levels of expertise [70]. Moreover, low levels of expertise bear a strong risk for vicious circles of negative outcomes, declining trust, and weak helping behaviours. Client management actions are required to break these circles [58,66,70,74]. Third, expertise affects the effort for KT. The lower the expertise, the higher the costs and time required for KT [9,11,47,54]. In light of the first two effects of expertise, higher costs and longer transition durations are a straightforward consequence from the higher need for cognitive load reduction mechanisms and for client management efforts, both of which typically demand effort from the client. Interestingly, expertise is not only a context factor but also the key outcome from KT [67]. This makes KT a dynamic processes where the outcomes from initial activities strongly affect subsequent activities [53,74].

Experts' Availability and Motivation. While expertise is a critical property of the knowledge recipients, key properties of the experts involved in KT are their availability and motivation. Especially when the knowledge recipients' expertise is low, the availability of experts and their motivation to share their knowledge with the knowledge recipients become critical [1,2,9,52,58,70,71]. Motivation may depend on a number of factors. First, experts are more willing to support KT if the client offers a perspective for attractive future work to them as individuals [58,66] and, in the case of between-vendor transitions, to the incumbent vendor [4,12]. Experts' motivation may also suffer from high turnover rates of SDU personnel. From the experts' perspectives, high turnover rates make KT a Sisyphean task, where they have to start KT from scratch many times [66]. Somewhat relatedly, motivation depends on a number of tightly related attitudes and beliefs held by experts, such as their general attitudes towards offshoring (in case of offshoring), outcome expectations, and trust in the new SDU [70,75,76]. These attitudes are often a result from turnover and from initial work outcomes [70,75].

Distance. The literature abounds with descriptions of how distances of various types hamper KT. Types of distances include cultural [41,42,48,56,58], geographic [11,58], semantic [11,48], temporal [49], status-related [49], and cognitive [46]. Interestingly, although these distances are of different kinds, they are similar in their effects on KT. They present barriers for social interaction, making thus KT more difficult and calling for greater amounts of client management involvement [11,49,58,70]. For instance, although offshore employees may initially have lower status than client employees and although this hampers social interaction and KT, client managers can raise the status of offshore employees by symbolic action that treats offshore employees as team members equal to client employees [49]. Although distances of various types all hamper KT, they do not hamper KT in all projects to the same extent. Some evidence suggests that distance is less problematic when the knowledge recipients' expertise is relatively high [11,41,42] and when tasks are rather simple, such as in the case of infrastructure outsourcing [44]. Consequently, in such cases less client management effort is required to overcome KT barriers [70].

Social Capital. Although most studies rely on the negative notions of distance (or differences) to explain barriers to interaction during KT, some work relies on the positive but highly related notion of social capital [49,66]. Social capital refers to resources embedded in the relationship between experts and knowledge recipients, including network ties, trust, shared norms and identities, and shared knowledge [66,72]. There is evidence that social capital facilitates KT [39]. This is not surprising given that some of the dimensions of social capital are positive equivalents of negative notions of distance. For instance, shared knowledge is largely equivalent to (lack of) cognitive distance. Other dimensions of social capital go beyond notions of distance. For instance, trust between experts and knowledge recipients can enable effective KT even under adverse distance configurations, such as when cultural and semantic differences are large [66,70,75].

Turnover. In particular in offshoring projects, turnover in the new SDU is often a problem [11,58,66]. Turnover often hits projects severely, requiring them to redo the KT [11] and reducing the experts' motivation to share knowledge [66]. The critical role of turnover also bears one interesting insight for the types of knowledge involved in transitions: It suggests that individual, rather than collective, knowledge is key. For, if only collective knowledge (e.g., practices, transactive memory systems) were critical, projects would be largely unaffected by the departure of individuals, with hardly any need for KT to be redone.

Task Complexity. Not surprisingly, KT in transitions related to complex tasks is more difficult than in transitions related to simple tasks [9,54,67]. Task complexity is to some extent exogenous because it depends, for instance, on the maturity and complexity of the software [9]. Task complexity is, however, also subject to choice given that managers may purposefully assign tasks to the new SDU based on task complexity [10,67].

In conclusion, the knowledge recipient's expertise is among the most critical context factors influencing KT efforts and the choice of KT mechanisms. Distance and low expert motivation complicate KT but can be mitigated by strong client management efforts.

11.3.4 KT management

The literature reveals five important KT management mechanisms: formal transition governance, boundary spanners, staff selection, team-based organization, and coexistence. Table 11.5 gives an overview including definitions and key findings.

Formal Governance. Case studies show that transitions often require substantial efforts for formal governance from the client [5,11,58]. Formal governance is often critical because it helps promote social interaction and break vicious circles of disappointing outcomes, weak social interaction, and weak learning [58,70,74,75]. Among the most important formal governance mechanisms are exit criteria, monitoring of KT outcomes, communication structures, task assignment, and transition plans. Clients may formally specify exit criteria, i.e., criteria that need to be fulfilled for the transition to be considered complete [2,44,70,10]. These criteria may dictate at which point the vendor takes over the majority of the work, which is a critical decision in any transition [10]. Clients may also monitor KT outcomes, such as by examining the types and numbers of questions asked by SDU personnel [47,53], and they compare these questions to the requirements according to exit criteria. Exit criteria and their monitoring are what the IS project control literature calls outcome control because they refer to the learning outcomes of the transition process [70,79]. Many clients and vendors find it important to define formal structures for regular communication such as through steering committees [2], other types of regular meetings [58], and mirroring. In mirroring, the vendor deliberately mirrors the organizational structure of the client in order to facilitate communication [1,65,43]. Although sometimes not acknowledged as part of KT management, task assignment is also a critical mechanism for managing KT [8,47,66,70]. Since learning tasks are among the most essential KT mechanisms, the assignment of particular tasks to particular engineers in the new SDU helps clients to make sure that engineers have sufficient learning opportunities [8,47,66,70] and that these learning opportunities fit

Table 11.5 Findings related to KT management

KT management mechanism	*Definition*	*Key findings*
Formal transition governance	The client management's attempts to influence behaviour in transitions through written and/or authority-based mechanisms	Formal transition governance mechanisms, such as exit criteria, monitoring, and task assignment, help promote social interaction and break vicious circles [58,74,75].
Boundary spanners	Individuals who have the formal role of facilitating KT (nominated boundary spanners) or who facilitate KT (boundary spanners-in-practice) by helping to establish connections between client and vendor [77]	Middle managers are in a strong position to act as boundary spanners-in-practice by treating offshore employees as team members equal to client employees [49,58].
Staff selection	The client's efforts to influence the selection of SDU engineers	Both in offshore outsourcing and in captive offshoring, clients often influence the selection of vendor staff to ensure minimum levels of initial expertise and communication skills [5,11,58,70].
Team-based organization	An organizing mode that views SDU engineers and client personnel as part of the same team (rather than SDU engineers as suppliers)	Team-based organization helps promote ongoing support by the client's engineers [5,49,66].
Coexistence	The simultaneous presence of experts from the client or the former SDU and of engineers from the new SDU.	Since coexistence is essential for the vendor's learning processes [78], too short coexistence phases can substantially delay the time after which the vendor is able to perform the service [74].

their level of expertise [67]. Last but not least, transition plans (or transition guides) can be an important mechanism for KT management, in particular in large-scale transitions [5]. Transition plans are blueprints for the activities to be conducted during KT [5]. The definition of communication structures, task assignment, and transition plans are what the IS project control literature calls behaviour control because they refer to the procedures through which knowledge is to be transferred [70,79].

Boundary Spanners. A second management mechanism is the institution of boundary spanners (or offshore coordinators or middlemen). Nominated boundary spanners are individuals whose formal role is to act as a bridge between the client and the SDU and to thereby facilitate KT. In contrast, boundary spanners in practice are individuals who act as a bridge without being formally assigned the role of boundary spanners [77]. Boundary spanners fulfil a number of functions such as coordinating both teams, cultivating and intensifying relationships, eliminating status differences, filling cultural gaps, and overcoming communication barriers [49,58,80]. The client's middle managers, with their strong knowledge of the project and their power to influence engineers, are in a strong position to act as boundary spanners [49,58].

Staff Selection. In many projects, clients make efforts to select the individuals working in the new SDU. Interestingly, clients select staff not only in captive offshoring, where the client concludes employment contracts with the individuals in the captive centre [5], but also in offshore outsourcing [11,58], where clients exert influence on staff selection by vendor managers. Since staff selection helps influence the engineers' initial expertise levels (by selecting individuals with substantial prior experience in the applications and technologies relevant in the project) and since their expertise is critical, staff selection can help strongly reduce transition durations [74]. But clients may select staff to influence not only their initial expertise but also their communication skills, helping thus to promote effective social interaction during KT [11,58].

Team-Based Organization. A fourth important KT management mechanism according to the literature is team-based organization, i.e., a mode in which SDU engineers and client personnel are seen as members of the same team rather than SDU engineers seen as suppliers to the client team. A supplier-based organizing mode often evokes the expectation that SDU engineers should be able to independently perform their work without the help of the client engineers [5,66]. Moreover, a supplier-based organizing mode reinforces status differences between client and SDU [49]. In contrast, a team-based organizing mode helps sensitize client engineers for the need to ongoingly support SDU engineers. Team-based organization is thus a strategy to ensure sufficient cognitive load reduction for SDU engineers even after formal KT has been completed.

Coexistence. Last but not least, decisions related to coexistence can have important impact on KT. Coexistence denotes the time during which experts and engineers from the new SDU are simultaneously present in the project. While some clients may decide that experts and SDU engineers are co-present over the course of an entire project (i.e., concurrent sourcing) [78], a more typical scenario is that after some time experts are released and are, hence, no longer available to help the engineers from the new SDU. Using simulation methods, Krancher and Dibbern [74] showed that short coexistence durations can have strong negative impact on transitions if the knowledge recipients' expertise is relatively low. In such conditions, too short coexistence durations result in SDU engineers being cognitively overloaded for relatively long time, giving rise to long periods of unsatisfactory performance and weak learning.

11.3.5 Summary

Taken together, the literature on transitions echoes a number of findings from the broader KT and expertise literatures but also goes beyond these literatures in a number of ways. The findings in two areas are largely consistent: KT *mechanisms* and context factors. While the expertise literature emphasizes that long periods of practice through scaffolded learning tasks are essential KT mechanisms, the literature on transitions is very much in line with this assertion. Studies abound with accounts of how projects initially attempted to transfer knowledge through formal information-sharing mechanisms but then managed to hand over the service to the new SDU only after the engineers from the new SDU had sufficient opportunities to work on learning tasks (authentic or real tasks). This points to an important potential for improving practice. Practitioners should be sensitized for the need for learning tasks, and for the limitations of popular information-sharing mechanisms such as documents. The findings on *context factors* are also highly consistent with the reference literature, in particular with the KT literature.

In line with that literature, the knowledge recipient's expertise, or absorptive capacity, turned out to be a key factor in many studies. Moreover, the important role of various types of distance and of social capital is highly consistent with the important role of relational factors in the KT literature.

While the literature on transitions thus invigorates findings from the reference literature around KT mechanisms and context factors, it goes beyond the reference literature in the other two topic areas covered by this review. With regard to *knowledge categories*, the important role of turnover and the accounts of cognitively overloaded individuals in the new SDU suggest that the transfer of individual, or personal [25], knowledge is often the most critical issue. Although more research is needed to compare the relative importance of individual with collective knowledge, the tentative conclusion that individual knowledge and individual learning are key in transitions would have important implications for practice. In practice, vendors often attempt to make their delivery robust to the departure of individuals, advertising their structured approaches for capturing knowledge [43]. A more successful approach for vendors might be to openly admit that their ability to provide the service strongly depends on the expertise of particular individuals and to deliberately manage this personal knowledge, rather than to attempt to obviate the need for it. The literature also gives important insights into knowledge domains. Going beyond the prevailing business-technical knowledge dichotomy, a number of studies have shown that, in application outsourcing, the most critical domain of knowledge is often application knowledge. Oftentimes, SDU engineers are able to take over tasks only after they have acquired sufficiently powerful cognitive schemas of the structure and functioning of existing software applications, typically through a relatively long period of practice.

The literature on transitions also goes beyond the KT and expertise literature in its findings on KT management, a topic that has enjoyed relatively little attention in the reference literature. The key finding is that the more adverse the circumstances in transitions (e.g., low knowledge recipient's expertise, high distance in many dimensions), the greater the management efforts required from the client. Moreover, the need for team-based structures and the findings related to coexistence suggest that, in some cases, permanent coexistence of the client's experts and SDU engineers may be the most effective way to avoid knowledge losses.

11.4 KT in transitions: implications in the era of digital business strategy and agility

While the preceding review provides a synthesis of themes that turned out important in the past, the remainder of this chapter draws on and extrapolates these findings to discuss implications from these findings for a future in which two themes are likely to gain importance: digital business strategy and agility.

As described in greater detail in other sections of this book, *digital business strategy* describes an organizational strategy in which IS are not seen as a separate function that exists independent of other business functions; instead, digital business strategy emphasizes how digital resources and business activities have become inseparable in attempts to create differential impact with digital technology [16]. Digital business strategy draws thus on a relational (or fusion) perspective of technology and social action in which technology and social action are not separate things but tightly entangled or inseparable [81,82]. For instance, it is difficult to separate a firm's digital products from technology since many digital products would not exist without the technology. Rather than to ask how an IS function can be aligned with the rest of an organization, the question becomes what the organization can do with digital resources in order to create and maintain competitive advantage. This draws attention to design moves, i.e., the ways in which organizations recombine, reconfigure, or design digital resources to change products or processes in response to market opportunities or competitors' action, and to design capital, i.e., the ways in which existing digital resources and knowledge enable or constrain these moves [83].

Agility is an organization's or a team's ability to rapidly create and react to change and to learn from it while focusing on customer value [18]. Agile organizations are able to quickly seize competitive market opportunities by quickly re-assembling digital resources and knowledge [84]. Although the concept of agility is highly related to digital business strategy, agility emphasizes the speed of change while digital business strategy emphasizes the relational nature of change.

The era of digital business strategy and agility is characterized not only by increasing desires for digital business reconfiguration and agility but also by new opportunities offered by technology. These opportunities are related to digital business strategy and agility in that they may enable organizations to more effectively implement digital business strategies and to increase agility. These technology-based opportunities include modular microservice-based architectures [85,86], digital ecosystems [87], and Cloud technologies [88].

We next discuss how desires and opportunities in the era of digital business strategy and agility are related to the findings from our literature review. From these considerations, we draw three normative implications for how to deal with KT in transitions in this new era: cultivate long-term, embedded, concurrent-sourcing relationships; accelerate learning by organizing KT around learning tasks and feedback; and reduce the need for learning by hiding complexity. Table 11.6 gives an overview of desires, opportunities, findings from our literature review, and the posited implications.

Table 11.6 Implications for the era of digital business strategy and agility

Desires and opportunities in the era of digital business strategy and agility	*Key findings of the literature on KT during transitions*	*Implications in the era of digital business strategy and agility*
Desire: Ability to reconfigure and recombine digital resources while not affecting their stability	• Application and business expertise are important. • The acquisition of sufficient expertise may take years. • In the absence of sufficient expertise held by the SDU, social ties are essential to reduce cognitive load.	Cultivate long-term, embedded, concurrent-sourcing relationships
Desire: Agility Opportunity: Technologies that enable rapid feedback	Learning tasks and feedback enable effective learning.	Accelerate learning by organizing KT around learning tasks and feedback
Opportunity: Technologies that reduce complexity	Cognitive load is often the bottleneck that limits the ability to change digital resources.	Reduce the need for learning by hiding complexity

11.4.1 Cultivate long-term, embedded, concurrent-sourcing relationships

At the heart of the concept of digital business strategy lies the desire to be able to reconfigure and recombine digital resources to enact new business opportunities [83]. At the same time, these reconfiguration or recombination moves shall not adversely affect existing operations. Three findings from our review of the literature on KT in transitions suggest that these desires pose *enormous cognitive and coordinative challenges*. A first finding is that strong business and, in particular, application knowledge are required to design and perform changes in digital resources that do not affect existing operations while allowing new business opportunities [5,70]. Engineers need powerful mental models of existing digital resources (i.e., application knowledge) before they are able to effectively change them. If these changes shall serve to enable novel business opportunities, strong knowledge of the organization's business is likely to be required as well. Engineers, who lack intimate familiarity with the client's existing digital resources and with the client's business, are likely to overlook ways to leverage existing digital resources or to overlook side effects that the changes have on existing operations and products. Moreover, strong cognitive schemas about existing infrastructures will help reduce the cognitive load on engineers, leaving them sufficient mental capacity to think about how to recombine and reconfigure the resources for innovation. Second, acquiring such strong application and business expertise often takes long time, in many cases years [9,10]. It is only through the repeated engagement in a variety of real or realistic tasks in the particular software applications that engineers will acquire such levels of expertise [57]. Third, if the SDU engineers lack such expertise, they will be cognitively overloaded unless social ties to experts allow them to obtain direction and information.

These findings suggest that a *new vendor*, who lacks familiarity with a client's existing digital resources and with the client's business, will typically be *unable*, for a relatively long time, to reconfigure and recombine the client's digital resources to create new business opportunities while not affecting the client's operations. The vendor's engineers would be unlikely to possess such application and business expertise given their lack of familiarity with the client's specific software applications. Even transitions with KT efforts of a few months are unlikely to elevate the vendor engineers' expertise to a level that would enable such transformative competent action. Instead, years of practice with a client's applications and business are likely to be required before the vendor engineers' expertise is high enough. However, transition durations of several years are neither economically viable nor desirable in terms of agility.

If years of practice are required but not economically viable, how can clients ensure that sufficient expertise is present while still relying on offshoring or outsourcing? A powerful solution is to cultivate long-term, embedded, concurrent-sourcing relationships with vendors. By long term, I mean that clients should seek to build relationships over many years. Although contracts need not necessarily be many years long, clients may need to offer a constant perspective for future business to the vendor, which may lead vendors to make long-term commitments for key personnel. Such long-term relationships will allow vendor engineers to acquire deep expertise in the client's applications and business in order to be able to guide the client at a transformative level. By embedded, I mean that vendor engineers collaborate with client personnel in team-based organizing forms. Such embedded forms provide the foundation for ongoing mutual support, which is essential to reduce the enormous cognitive load associated with transformative moves. By concurrent sourcing [78], I mean that the client keeps in-house experts involved in the projects. In-house experts will grant greater continuity of expertise. Their expertise will be needed to guide those SDU engineers who have not yet several years of working experience with the client. Positive side effects of retaining in-house experts include that these in-house experts also help the client to control projects more effectively [79] and that they reduce the dependency on the vendor, mitigating hold-up risks [10,14].

11.4.2 Accelerate learning by organizing KT around learning tasks and feedback

A key facet of the desire for greater agility lies in the speed at which organizations are able to realize and implement opportunities for innovation by modifying digital infrastructure. While typical lead times for changes in digital infrastructure were in the range of years some years ago, organizations now increasingly strive for lead times in the range of weeks [89,90]. Clearly, transition durations of months or years do not align well with such desired lead times. This is why clients should cultivate long-term embedded relationships with their SDU to make sure that expertise is retained rather than needs to be transferred for each new project. However, even in long-term embedded relationships not all required expertise may be on board from the beginning. Especially projects that strive for innovation will likely need not only to access existing expertise but also to engender quick learning in unfamiliar areas.

Research on transitions suggests that learning tasks and quick feedback are effective mechanisms to engender such quick learning. The concept of learning tasks implies that teams should do, rather than plan, the recombination and reconfiguration moves they are considering. If engineers need to acquire knowledge about new technologies or about applications or business areas with which they are not familiar, engineers may attempt to acquire this knowledge by working on a number of tasks in these areas (e.g., a number of simple tasks that occurred in the past), rather than by studying documents. The claim that learning tasks rather than formal information sharing are effective strategies for learning under high innovation pressures is also in line with research that found traditional organizational memory systems to be inappropriate in turbulent environments [17]. Quick feedback implies that teams should perform these moves and observe the outcomes as quickly and frequently as possible. This will help business users and engineers quickly build mental models of potential moves and their prerequisites, complications, and consequences. Moreover, by relying on learning tasks and feedback, engineers will rapidly learn the specifics of new technologies that are involved in the moves. The key role of learning tasks and feedback is also corroborated by organizational research on collaboration projects, which found that experimenting through active learning activities is key [91].

Although organizations may find it difficult to change their processes towards quicker feedback, recent research suggests that Cloud technologies can serve as an enabler in this mission [92]. Due to its on-demand self-service characteristic [93], Cloud technology allows engineers to instantly set up infrastructure and to instantly make changes to such infrastructure. Thus, Cloud technology can help accelerate feedback processes and learning [94].

11.4.3 Reduce the need for learning by hiding complexity

As argued above, the era of digital business strategy and agility will yield a strong increase in the cognitive and coordinative demands on SDUs. The speed at which digital infrastructures are created, reconfigured, and recombined and the complexity of existing infrastructures (e.g., the number of applications, the number of interfaces) might increase. SDU engineers require strong cognitive schemas of these increasingly large and entangled infrastructures.

Their cognitive schemas need to be so powerful that engineers are able to think about these existing infrastructures and still have sufficient cognitive capacity to think about innovative changes to digital infrastructures. These increasing cognitive demands may make any outsourcing or offshoring attempts, which necessitate transitions of years, economically prohibitive, unless deliberate attempts are made to reduce complexity.

Fortunately, technologies that help reduce complexity by hiding information are increasingly powerful and common. Three such important technologies are microservices, digital ecosystems, and Cloud computing. Microservices are a strategy to encapsulate elementary business functionality in a reusable service [85,86,95]. Microservices hide the complexity that lies behind the implementation of the service to those that want to consume the service. This helps reduce cognitive load in attempts to recombine existing digital infrastructures because engineers need not process any information related to the inner workings of the service. A highly related concept is modularity [96]; microservices enable a modular architecture of digital resources. Digital ecosystems, such as the Salesforce AppExchange or Heroku Elements, provide existing, often relatively small, applications or application components that organizations can reuse and recombine [87]. They hide the complexity that lies behind the implementation of the application or application component. Cloud services are on-demand self-services that include virtualized infrastructure (Infrastructure-as-a-Service), application environments (Platform-as-a-Service), or ready-to-use software (Software-as-a-Service). Cloud services hide the complexity by abstracting away the details behind setting up infrastructure, environments, and readily configure software components [94]. The use of these three technologies reduces cognitive load by eliminating information elements that engineers need to process. Thus, the use of these technologies may serve to counterbalance the increased cognitive demands through the increasing variety of existing applications and the increased need for speed. Organizations that manage to leverage these technologies to a great extent may even be able to shorten transition durations and, hence, need to rely to a lesser extent on long-term, embedded, dual-sourcing relationships. Whether and under what circumstances transitions will thus become simpler or more complex remains an intriguing question for future research.

11.5 Conclusion

KT during transitions is one of the key issues in outsourcing and offshoring relationships. The critical nature of KT has given rise to a vibrant stream of research that has examined what knowledge needs to be transferred, through what mechanisms knowledge can be transferred effectively, how context factors influence KT, and how KT can be effectively managed. This research has invigorated some findings from the broader KT literature and from expertise literature. These findings include that the knowledge recipients' expertise is among the most important context factors and that learning tasks (i.e., practice based on authentic tasks) are among the most effective KT mechanisms, although many projects initially rely on formal information-sharing mechanisms (e.g., documents). Research on transitions goes also beyond reference literatures by revealing what categories of knowledge are critical in IS transitions and how KT can and should be managed. Although evidence is not yet conclusive, findings suggest that in many transitions the transfer of individual, rather than of collective, knowledge is the key bottleneck, and that application knowledge is often the most important knowledge domain. A key finding on KT management is that the more adverse the circumstances (e.g., low knowledge recipient's expertise, high distances in a number of dimensions), the more client management involvement is needed.

While these findings stem from the past, they are also informative in a future in which digital business strategy and agility will become increasingly important. The findings sensitize for the high amounts of expertise that are required to understand existing digital resources and for the long time it takes to build such expertise. At a time when rapid innovation by reconfiguring and recombining digital resources is much sought, the need for such expertise is unlikely to decline. Clients may therefore need to cultivate long-term, embedded, concurrent-sourcing relationships with vendors to ensure that expertise is retained. KT approaches based on learning tasks and feedback, rather than based on documents, are likely to become even more needed to satisfy the need for quick learning and agility. Although cognitive demands may increase with the increasing variety of existing digital resources and with the increasing demand for agility, the era of digital business strategy and of agility also offers new technologies that may help reduce cognitive demands by hiding complexity. These technologies include microservices, digital ecosystems, and Cloud computing. Organizations that leverage these technologies may even be able to shorten transition durations. Future practice and research will yield insights into whether and how important these changes are, how they interact, and how they will inform KT during transitions.

References

1. Tiwari, V. Transition during Offshore Outsourcing: A Process Model. In The 30th *International Conference of Information Systems*, Phoenix, AZ, 2009.
2. Beulen, E. and V. Tiwari, Parallel Transitions in IT Outsourcing: Making It Happen. In *Global Sourcing of Information Technology and Business Processes*, I. Oshri and J. Kotlarsky, Editors. 2010, Heidelberg: Springer, pp. 55–68.
3. Ravindran, K., Asset Transfer in IT Outsourcing: Divesting Commodities or Inviting Investment?, In *The Dynamics of Global Sourcing. Perspectives and Practices*, J. Kotlarsky, I. Oshri, and L. Willcocks, Editors. 2012, Heidelberg: Springer, pp. 39–60.
4. Sia, S.K., W.K. Lim, and K.P. Periasamy, Switching IT Outsourcing Suppliers: Enhancing Transition Readiness. *MIS Quarterly Executive*, 2010. 9(1): pp. 203–213.
5. Chua, A.L. and S.L. Pan, Knowledge Transfer and Organizational Learning in IS Offshore Sourcing. *Omega*, 2008. 36(2): pp. 267–281.
6. Blumenberg, S., H.T. Wagner, and D. Beimborn, Knowledge Transfer Processes in IT Outsourcing Relationships and their Impact on Shared Knowledge and Outsourcing Performance. *International Journal of Information Management*, 2009. 29(5): pp. 342–352.
7. Carmel, E. and P. Tjia, *Offshoring Information Technology: Sourcing and Outsourcing to a Global Workforce*. Cambridge: Cambridge University Press, 2005.
8. Madsen, S., K. Bødker, and T. Tøth, Knowledge Transfer Planning and Execution in Offshore Outsourcing: An Applied Approach. *Information Systems Frontiers*, 2015. 17(1): pp. 67–77.
9. Smite, D. and C. Wohlin, Strategies Facilitating Software Product Transfers. *IEEE Software*, 2011. 28(5): pp. 60–66.
10. Williams, C. and S. Durst, Exploring the Transition Phase in Offshore Outsourcing: Decision Making Amidst Knowledge at Risk. *Journal of Business Research*, 2019. 103: pp. 460–471.
11. Dibbern, J., J. Winkler, and A. Heinzl, Explaining Variations in Client Extra Costs between Software Projects Offshored to India. *MIS Quarterly*, 2008. 32(2): pp. 333–366.
12. Chua, C.E.H., et al., Client Strategies in Vendor Transition: A Threat Balancing Perspective. *The Journal of Strategic Information Systems*, 2012. 21(1): pp. 72–83.
13. Willcocks, L. and M.C. Lacity, *Global Sourcing of Business and IT Services*. London: Palgrave Macmillan, 2006.
14. Williamson, O.E., Transaction-Cost Economics: The Governance of Contractual Relations. *The Journal of Law & Economics*, 1979. 22(2): pp. 233–261.
15. Conner, K. and C. Prahalad, A Resource-Based Theory of the Firm: Knowledge versus Opportunism. *Organization Science*, 1996. 7(5): pp. 477–501.
16. Bharadwaj, A., et al., Digital Business Strategy: Toward a Next Generation of Insights. *MIS Quarterly*, 2013. 37: pp. 471–482.

17. Pavlou, P.A. and O.A. El Sawy, The "Third Hand": IT-Enabled Competitive Advantage in Turbulence Through Improvisational Capabilities. *Information Systems Research*, 2010. 21(3): pp. 443–471.
18. Conboy, K., Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development. *Information Systems Research*, 2009. 20(3): pp. 329–354.
19. Grant, R.M., Toward a Knowledge-Based Theory of the Firm. *Strategic Management Journal*, 1996. 17(10): pp. 109–122.
20. Spender, J.C., Making Knowledge the Basis of a Dynamic Theory of the Firm. *Strategic Management Journal*, 1996. 17: pp. 45–62.
21. Szulanski, G., Exploring Internal Stickiness: Impediments to the Transfer of Best Practice within the Firm. *Strategic Management Journal*, 1996. 17(1): pp. 27–43.
22. Tsai, W., Knowledge Transfer in Intraorganizational Networks: Effects of Network Position and Absorptive Capacity on Business Unit Innovation and Performance. *Academy of Management Journal*, 2001. 44(5): pp. 996–1004.
23. Ko, D.G., L.J. Kirsch, and W.R. King, Antecedents of Knowledge Transfer from Consultants to Clients in Enterprise System Implementations. *MIS Quarterly*, 2005. 29(1): pp. 59–85.
24. Szulanski, G., The Process of Knowledge Transfer: A Diachronic Analysis of Stickiness. *Organizational Behavior and Human Decision Processes*, 2000. 82(1): pp. 9–27.
25. Polanyi, M., *Personal Knowledge. Towards a Post-Critical Philosophy.* Chicago, IL: The University of Chicago Press, 1962.
26. Van Wijk, R., J.J. Jansen, and M.A. Lyles, Inter-and Intra-Organizational Knowledge Transfer: A Meta-Analytic Review and Assessment of Its Antecedents and Consequences. *Journal of Management Studies*, 2008. 45(4): pp. 830–853.
27. Lam, A., Tacit Knowledge, Organizational Learning and Societal Institutions: An Integrated Framework. *Organization Studies*, 2000. 21(3): p. 487.
28. Chase, W.G. and H.A. Simon, Perception in Chess. *Cognitive Psychology*, 1973. 4(1): pp. 55–81.
29. Adelson, B., Problem Solving and the Development of Abstract Categories in Programming Languages. *Memory & Cognition*, 1981. 9(4): pp. 422–433.
30. Doane, S.M., J.W. Pellegrino, and R.L. Klatzky, Expertise in a Computer Operating System: Conceptualization and Performance. *Human-Computer Interaction*, 1990. 5(2): pp. 267–304.
31. Voss, J.F., et al., Problem-Solving Skill in the Social Sciences. In *The Psychology of Learning and Motivation*, G.H. Bower, Editor. 1983, Stanford, CA: Stanford University. pp. 165–213.
32. Ericsson, K.A., R.T. Krampe, and C. Tesch-Römer, The Role of Deliberate Practice in the Acquisition of Expert Performance. *Psychological Review*, 1993. 100(3): p. 363.
33. Dreyfus, H.L. and S.E. Dreyfus, Peripheral Vision: Expertise in Real World Contexts. *Organization Studies*, 2005. 26(5): pp. 779–792.
34. Ericsson, K.A., M.J. Prietula, and E.T. Cokely, The Making of an Expert. *Harvard Business Review*, 2007. 85(7/8): pp. 115–121.
35. Sweller, J., J.J.G. Van Merriënboer, and F.G.W.C. Paas, Cognitive Architecture and Instructional Design. *Educational Psychology Review*, 1998. 10(3): pp. 251–296.
36. Van Merriënboer, J.J.G., R.E. Clark, and M.B.M. De Croock, Blueprints for Complex Learning: The 4C/ID-Model. *Educational Technology Research and Development*, 2002. 50(2): pp. 39–64.
37. Cha, H., D. Pingry, and M. Thatcher, Managing the Knowledge Supply Chain: An Organizational Learning Model of Information Technology Offshore Outsourcing. *Management Information Systems Quarterly*, 2008. 32(2): p. 7.
38. Teo, T.S. and A. Bhattacherjee, Knowledge Transfer and Utilization in IT Outsourcing Partnerships: A Preliminary Model of Antecedents and Outcomes. *Information & Management*, 2014. 51(2): pp. 177–186.
39. Zimmermann, A., et al., Sourcing In or Out: Implications for Social Capital and Knowledge Sharing. *The Journal of Strategic Information Systems*, 2018. 27(1): pp. 82–100. doi:10.1016/j.jsis.2017.05.001
40. Lee, J.-N., The Impact of Knowledge Sharing, Organizational Capability and Partnership Quality on IS Outsourcing Success. *Information & Management*, 2001. 38(5): pp. 323–335.
41. Chen, J. and R.J. McQueen, Knowledge Transfer Processes for Different Experience Levels of Knowledge Recipients at an Offshore Technical Support Center. *Information Technology & People*, 2010. 23(1): pp. 54–79.
42. Chen, J., P.Y. Sun, and R.J. McQueen, The Impact of National Cultures on Structured Knowledge Transfer. *Journal of Knowledge Management*, 2010. 14(2): pp. 228–242.

43. Kotlarsky, J., H. Scarbrough, and I. Oshri, Coordinating Expertise across Knowledge Boundaries in Offshore-Outsourcing Projects: The Role of Codification. *MIS Quarterly*, 2014. 38(2): pp. 607–627.
44. Hawk, S., W. Zheng, and R.W. Zmud, Overcoming Knowledge-Transfer Barriers in Infrastructure Management Outsourcing: Lessons from a Case Study. *MIS Quarterly Executive*, 2009. 8(3): pp. 123–129.
45. Wende, E., G. King, and G. Schwabe, *Exploring Storytelling as a Knowledge Transfer Technique in Offshore Outsourcing*. Auckland, New Zealand: International Conference on Information Systems, 2014.
46. Nicholson, B. and S. Sahay, Embedded Knowledge and Offshore Software Development. *Information and Organization*, 2004. 14(4): pp. 329–365.
47. Leonardi, P.M. and D. Bailey, Transformational Technologies and the Creation of New Work Practices: Making Implicit Knowledge Explicit in Task-Based Offshoring. *MIS Quarterly*, 2008. 32(2): pp. 159–176.
48. Imsland, V. and S. Sahay, 'Negotiating Knowledge': The Case of a Russian–Norwegian Software Outsourcing Project. *Scandinavian Journal of Information Systems*, 2005. 17(1): p. 3.
49. Levina, N. and E. Vaast, Innovating or Doing as Told? Status Differences and Overlapping Boundaries in Offshore Collaboration. *MIS Quarterly*, 2008. 32(2): pp. 307–332.
50. Deng, C.-P. and J.-Y. Mao, Knowledge Transfer to Vendors in Offshore Information Systems Outsourcing: Antecedents and Effects on Performance. *Journal of Global Information Management (JGIM)*, 2012. 20(3): pp. 1–22.
51. Iivari, J., R. Hirschheim, and H. Klein, Towards a Distinctive Body of Knowledge for Information Systems Experts: Coding ISD Process Knowledge in Two IS Journals. *Information Systems Journal*, 2004. 14(4): pp. 313–342.
52. Feng, Y., H. Ye, and S.L. Pan, Delivering Knowledge across Boundaries: A Process Model of Knowledge Delivery in Offshoring Projects. *Pacific Asia Conference on Information Systems*. Taipei, Taiwan, 2010.
53. Schott, K. Vendor-Vendor Knowledge Transfer in Global ISD Outsourcing Projects: Insights from a German Case Study. *PACIS*, 2011.
54. Vlaar, P., P. van Fenema, and V. Tiwari, Cocreating Understanding and Value in Distributed Work: How Members of Onsite and Offshore Vendor Teams Give, Make, Demand, and Break Sense. *MIS Quarterly*, 2008. 32(2): p. 5.
55. Park, J.Y., K.S. Im, and J.S. Kim, The Role of IT Human Capability in the Knowledge Transfer Process in IT Outsourcing Context. *Information & Management*, 2011. 48(1): pp. 53–61.
56. Wende, E., et al. Why They Do Not Understand – A Model of Knowledge Discourse in the Transition Phase of Globally Distributed Projects. In System Sciences (HICSS), 2013 46th Hawaii International Conference On. IEEE, 2013.
57. Krancher, O. and J. Dibbern. Knowledge in Software-Maintenance Outsourcing Projects: Beyond Integration of Business and Technical Knowledge. In The *48th Hawaii International Conference on System Sciences*, Kauai, HI, 2015.
58. Gregory, R., R. Beck, and M. Prifling, Breaching the Knowledge Transfer Blockade in IT Offshore Outsourcing Projects – A Case from the Financial Services Industry. In The *43th Hawaii International Conference on Systems Sciences*. Big Island, HI, 2009.
59. Williamson, O.E., The Economics of Organization: The Transaction Cost Approach. *American Journal of Sociology*, 1981. 87(3): pp. 548–577.
60. Tiwana, A., Knowledge Partitioning in Outsourced Software Development: A Field Study. In *Twenty-Fourth International Conference on Information Systems*. 2003. pp. 259–270.
61. Tiwana, A., An Empirical Study of the Effect of Knowledge Integration on Software Development Performance. *Information and Software Technology*, 2004. 46(13): pp. 899–906.
62. Banker, R.D., et al., Software Errors and Software Maintenance Management. *Information Technology and Management*, 2002. 3(1–2): pp. 25–41.
63. Boh, W.F., S.A. Slaughter, and J.A. Espinosa, Learning from Experience in Software Development: A Multilevel Analysis. *Management Science*, 2007. 53(8): pp. 1315–1331.
64. Williams, C., Client–Vendor Knowledge Transfer in IS Offshore Outsourcing: Insights from a Survey of Indian Software Engineers. *Information Systems Journal*, 2011. 21(4): pp. 335–356.
65. Oshri, I., P. Van Fenema, and J. Kotlarsky, Knowledge Transfer in Globally Distributed Teams: The Role of Transactive Memory. *Information Systems Journal*, 2008. 18(6): pp. 593–616.

66. Zimmermann, A. and M. Ravishankar, Knowledge Transfer in IT Offshoring Relationships: The Roles of Social Capital, Efficac and Outcome Expectations. *Information Systems Journal*, 2014. 24(2): pp. 167–202.
67. Krancher, O. and J. Dibbern. Learning Software-Maintenance Tasks in Offshoring Projects: A Cognitive-Load Perspective. In *The 33rd International Conference on Information Systems*. Orlando, FL, 2012.
68. Cognizant. How to Conduct Knowledge Transfer Efficiently. 2002 June 6th, 2014; Available from: http://www.cognizant.com/InsightsWhitepapers/amseries-kt.pdf.
69. Tsoukas, H., Do We Really Understand Tacit Knowledge. The Blackwell Handbook of Organizational Learning and Knowledge Management, 2003: pp. 410–427.
70. Krancher, O. and S.A. Slaughter, 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, pp. 4406–4415, 2013.
71. Beulen, E., V. Tiwari, and E. van Heck, Understanding Transition Performance during Offshore IT Outsourcing. *Strategic Outsourcing: An International Journal*, 2011. 4(3): pp. 204–227.
72. Nahapiet, J. and S. Ghoshal, Social Capital, Intellectual Capital, and the Organizational Advantage. *Academy of Management Review*, 1998. 23: pp. 242–266.
73. Cohen, W.M. and D.A. Levinthal, Absorptive Capacity: A New Perspective on Learning and Innovation. *Administrative Science Quarterly*, 1990. 35(1): pp. 128–152.
74. Krancher, O. and J. Dibbern, Managing Knowledge Transfer in Software-Maintenance Outsourcing: A System-Dynamics Perspective. In *Information Systems Outsourcing: Towards Sustainable Business Value*, R. Hirschheim, A. Heinzl, and J. Dibbern, editors. Springer: Heidelberg, pp. 223–243, 2014.
75. Zimmermann, A., K. Raab, and L. Zanotelli, Vicious and Virtuous Circles of Offshoring Attitudes and Relational Behaviours. A Configurational Study of German IT Developers. *Information Systems Journal*, 2013. 23(1): pp. 65–88.
76. Westner, M. and S. Strahringer, Determinants of Success in IS Offshoring Projects: Results from an Empirical Study of German Companies. *Information & Management*, 2010. 47(5): pp. 291–299.
77. Levina, N. and E. Vaast, The Emergence of Boundary Spanning Competence in Practice: Implications for Implementation and Use of Information Systems. *MIS Quarterly*, 2005. 29(2): pp. 335–363.
78. Tiwana, A. and S.K. Kim, Concurrent IT Sourcing: Mechanisms and Contingent Advantages. *Journal of Management Information Systems*, 2016. 33(1): pp. 101–138.
79. Kirsch, L.J., The Management of Complex Tasks in Organizations: Controlling the Systems Development Process. *Organization Science*, 1996. 7(1): pp. 1–21.
80. Strasser, A. and M. Westner. Knowledge Transfer in IS Offshoring: Processes, Roles, and Success Factors. In *PACIS*, 2015.
81. Orlikowski, W.J. and S.V. Scott, Sociomateriality: Challenging the Separation of Technology, Work and Organization. *The Academy of Management Annals*, 2008. 2(1): pp. 433–474.
82. Riemer, K. and R.B. Johnston, Clarifying Ontological Inseparability with Heidegger's Analysis of Equipment. *MIS Quarterly*, 2017. 41(4): pp. 1059–1081.
83. Woodard, C.J., et al., Design Capital and Design Moves: The Logic of Digital Business Strategy. *MIS Quarterly*, 2013. 37(2): pp. 537–564.
84. Sambamurthy, V., A. Bharadwaj, and V. Grover, Shaping Agility through Digital Options: Reconceptualizing the Role of Information Technology in Contemporary Firms. *MIS Quarterly*, 2003. 27(2): pp. 237–263.
85. Yoo, Y., O. Henfridsson, and K. Lyytinen, Research Commentary—The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research. *Information Systems Research*, 2010. 21(4): pp. 724–735.
86. Grover, V. and R. Kohli, Revealing Your Hand: Caveats in Implementing Digital Business Strategy. *MIS Quarterly*, 2013. 37(2): pp. 655–662.
87. Tiwana, A., B. Konsynski, and A.A. Bush, Research Commentary-Platform Evolution: Coevolution of Platform Architecture, Governance, and Environmental Dynamics. *Information Systems Research*, 2010. 21(4): pp. 675–687.
88. Yang, H. and M. Tate, A Descriptive Literature Review and Classification of Cloud Computing Research. *Communications of the Association for Information Systems*, 2012. 31(2): pp. 35–60.
89. Lyytinen, K. and G.M. Rose, Information System Development Agility as Organizational Learning. *European Journal of Information Systems*, 2006. 15(2): pp. 183–199.

90. Yoo, Y., et al., Organizing for Innovation in the Digitized World. *Organization Science*, 2012. 23(5): pp. 1398–1408.
91. Martin, J.A. and K.M. Eisenhardt, Rewiring: Cross-Business-Unit Collaborations in Multibusiness Organizations. *Academy of Management Journal*, 2010. 53(2): pp. 265–301.
92. Krancher, O. and P. Luther, Software Development in the Cloud: Exploring the Affordances of Platform-as-a-Service. In *The 36rd International Conference on Information Systems*. Fort Worth, TX, 2015.
93. Mell, P. and T. Grance. The NIST Definition of Cloud Computing. 2011 Feb 27th, 2018; Available from: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf.
94. Krancher, O., P. Luther, and M. Jost, Key Affordances of Platform-as-a-Service: Self-Organization and Continuous Feedback. *Journal of Management Information Systems*, 2018. 35(3): pp. 776–812. doi:10.1080/07421222.2018.1481636.
95. Balalaie, A., A. Heydarnoori, and P. Jamshidi, Microservices Architecture Enables Devops: Migration to a Cloud-Native Architecture. *IEEE Software*, 2016. 33(3): pp. 42–52.
96. Baldwin, C.Y. and K.B. Clark, *Design Rules: The Power of Modularity*. MIT Press, Cambridge, Vol. 1, 2000.