Supporting Organisational Agility through Boundary Spanning and Knowledge Brokering in a Dual Operating System: A Case from the Software Industry

Christoph Rosenkranz
Depart of Information Systems
University of Cologne

Karlheinz Kautz
Department of Information Systems and Business Analytics
RMIT University

Vanessa Cooper
Department of Information Systems and Business Analytics
RMIT University
Email: vanessa.cooper@rmit.edu.au

Abstract

One challenge that organisations face today is how to develop organisational agility to remain competitive within a constantly turbulent, dynamic, and potentially disruptive environment. In this paper, we examine how a software development company operationalises Kotter's concept of a dual operating system to support organisational agility. The first operating system, the traditional hierarchical structure, enables the company to focus on providing efficiency and stability in its core operations. The second operating system prioritises flexibility. It is operationalised via boundary spanning and knowledge brokering activities rooted in a dedicated organisational unit that enables rapid response to environmental changes. We analyse the unit's and its members' approach and identify well-functioning activities and practices contributing to organisational agility as well as challenges and problems. As such we contribute to an improved understanding of how an intertwined strategy of boundary spanning and knowledge brokering, operating alongside a traditional hierarchical structure, forms a balancing mechanism between flexibility and stability, thus enabling organisational agility.

Keywords: Organisational agility, Boundary spanning, Knowledge brokering, Dual operating system, Software development.

1 Introduction

As a consequence of growing digitalisation and digital transformation, all types of products and services increasingly incorporate digital hardware and software components (Brynjolfsson & McAfee, 2014; Laato, Mäntymäki, Islam, Hyrynsalmi, & Birkstedt, 2023; Wessel, Baiyere, Ologeanu-Taddei, Cha, & Jensen, 2021). At the same time, companies need to address the challenges produced by mounting complexity and rapid change in today's competitive and constantly turbulent, and potentially disruptive, business environment (Kotter, 2012). Increasingly, therefore, software development takes place in dynamic and constantly changing situations as well as environments with rising complexity and demands for efficiency (Benbya, Nan, Tanriverdi, & Yoo, 2020; Sommerville, Cliff, Calinescu, Keen, Kelly, Kwiatkowska, Mcdermind, & Paige, 2012). This underscores the need for software development companies to become more flexible and agile (Baham & Hirschheim, 2022; Harris, Collins, & Hevner, 2009; Overby, Bharadwaj, & Sambamurthy, 2006) and to "grow

Received 21 Feb 2024, accepted 11 Feb 2025 after 2 rounds of review with Henry Linger as accepting Senior Editor, M. Davern & S. Black as Editors in Chief

agile beyond the system development team to accommodate the required interactions [with other functional areas of the organisation]" (Ågerfalk, Fitzgerald, & Slaughter, 2009). That is, a shift beyond agile development of IT towards wider organisational agility is required (Tallon, Queiroz, Coltman, & Sharma, 2019).

Organisational agility, also called enterprise or business agility, is the ability to be flexible enough to speedily respond to customer requests, market dynamics, emerging technology options, and to adapt to a turbulent environment (Franco, Guimarães, & Rodrigues, 2023; Mathiassen & Pries-Heje, 2006; Ravichandran, 2018; Sambamurthy, Bharadwaj, & Grover, 2003) as well as to be stable enough to have sufficient processes, frameworks and structures to enable efficiency and productivity and avoid disorderly disintegration (Franco et al., 2023; Mathiassen & Pries-Heje, 2006). Although there is a growing body of research on organisational agility there remains a shortage of studies addressing the methodologies that support organisational agility (Franco et al., 2023).

One mechanism that has prominently been suggested to cope with the challenges produced by mounting complexity and rapid change is a second organisational operating system[1], which uses an agile and network-like structure and a very different set of processes than the traditional, hierarchical structure of an enterprise (Kotter, 2012). Kotter (2012) introduced the idea of a second operating system to address the need for organisations to simultaneously manage their ongoing operations via their first operating system, while undertaking significant transformations through a second operating system. This "[second] operating system continually assesses the business, the industry, and the organization, and reacts with greater agility, speed, and creativity than the existing one. It complements rather than overburdens the traditional hierarchy, thus freeing the latter to do what it's optimized to do" (Kotter, 2012). It is far from clear, however, what exactly such a second operating system should look like, how it should be set up, how such a system operates in detail, and if it increases organisational agility in the end. Thus to date Kotter's (2012) proposed concept of a second operating system, is largely uncorroborated.

Meanwhile, organisational research on environmental turbulence has long suggested boundary spanning as one important mechanism to cope with increasing complexity (Aldrich & Herker, 1977; Ancona & Caldwell, 1990; Keszey, 2018), and to manage interfaces between the organisation and its environment (Shi, Cui, & Kurnia, 2023; Tushman, 1977, 1979), as well as between organisational units (Ancona & Caldwell, 1990; Levina, 2005; Levina & Vaast, 2005). Moreover, in addition to boundary spanning, knowledge brokering between diverse occupational groups takes place at knowledge boundaries and is important for facilitating knowledge sharing (Argote & Miron-Spektor, 2011; Crupi, Del Sarto, Di Minin, Gregori, Lepore, Marinelli, & Spigarelli, 2020; Pawlowski & Robey, 2004). The concepts of boundary spanning and knowledge brokering are both closely related to networks which makes them a prime choice for conceptualising and operationalising the network-like structure of a second operating system. The concepts share the common goal of connecting disconnected members

[1] The concept of organisational operating systems formed by an organisation's structures and processes should not be confused with the technical concept of operating systems which denote the software programs that manage computer hardware and software resources and provide common services for a computer's programs.

across networks (Cross & Cummings, 2004; Fleming & Waguespack, 2007; Kwon, Rondi, Levin, De Massis, & Brass, 2020; Reagans & McEvily, 2003; Tortoriello, Reagans, & McEvily, 2011) but remain distinct wherein they describe different types of interaction (Fleming & Waguespack, 2007). Both concepts have not been investigated together in much detail, except for open innovation communities (Fleming & Waguespack, 2007), and neither has been investigated with regard to their relation to organisational agility so far. The current role and interplay of both for enabling organisational agility in the software industry is unclear, as is their role for operationalising the kind of second operating system suggested by Kotter (2012). Therefore, we address the following research question: *"How do boundary spanning and knowledge brokering enable organisational agility via a second operating system in the software industry?"*

In this paper, we present the results of an extensive case study of a software development company. We explore the role of boundary spanning and knowledge brokering in operationalising a second operating system in an organisational unit whose members engage in both knowledge brokering and boundary spanning between the company and its environment as well as within the company. We analyse the unit's and its members' approach and identify well-functioning activities and practices as well as challenges and drawbacks. As such, we contribute to an improved understanding of how an intertwined strategy of boundary spanning and knowledge brokering can establish and evolve a second operating system, allowing the first operating system, the traditional hierarchical structure, to focus on providing efficiency and stability in core operations. We thus provide empirical evidence of how a dual operating system can provide a balancing mechanism between flexibility and stability to support organisational agility in a software development company.

The remainder of the paper is structured as follows. The next section discusses the related work and theoretical background of our study, detailing the concepts of organisational agility, the dual operating system, boundary spanning and knowledge brokering. We then present results from the exploratory study of a specialised unit within a software development company and develop an explanation of how the knowledge brokering and boundary spanning activities of this unit, operating alongside its traditional hierarchical structure, enable the organisation to balance flexibility and stability and support organisational agility. We summarise and discuss our findings and conclude with an indication of limitations and an outlook on further research.

2 Related Work and Theoretical Background

2.1 Organisational Agility

In the face of intense competition, globalisation and time-to-market pressures, organisations must be agile, that is, have the capability to rapidly detect and seize market opportunities (Sambamurthy et al., 2003). Organisational agility is reported to enhance organisational performance overall, as well as in specific areas such as finance, supply chain management and innovation (Walter, 2021). In addition to focusing on the organisational outcomes of agility, research has concentrated on how agility should be defined and measured, along with its antecedents, moderators and mediators (see Nguyen, Le, Nguyen, Nguyen, Lien, & Nguyen, 2024; Walter 2021). In their recent literature review, Nguyen et al. (2024) identified various antecedents to encompass organisational capabilities such as knowledge management, IT and cooperation, coordination and collaboration. Moderators of the relationship between agility and organisational outcomes were found to include the business environment, market complexity, industry, technology, customer loyalty and other organisational factors such as leadership and firm size/age (Nguyen et al., 2024). Lastly they (Nguyen et al., 2024) grouped the mediators of agility and organisational outcomes within the following fifteen categories: technology, innovation, analytics capabilities, strategy and structure, competitiveness, social capital and networks, customers, market, marketing, supply chain, operation, leadership, workforce and work arrangement, knowledge and sustainability.

In the IS literature, researchers have been particularly interested in the relationship between IT capability (Lu & Ramamurthy, 2011), strategic IT alignment (Tallon & Pinsonneault, 2011) and IT investment (Sambamurthy et al., 2003) and organisational agility. With extensive interest in digital transformation, IS researchers have also highlighted how IT-enabled transformations support organisational agility (Vial, 2019). However, while it is often implied that firms need to be in a constant state of transformation, transformations are not always needed, may not be possible and come at a cost (Teece, Peteraf, & Leih, 2016). Instead, organisations in pursuit of agility need to balance flexibility with stability to ensure they can respond effectively to changes in the environment while simultaneously being efficient and productive. How organisations achieve this balance and develop organisational agility is a primary concern of this paper.

Investing in modular IT architectures which allow organisations to maintain core stability while responding rapidly to changes (Bharadwaj, El Sawy, Pavlou, & Venkatraman, 2013) is important for balancing flexibility and stability. It has also been suggested in the practitioner literature that a bimodal approach to IT strategy wherein one mode concentrates on stability and efficiency while the other focuses on flexibility and innovation is useful (Gartner, 2016; Haffke, Kalgovas, & Benlian, 2016). Organisational structure too, is important for balancing flexibility with stability (Kotter, 2012). This is because structure can determine how information flows, how decisions are made and the level of flexibility an organisation has. We consider Kotter's notion of the dual operating system for this purpose, next.

2.2 The Dual Operating System

Kotter (2012) introduces the concept of dual operating system. It is rooted in organisational theory which recognises that while traditional hierarchical structures are effective for managing routine operations and ensuring efficiency (Burns & Stalker, 1961), they often struggle with rapid innovation and adaptability (Mintzberg, 1979). In this way they compromise an organisation's pursuit of agility. In response, Kotter (2012) introduces the idea of a parallel network-like structure to operate alongside an organisation's traditional hierarchy as a second operating system. The structure is designed to be dynamic, allowing for quick decision-making and innovation while leaving the traditional hierarchy to optimise operational efficiency. Kotter (2012) describes the structure as comprising a "volunteer army" of employees across the entire organisation and up and down the ranks, who are empowered to drive change, innovate and respond quickly to challenges. Notably, a dual operating system need not start fully formed and can start in one part of an organisation and expand (Kotter, 2012). Overseeing the dual operating system should be a governance process that aligns the two operating systems with the strategic goals of the organisation and which reconciles any conflicts between the systems (Kotter, 2012). Kotter (2012) identifies eight accelerators to enable a dual operating system to function. These centre on a modern approach to change management and emphasise agility, rapid adaptation, and broad-based engagement, in contrast to the more rigid, controlled, and hierarchical nature of traditional management approaches. However, while the accelerators provide some direction for leaders embarking on a dual operating system, Kotter (2012) does not provide detailed insight into the underlying activities that members of the network-like structure need to undertake to support and utilise a dual operating system. As a key objective of the second operating system is to "liberate information from silos and hierarchical layers" and enable information to "flow with far greater freedom and accelerated speed" (Kotter, 2012), we examine the role of boundary spanning and knowledge brokering for operationalising the second operating system and their links with organisational agility.

2.3 Boundary Spanning and Knowledge Brokering

One critical implication of today's business context and its heightened emphasis on organisational agility, is the need to gather and harness internal and external knowledge to enhance organisational responsiveness and innovation. Boundary spanning and knowledge brokering are both concepts closely related to this goal.

Boundary spanning takes place at organisational boundaries and comprises external boundary spanning between an organisation and its environment as well as internal boundary spanning between different organisational units within the same organisation (Ancona & Caldwell, 1990; Levina, 2005; Levina & Vaast, 2005). Accordingly, boundary spanners are individuals, who are part of one organisational entity and who engage in boundary spanning activities (more on these activities later) towards organisational entities other than their own (Aldrich & Herker, 1977). Organisations that engage in effective boundary spanning are better able to sense changes in their external environment early, understand emerging trends and customer needs, and adapt quickly to external changes (Ancona & Caldwell, 1992).

Similarly, knowledge brokers facilitate the transfer of knowledge across knowledge boundaries (Fleming & Waguespack, 2007; Pawlowski & Robey, 2004; Pawlowski, Robey, & Raven, 2000). They work to connect individuals or groups, who possess different types of knowledge or expertise, help to translate and frame the elements of one group in terms of the perspective of another, therein enabling them to collaborate and exchange ideas effectively (Pawlowski & Robey, 2004). Knowledge brokering ensures valuable information is disseminated quickly across the organisation which enables faster decision-making, innovation, and adaptation to changing circumstances (Pawlowski & Robey, 2004).

In terms of their relationship with organisational agility, boundary spanning activities ensure the organisation is connected to external trends and changes, while knowledge brokering facilitates the translation, internal dissemination and application of this external knowledge, and together they allow the organisation to quickly adapt its strategies, products, or services based on external developments (Ancona & Caldwell, 1992; Pawlowski & Robey, 2004).

Both boundary spanning and knowledge brokering foster an environment where new ideas can emerge and be implemented swiftly. By bringing in diverse perspectives and insights, breaking down silos and promoting cross-functional collaborations, organisations can innovate more effectively in response to market changes or competitive pressures. By improving the flow of information and knowledge both externally and internally, boundary spanning and knowledge brokering reduce the time it takes for the organisation to detect changes and make decisions (Ancona & Caldwell, 1992; Pawlowski & Robey, 2004). This agility in decision-making is crucial for staying competitive in dynamic markets.

Boundary spanning and knowledge brokering are related concepts, at least regarding knowledge sharing across boundaries (Neal, Posner, & Brutzman, 2023). Also, both knowledge brokers and boundary spanners facilitate relationships, disseminate evidence, find alignment, build capacity, and inform decision making (Neal et al., 2023). While research has struggled to clearly delineate between the concepts and has identified there is overlap in the activities that boundary spanners and knowledge brokers undertake, there are differences (Pawlowski & Robey, 2004). We now turn our focus to the different boundary spanning and knowledge brokering activities that can potentially support a second operating system as proposed by Kotter (2012).

2.4 Boundary Spanning Activities

In their studies of new product development teams, Ancona and Caldwell (1990, 1992) identified and summarised boundary spanning activities as ambassadoring, coordinating tasks, scouting, and guarding (see Table 1, which also contains some examples for the activities). Ambassadoring covers buffering activities and representational activities.

Boundary Spanning Activity	**Description**	**Examples**
Ambassadoring	Includes buffering activities and representational activities of individuals within a team or department. This activity is somewhat political; identifying threats and opposition in top levels of the organisation and working to build support from these powerful outsiders.	Absorbing pressures and protecting the team Persuading others to support the team Keeping higher levels informed of team activities Informing the team of company strategy Informing the team of potential threats or opposition Lobbying for resources
Task coordination	Involves interactions aimed at coordinating technical or design issues. This activity represents specific efforts to coordinate and synchronise with other groups, typically those involved with the specific product or service.	Discussing design problems with others Obtaining feedback on the product design Coordinating and negotiating with outsiders
Scouting	Comprises activities that involve general scanning for ideas and information about the competition, the market, or the technology. This activity represents a more general scanning than task coordination, aimed at obtaining competitive, market, and technical ideas.	Undertaking research on market trends Exploring emerging technology trends Studying the strategies competitors are employing
Guarding	Involves controlling the team's release of information. It represents a means of decreasing, rather than meeting, dependence through the control of information flow out of the group.	Activities aimed at keeping information within the team's boundaries to protect the team Presenting a specific image of the team to outsiders

Table 1. Summary of Key Concepts for Boundary Spanning (Ancona & Caldwell, 1990; 1992)

These activities are somewhat political and contain both protective and persuasive goals such as obtaining personnel, funding, equipment, and legitimacy from management. Task coordination consists of interactions aimed at coordinating technical or design issues. This might involve a software designer liaising with marketing and sales about the scope of a software update and its associated timeline. Scouting involves general scanning for ideas and information about the competition, the market, or the technology and involves more general scanning than task coordination. Both task coordination and scouting manage the dependence on other functions or groups that have critical information, expertise, and creative ideas. Ancona and Caldwell (1992) note that scouting is best undertaken during the early life of a project and should not be an enduring pattern because if a team continues scouting without committing to a plan, it may have a detrimental impact on performance. Finally, guarding comprises controlling the team's release of information. This activity differs from the others in that it represents internal activities to keep things from the environment.

Boundary spanning involves political manoeuvring, management, and coordination as well as knowledge sharing activities (Ancona & Caldwell, 1998), which goes beyond Fleming and Waguespack's (2007) view, who see boundary spanning as primarily bridging technological boundaries. Studies on boundary spanning in various fields, including information systems development projects (Gasson, 2006; Gopal & Gosain, 2009; Levina & Vaast, 2005; Shi et al., 2023) confirm the applicability of this broader perspective on boundary spanning.

2.5 Knowledge Brokering Activities

In their studies of IT professionals, Pawlowski et al. (2000) and Pawlowski and Robey (2004) identified and summarised four knowledge brokering activities, namely, crossing boundaries, surfacing and challenging assumptions, translating and interpreting information and relinquishing ownership (see Table 2, which also contains some examples for the activities). Crossing boundaries involves not just crossing knowledge boundaries or social boundaries, which Fleming und Waguespack (2007) see as the primary boundaries that brokers are crossing, but also crossing organisational boundaries between units to share information and to leverage resources; it also includes the effort of gaining permission from business units to cross organisational boundaries that are closed to others. Surfacing and challenging assumptions involves stimulating reflection and change by asking 'why' as a means to uncover existing beliefs, practices and assumptions underlying the work of different divisions. (Pawlowski & Robey, 2004). Translating and interpreting information involves the framing of elements of the worldview of one group in terms of the perspective of another. This involves reframing, explaining, and clarifying information in the context of the work practice of one group to the context of another. Relinquishing ownership and maintaining a façade of objectivity includes the creation of the illusion that one is impartial and prepared to support any solution, even though a particular one is favoured. This might involve empowering users or transferring ownership of a project to business stakeholders while simultaneously maintaining influence over the outcomes. Overall, knowledge brokering focuses on knowledge sharing and processes of translation, coordination, and alignment between perspectives (Wenger, 1998).

Based on Ancona and Caldwell's (1990, 1992) categorisation of boundary spanning activities and Pawlowski and Robey's (2004) classification of knowledge brokering activities, in the following study we examine the knowledge brokering and boundary spanning activities undertaken by a dedicated organisational unit in a software development company as part of its wider strategy to balance flexibility and stability to support organisational agility via a dual operating system.

Knowledge Brokering Activity	Description	Examples
Crossing boundaries	Routinely crossing organisational boundaries between units, sharing information, and leveraging resources. This includes the exertion of considerable effort in gaining permission from business units to cross organisational boundaries that are closed to other groups.	Participating in Communities of Practice (CoPs) comprising members from different business units Managing relationships with IT vendors
Surfacing and challenging assumptions	Stimulating reflection and change by asking 'why' as a means to uncover existing beliefs, practices and assumptions underlying the work of different divisions.	Prompting discussion by promoting alternative solutions Challenging the technical feasibility of products to prompt critical discussions
Translation and interpretation	Framing elements of the worldview of one group in terms of the perspective of another.	Translating technical language into business terms Interpreting business needs for technical developers Facilitating cross-disciplinary communication
Façading / ostensibly relinquishing ownership	Creating the illusion that one is impartial and prepared to implement any solution requested by the user, even though a particular outcome is favoured (façade of impartiality).	Creating customer steering committees but guiding these through setting the agenda, disseminating the information, providing guidance and subtly influencing decisions. Soliciting customer feedback but only selectively incorporating this into the product or service design

Table 2. Summary of Key Concepts for Knowledge Brokering (Pawlowski and Robey 2004).

3 Research Design and Method

We conducted an exploratory, interpretive single in-depth case study to develop an understanding and theoretical explanation of how boundary spanning and knowledge brokering together enable organisational agility by operationalising the second operating system within the dual operating system of a software development company. The case organisation is a medium-sized German software solutions provider called SoftCorp (anonymised) that has undergone a transformation towards becoming a more agile and flexible organisation. The company is doing well in its dynamic environment with increasing numbers of sold licenses, acquired major customers and steadily rising customer satisfaction. The selected case organisation represents the main unit of analysis, but the individual departments, groups, and employees each represent analytical subunits, which allows for a multi-level analysis as called for by Marrone, Tesluk, and Carson (2007) and is in line with multi-perspective innovation studies (cf. Kautz & Nielsen, 2004 for a detailed argument).

We observed established guidelines for our research design (Miles & Huberman, 1994; Miles, Huberman, & Saldaña, 2014). The two main methods for data collection were interviews and observation (see Table 3). During a period of one year, we visited the company three times for one to three days each visit. We conducted open interviews with selected key informants, group interviews, and observations of activities. All interviews, which lasted from 30 to 120 minutes, and meetings were audio-recorded and transcribed. We followed a two-stage process of inductive and deductive coding of data, building upon and adapting the recommendations by Miles and Huberman (1994). First, two authors scrutinised and coded the data independently of each other. Inspired by previous work on boundary spanning (Ancona & Caldwell, 1990; 1992) and on knowledge brokering (Pawlowski & Robey, 2004; Pawlowski et al., 2000), we started with initial seed codes and searched separately for evidence of boundary spanning (see Table 1) and knowledge brokering (see Table 2) activities in relation to organisational agility. Each author also added new codes in exploring the data in more detail through coding events and activities that they found important (e.g., several new codes emerged related to task variety and uncertainty/variability in processes). Subsequently, the authors discussed their interpretations in person or using e-mail and teleconferencing.

Data Type	Description
Single interviews	10 participants (4x leaders Product Management Team, 1x developer Product Management Team, 1x manager Product Management Team, 1x Professional Services consultant, 1x manager Professional Services, 1x CEO, 1x manager Marketing)
Group interviews	8 participants (3x Product Management Team, 1x apprentice Product Management Team, 2x manager and leader Product Management Team, 1x lead developer and manager Core Development Team, 1x manager Product Management & Professional Services consultant)
Observations	1x Retrospective, 1x Review Meeting, Daily Stand-up Meetings on 3 days
Documentation	Product and company information (internal documents, reports, printouts)

Table 3. Overview of Data & Data Collection Methods

It is worth noting that we follow Alvesson and colleagues, who emphasise reflexivity as a key quality of rigorous thinking in the sense of identifying and challenging assumptions in the research literature and in research data (Alvesson Hardy, & Harley, 2008). Alvesson and Sandberg (2013) call for "a more active construction of empirical material in ways that are interesting, and not just waiting passively for data to show us the route to something interesting, as is typically the case in more conventional research". In contrast to the conventional concept of rigour they advocate "some relaxation of the pressure for the required standard in incremental gap-spotting research that emphasises rules, mechanics [of detailed data codification procedures] and data management" (Alvesson & Sandberg, 2013). This resulted in the following analysis.

4 Case Report and Analysis

4.1 Software Development at SoftCorp

SoftCorp is a German software development company, founded in 1999, it is now a subsidiary company of an exchange-traded IT service provider. SoftCorp employs around 70 staff at its headquarters, with about 90 employees in total and sales offices across Europe and the US. The core software product of SoftCorp is a Content Management System (CMS). Numerous companies from many industries worldwide use the CMS for managing their Internet presences as well as their intranet portals. Professionals from SoftCorp or selected partner companies provide consultancy and project services such as implementation, tailoring, or configuration of the CMS to end customers. SoftCorp's strategy as regards to the CMS is product-driven and not customer-driven:

"[…] the complete firm acts on the maxim or notion that we will not be driven by customer requests. We develop the product following our own vision […] in many areas, in many, many decisions, and for a long number of years, we simply didn't follow the trend on the market, which essentially is determined by Marketing or Sales, but we had our own thoughts about it." (Mike[2], member of the Executive Board)

As a result of this strategy, SoftCorp focuses on the development of a stable software core that is compatible between release versions. End customers need to run their own development projects, possibly supported by professionals from SoftCorp or partner companies, for customer-specific extensions of the CMS' core features. Such projects can range from simple extensions, so-called "modules", to complex web application projects. For example, modules allow the application of e-mail marketing, the display of content on mobile devices, or the integration with enterprise application servers. Most of the time, these extensions are not subsequently integrated into the core features of the CMS.

Figure 1 summarises SoftCorp's organisational structure. Consulting services and support for Internet and intranet projects of end customers are provided by consultants and project managers from the Professional Services unit (31 employees). The Research & Development unit (21 employees) comprises the development, quality assurance and internal IT teams. The Core Development Team at SoftCorp's headquarters (eight developers plus one manager) has exclusive responsibility for developing and maintaining the CMS as the core product.

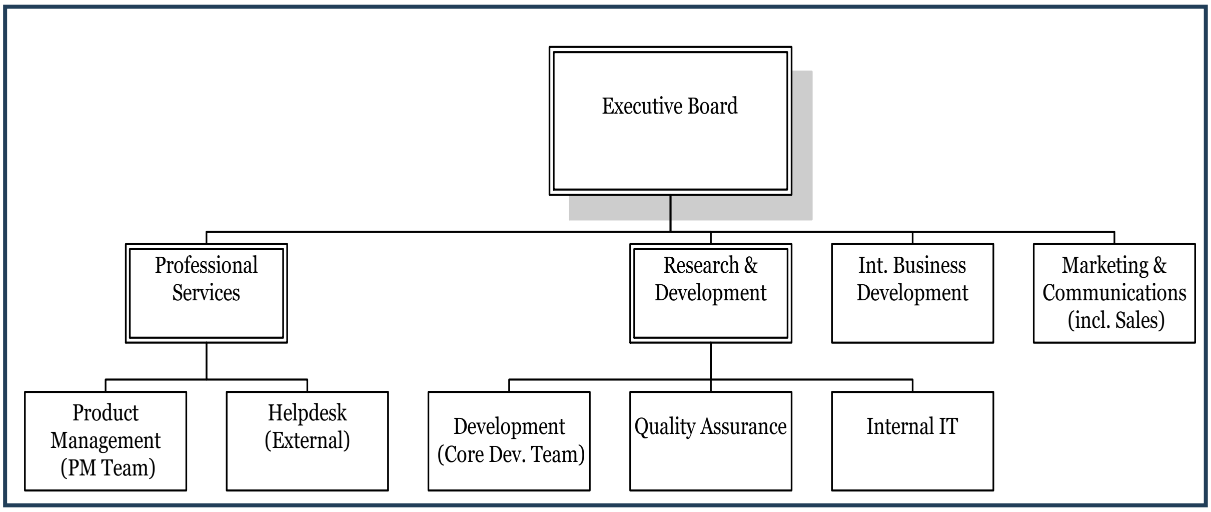
Executive Board
Professional Services
Research & Development
Int. Business Development
Marketing & Communications (incl. Sales)
Product Management (PM Team)
Helpdesk (External)
Development (Core Dev. Team)
Quality Assurance
Internal IT

Figure 1. Summary of Organisational Structure of SoftCorp

The Core Development Team continuously develops and advances the CMS. In general, a release cycle takes several months, resulting in a gap between major releases of several years. The development process follows a mixture of practices from traditional software engineering (e.g., variants of stage-gate models) and modern approaches (e.g., time boxing). The strategy to shield the Core Development Team from outside pressure, enabling them to develop the core product in a sustainable and stable way, results in a kind of wall to the external environment and to other departments:

[2] All names are anonymous for reasons of confidentiality.

"[…] Intentionally, we actually afford ourselves at the product side a small ivory tower, one can say that, an insular product development unit, many competitors work differently than that. And we actually make a point of doing so, that there exists this kind of wall […]" (Tom, Manager Professional Services)

Since spring 2011, the Product Management sub-unit (in the following: PM Team) of the Professional Services unit provides a second development team (six full-time employees and four apprentices). The PM Team is responsible for developing modules for the CMS that address specific noncore features (e.g., video management). The PM Team as a second development unit was created because of a felt need to be able to accelerate the development of modules to be able to react more quickly and faster to end customers' and partners' requests as well as to internal feature requests:

"The module development team [the PM Team] has been primarily brought into being in order to react quickly to requirements that come from the market because our development department simply cannot do that because they have firmly defined the work packages for the next one and a half years." (Charles, member of the PM Team)

Thus, a specialisation exists as regards software development. While the Core Development Team develops the CMS as a stable core product in a steady way, with a pace and time horizon of years between releases, the PM Team develops modules at a much faster pace, with a time horizon of months. Using two distinct development teams allows SoftCorp to react more quickly to customer demands, without having to jeopardise the stability of its core product. The PM Team also employs a different approach to software development than the Core Development Team, using a variation of Lean Software Development (Poppendieck & Poppendieck, 2003).

4.2 The PM Team: The Jack of All Trades

The PM Team is not only responsible for software development in the form of new or customer-specific software modules for the CMS. In parallel, a variety of other tasks are situated with the team (see Table 4), and it sometimes acts as an internal fire-fighter, or indeed a jack of all trades:

"[…] internally, the Product Management Team per se is an all-rounder and jack of all trades, and all sorts of people from all sorts of departments stop over all the time and want something to be done or made, have questions, and so forth." (Joe, apprentice in the PM Team)

As such, the team is approached by colleagues from other units such as Marketing & Communications and by external partners regarding a variety of topics related to the CMS. It thus provides internal product support and acts as an internal help desk. But it also develops show cases which involves the design and implementation of concepts and presentations for new CMS features and modules to support Marketing & Communications at end customer demonstrations or fairs. In addition, the team provides intranet care, and maintains and supports SoftCorp's intranet, both content-wise and infrastructure-wise. Furthermore, it supports the CMS community, where end customers and partner companies can exchange ideas, knowledge, and experiences with each other and with employees from SoftCorp using an Internet-based bulletin board. The PM Team answers questions regarding the CMS and provides content for community members respectively. Finally, three team members serve as stand-by personnel, and are occasionally used as consultants within end customer projects:

"[...] somebody such as Luke who is something of a 'jack of all trades', who can do everything. […] I don't need a developer, who sits there and looks at the code […] I need someone, who sits there and sometimes gives the developers a piece of one's mind." (Sean, Consultant Professional Services)

Task	Description
Software development	Development of new or customer-specific software modules for the CMS
Development of show cases	Design of concepts and presentations for new CMS features and modules, supporting Marketing & Communications, for example, for end customer demonstrations or fairs
Intranet care	Maintain and support SoftCorp's intranet, both content-wise and infrastructure-wise
Support of the CMS community	End customers and partner companies can exchange ideas, knowledge, and experiences with each other and with employees from SoftCorp using an Internet-based bulletin board; the PM Team answers questions regarding the CMS and provides content for the community members
Support as stand-by personnel	Three members of the PM Team occasionally act as stand-by personnel and are used by Professional Services as consultants within end customer projects
Internal product support	The PM Team acts as an internal helpdesk and is approached by colleagues (e.g. from Marketing & Communications) and by external partners regarding a variety of topics related to the CMS

Table 4. Overview of Tasks of the PM Team

4.3 Knowledge Brokering in the PM Team

Knowledge brokering by the PM Team involves crossing boundaries, translation and interpretation and surfacing and challenging assumptions. None of the interviewees described knowledge brokering activities that comprise façading or ostensibly relinquishing ownership.

The PM Team routinely crosses internal boundaries between other departments and units within SoftCorp. This results in the sharing of knowledge between units which would otherwise not talk to each other:

"[…] So these review meetings, we also conduct them in order to exemplify to others a little bit. Because in reality, we wish us the same from these people. That's one of the reasons why we do this, among other things. […] So that's one reason, because we would like to actually establish this kind of openness and this including of others throughout the company. And we are now the only ones who do it in this way." (Paul, manager of the PM Team)

The PM Team members have direct access to members of the Core Development Team, which members from other departments do not have. Due to the PM Team members' extensive technical knowledge of the CMS, when undertaking internal product support and developing show cases, the team often needs to translate and interpret the technical perspective of the Core Development Team and those of more business and customer-oriented departments such as Professional Services or Marketing & Communications, so they can each understand the others' views:

"[…] So to speak, we are internally the contact for all departments that have questions regarding the product. So when our presales or our sales have questions in any pitch situations, 'Can we do that? How is our position as regards … ?', then product management is asked [instead of the Core Development Team]." (Charles, member of the PM Team)

Specifically, during the development of modules for the CMS, the PM Team also engages in activities that surface and challenge assumptions. This includes both product-level (technical and business issues) as well as organisational-level topics. For example, as regards the former, the PM Team continuously questions the way things are done in the CMS during the development of modules, and tries to find new architectural solutions (e.g., challenging the pre-generation of content paradigm of the CMS in certain areas):

"Yes, yes, but as I said before, the reaction [to our new module] was very clear 'oh, we haven't done it that way for the last 10 years and it is just something new'. Well a completely new dimension, which had not been considered by the projects, as they until now always got away quite well with the existing solution." (Charles, member of the PM Team)

The PM team also regularly challenges the existing practices and ways of doing things at SoftCorp, for example, by actively choosing to manage projects differently than the Core Development Team, using Lean Software Development, and by being much more communicative than the Core Development Team.

4.4 Boundary Spanning in the PM Team

As regards to boundary spanning, the PM team presents a very interesting case, undertaking ambassadoring, coordinating, scouting, and guarding activities on behalf of SoftCorp between outsiders such as external partners and end customers as well as between insiders such as staff from Marketing & Communications, and other employees from Professional Services. When external partners or those from other SoftCorp units experience issues with the existing CMS the PM Team seeks to provide support. This can involve scouting for information inside the firm to answer questions that arise or scouting for new knowledge to identify ideas for new functionality and solutions. Moreover, the PM Team engages in ambassadoring activities during their tasks as stand-by personnel. This gives PM Team members the opportunity to inform partners of SoftCorp's product strategy and garner their support while simultaneously protecting the Core Development Team from outside pressure and informing them of potential threats and opportunities for product improvement:

"And then we have topics such as that you sometimes visit the partner and talk about the product or project decisions, and explore how you can improve the product so that developers and partners can work in a better way using it." (Luke, member of the PM Team)

The PM Team's development task developing modules for the CMS also involves frequent instances of coordinating activities regarding technical or design issues:

"For a new module, we always at the outset conduct a corresponding workshop, to which we invite people from Sales, from presales, from Professional Services, to jointly define requirements, at least roughly, or at least the expectations for the module as regards functionality." (Charles, member of the PM Team)

Similarly, other tasks show activities of scouting the external environment in relation to the competition, the market in general, or innovative technologies. For example, providing consulting support as stand-by personnel involves scouting:

"[…] and we have to look – we do quite some interrogation of third party products – what's up on the market, and this is a very different kind of Research & Development. A bit more looking outside, we have to look around a bit more and not just look how can we do that better, how can we optimise, but it's really, more, yes, putting on the market glasses." (Paul, manager of the PM Team)

Finally, as regards guarding, the PM Team responds quickly to staff, especially to customer and market demands, and thus shields the Core Development Team from external and internal contacts so that the latter can concentrate on developing the CMS:

"While the Core Team says 'we have to defend the core asset of the firm for the next ten years, enhance and develop it in a clean way, and create a stable foundation on which customers can rely on using many, many interfaces in all kinds of directions', the PM Team is the more agile division, which can react considerably faster to a customer request, which can react considerably faster – that is also important – if we see opportunities within the market" (Mike, member of the Executive Board)

4.5 Organisational Agility: Balancing Flexibility and Stability through a Dual Operating System

SoftCorp balances flexibility and stability based on market expectations and demands. The company does this with what we identify as a dual operating system: the first operating system is reflected in a traditional organisational hierarchy and provides stability by allowing the Core Development Team to design the CMS with a long-term vision operationalised by the original, shielded development unit while the second operating system is designed to react more quickly and nimbly to market, staff, and customer requests operationalised by the boundary spanning and knowledge brokering activities and tasks of the PM Team:

"[…] an important factor for the core product is the market expects that you to have at least a 3 years development roadmap; if you don't have it you have a problem. As for the modules, we look maximum 12 months forward into the future, that's it. If we would be forced to plan as long term [as the Core Development team] we would be in the same pattern." (Paul, manager of the PM Team)

The individual members of the PM Team, who are organised in this unit have business as well as detailed technical knowledge of the core CMS product, its software code, and the CMS implementation projects because of their diverse tasks. Gathering these skill sets in one dedicated organisational unit created a broad knowledge base which is considered beneficial for the organisation:

"Previously, we didn't have that department. And we had very big problems because, yes, because this knowledge was not focused at one single point, so to speak. […] I believe that this unit contributes very strongly to the fact that we can develop a good, broad knowledge and a good view of the company and for the product itself and for the market as such, yes." (Paul, manager of the PM Team)

Against this background the PM Team undertakes diverse forms of knowledge brokering and boundary spanning activities between SoftCorp's different units and towards SoftCorp's environment. It was described by one member of the PM Team as the interface between all departments and having important influence on the direction of SoftCorp's product development. Importantly, having the PM Team focus on responding to market, staff, and customer requests provides the Core Development Team with a solidity that has had a positive effect on customer satisfaction:

"This deceleration [i.e., shielding the Core Development Team] has meant that customer satisfaction has increased massively, it now has much less problems, I'm also helpdesk manager, the number of tickets has clearly decreased." (Tom, manager Professional Services)

Boundary spanning and knowledge brokering as performed by committed individuals, who are organised in a dedicated organisational unit in SoftCorp then, contributes to increased organisational agility by (1) enabling faster responses to customer requests and influences from the market and external environment; (2) spotting opportunities faster; and (3) augmenting speed-to-market by shielding and taking pressure off the Core Development Team to enable it to pursue a sustainable long-term vision and allowing for a steady development pace that delivers a stable core product.

The boundary spanning and knowledge brokering activities undertaken by the PM Team are key in operationalising a second operating system at SoftCorp and thus supporting the company to balance flexibility and stability via a dual operating system.

As for boundary spanning, ambassadoring is reflected in the PM Team's engagement in a range of activities for SoftCorp, especially with the internal and external community tasks, their own development tasks, and as stand-by personnel in customer projects. With respect to coordinating tasks, this is evident when the PM Team takes on various activities related to managing design and/or technical issues again through their performance of development tasks, their community tasks, and their role as stand-by personnel. In the identified scouting activities, the PM Team explores the external environment, the market in general, their competitors as well as novel and innovative technologies on behalf of SoftCorp during their work tasks. Finally, when guarding SoftCorp from outside and inside influences so that the Core Development Team can concentrate on developing the core of the CMS, the PM Team through their development work, performance of community tasks, and also as stand-by personnel plays a significant role in balancing stability and flexibility and thus SoftCorp's agility.

As for knowledge brokering, in terms of crossing boundaries, the PM Team routinely crosses internal boundaries between other departments and units within SoftCorp, which would otherwise not talk to each other. Regarding surfacing and challenging assumptions, the PM Team regularly challenges the existing product design decisions as well as organisational practices and ways of doing things at SoftCorp. Lastly, when engaging in translation and interpretation the PM Team frequently outlines elements of the technical views of the Core Development Team regarding the core CMS product in terms of the perspective of other units and departments such as Professional Services, Marketing & Communications, and it also translates market, customer and staff requirements to technical concepts, which they either themselves develop into modules or pass on to the Core Development Team.

The role of the PM team as dedicated unit that engages in both knowledge brokering and boundary spanning between other departments and between partners, end customers and the market, thus contributes to SoftCorp's organisational agility. The tasks and activities of the PM Team operationalise a second operating system, which provides flexibility as it reacts to changes in a much more accommodating way than the other SoftCorp units, and is more dialogical and interactive, both with the outside environment and with other units. It neither only conducts knowledge brokering nor boundary spanning; it does both in an intertwined way. The other units found in the traditional organisational hierarchy, the support functions and Core Development Team, operationalise the first operating system, wherein the Core Development Team is intentionally shielded from outside and inside influences, and only has minimal contact to other units and the external environment. This increases SoftCorp's stability as it allows for a sustainable development of the core product with a long-term vision. SoftCorp's approach to a dual operating system for organisational agility is summarised in Figure 2.

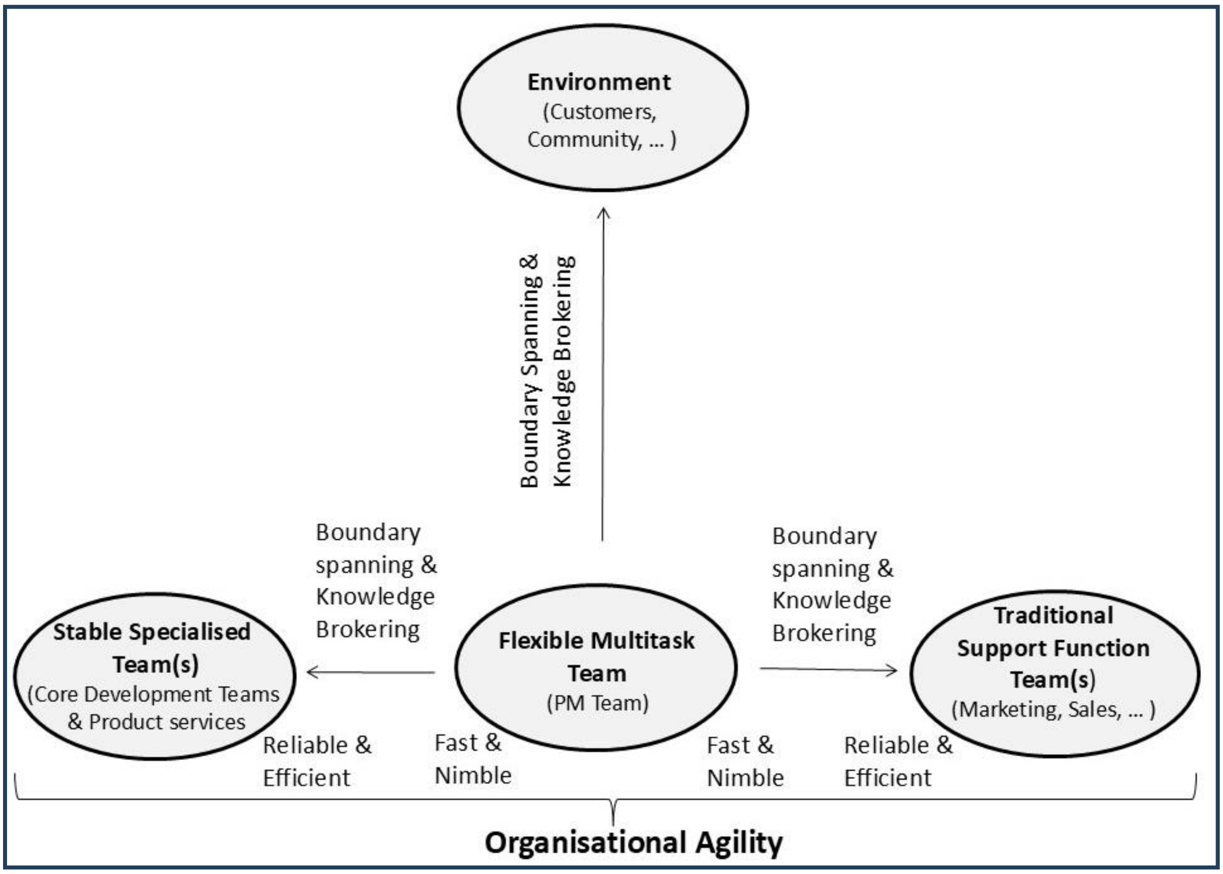
Environment
(Customers, Community, ...)
Boundary Spanning & Knowledge Brokering
Boundary spanning & Knowledge Brokering
Boundary spanning & Knowledge Brokering
Stable Specialised Team(s)
(Core Development Teams & Product services
Flexible Multitask Team
(PM Team)
Traditional Support Function Team(s)
(Marketing, Sales, ...)
Reliable & Efficient
Fast & Nimble
Fast & Nimble
Reliable & Efficient
Organisational Agility

Figure 2. SoftCorp’s Dual Operating System for Organisational

4.6 Benefits and Drawbacks of the Second Operating System

The boundary spanning and knowledge brokering activities of the PM Team in the second operating system clearly have several positive effects for SoftCorp as a whole. Together, they enable the PM Team to undertake agile development of modules using lean development, allowing for stable development of the core product with a clear, long-term vision by the Core Development Team in the first operating system. By undertaking a variety of boundary spanning and knowledge brokering activities, the PM Team contributes to organisational agility by acquiring knowledge on broad topics from diverse sources enabling it to react fast to changes in the environment. The manager of the PM Team advocated that this enables the team to develop a broad understanding of business and product and to stay in touch with the company vision in a way that would not be possible if the team were only focused on development. However, while the activities of the PM Team produce a range of benefits, they come at a cost and have a number of observed drawbacks. We summarise these drawbacks here also as a note of caution for organisations that may consider a similar approach.

Although it helps to accumulate knowledge in diverse fields, the variety of different potential units that want to contact the PM Team sometimes leads to overload as regards to the capacity of the PM Team to handle all tasks and requests. Thus, a fundamental problem for enhancing the efficiency of the PM Team, and thus the second operating system, is the variety of the tasks that are executed besides developing modules. Each task domain is a business process, with a separate input, output, and flow unit, and is completely different as regards to its characteristics. For example, developing modules is different from developing show cases, or from supporting the community on the online bulletin board. Furthermore, demand for each task is independent from the demand of other tasks:

"[…] The disadvantage is, of course, that all come running to us and always want to know something, yes, of course, this is the flip side of the coin. You somehow have to steer this into regulated channels or defend yourself in a sufficiently pushy way, as the case may be [laughs]." (Paul, manager of the PM Team)

This task variety, with different goals, in turn leads to more variability and uncertainty in the development process and in all other tasks respectively. This becomes apparent when looking at how the PM Team organises its work: using one global Kanban board with the same activities for all tasks, even if every task is in fact a separate process. For example, sometimes the prioritisation of other tasks such as the building of show cases or the use of PM Team members as stand-by personnel leads to bottlenecks and waste and cases where the development of modules is put on hold because developers are missing:

"[on the role of stand-by personnel:] Yes, all right. And Luke was now, I would say; the last resort because he actually works in PM and actually has other tasks, but the other expert we wanted to have, first, he was on vacation and second, he also is a highly sought-after person with use here in the house." (Sean, Consultant Professional Services)

"[…] So we have to cut somewhere, consequentially […] which can mean that before CeBIT [trade fair]*, we hardly do any module development, so we can a little bit absorb this, internally, so almost no module development, but show cases, and accordingly vice versa, after CeBIT."* (Charles, member of the PM Team)

Table 5 summarises benefits and drawbacks of SoftCorp's second operating system through the PM Team.

Benefits	**Drawbacks**
• Quick, agile development of modules using lean development (Kanban) while allowing for stable development of core product with a clear, long-term vision • Create business agility and react fast to changes in the environment • Acquisition of broad knowledge from a wide variety of tasks • Increased knowledge brokering between other internal units and external parties • Spanning of internal boundaries and boundaries with external end customers and partners	• Variety of tasks leads to uncertainty and variability in the processes • Uncertainty and variability in the processes lead to bottlenecks • Variety of tasks clashes with lean management approach

Table 5. Summary of Benefits and Drawbacks of SoftCorp's Operationalisation of the Second Operating System

5 Discussion and Conclusion

We performed a study that, building on Kotter's work on dual organisational operation systems (Kotter, 2012), investigated how the boundary spanning and knowledge brokering activities rooted in a dedicated unit operationalise a second operating system, which when operating alongside a traditional hierarchical structure, the first operating system, enable the organisational agility of a software development company.

We established earlier in Section 1, that the study is further based on an understanding that the concepts of boundary spanning and knowledge brokering share the common goal of connecting disconnected organisational members across networks. Thus, they are closely related to the network concept. This makes them a prime choice for conceptualising and operationalising the network-like structure of a second operating system as proposed by Kotter (2012).

In the case setting, the boundary spanning and knowledge brokering activities supplement each other and contribute to the flexibility required to enable the organisation to respond rapidly to environmental changes, while allowing the first operating system to focus on providing efficiency and stability in core operations. Thus, when part of a dual operating system, boundary spanning and knowledge brokering help balance flexibility and stability and support organisational agility.

The results therefore suggest that the boundary spanning and knowledge brokering activities undertaken by the PM Team in the second operating system serve an important function for SoftCorp. Through their enactment the PM Team offers an important structure to help balance the demand for more flexibility in a dynamic and increasingly complex environment and the demand for stability that is needed for the efficient development of the core product. This is aligned with the paradoxical approach to managing agility advocated by Lewis, Andriopoulos, and Smith (2014) which requires managing stability while having enough flexibility to be responsive to a rapidly changing situations. The boundary spanning and knowledge brokering activities of the PM Team provide a mechanism through which SoftCorp can act more quickly and make sense of changes in its environment, at the same time shielding the Core Development Team from these influences. Importantly, the PM Team's dual roles of boundary spanning and knowledge brokering are both necessary for maintaining SoftCorp's viability. This helps with keeping the Core Development Team isolated in order to develop the CMS in a stable, slow-paced manner and allows SoftCorp to react quickly to changes in its environment. This dual strategy of stable core product vs. fast modules enables SoftCorp to support organisational agility and is a basis for their success.

Our study presents an empirical contribution (Ågerfalk, 2014) in the form of a novel account of an empirical phenomenon – organisational agility – that reveals how the balance between flexibility and stability, which characterises organisational agility, can be achieved by the boundary spanning and knowledge brokering activities undertaken by a dedicated unit operating alongside a traditional hierarchical structure. This has been previously undocumented. We reveal insights into this phenomenon, while we rely explicitly on the a priori conceptualisations of boundary spanning and knowledge brokering. This has three important theoretical implications for further research.

First, previous literature argues for the importance of such specialised units for balancing flexibility and stability (e.g., Aghina, De Smet, & Weerda, 2015; Baškarada & Koronios, 2018; Harter & Krone, 2001; Kotter, 2012). The data reported here support these arguments. Yet, in contrast to Harter and Krone (2001), who revealed the role of a cooperating support organisation acting as a boundary spanner and thereby helping the cooperative's member organisations to balance change and stability, we found that our case organisation has a dedicated team inside its own organisation that contributes to balancing flexibility and stability by undertaking boundary spanning and knowledge brokering activities. As such, our work supplements the work of Kotter (2012), who advocates to extend traditional organisational units with a second operating system that uses an agile, network-like structure and different processes to assess the business, the industry, and the organisation, and that reacts with greater nimbleness, speed, and creativity than the existing units. Kotter (2012), however, does not consider how knowledge in such structures is shared with the organisation, and does not go into detail on how such a structure operates and looks like. We provide evidence for such a possible structure in the form of a dedicated unit and illuminate how knowledge brokering and boundary spanning underpin this. In doing so, we show that a second operating system need not be a parallel network-like structure comprised of volunteers but can be a part of the formal organisational hierarchy (i.e., the first operating system), yet by the distinct nature of the activities undertaken, it is a second operating system. Further research could investigate other structural configurations and examine the mutually reinforcing role of the concepts.

Second, while the literature provides evidence for boundary spanning and knowledge brokering as two separate sets of activities (e.g., Levina & Vaast, 2005; Marrone et al., 2007; Pawlowski & Robey, 2004), we extend, based on our case, the common body of knowledge and establish that a unit that intertwines these activities can usefully contribute to organisational agility. Having a dedicated unit in a second operating system, with members with complimentary knowledge and skill sets in order to be able to both span boundaries and broker knowledge between others, is a promising strategy for other companies that also want to achieve organisational agility. In previous work, Fleming and Waguespack (2007) studied brokering and spanning together, however in open innovation communities, which are different from individual commercial companies. We provide an empirical confirmation for their proposition that boundary spanning and knowledge brokering are closely related in such a setting, too. They also put forward that brokers can perform spanning activities and spanners can perform brokering activities but found that performing these activities simultaneously can have negative consequences with regard to the individuals' roles in these communities. This might be related to the environment in which they performed their study and their focus on individuals. We do not find these negative effects; on the contrary, both sets of activities in our setting contribute to organisational agility without jeopardising the individuals involved. However, we did find drawbacks in the second operating system for the PM Team as a whole, resulting from the increasing task variety and the resulting variability and uncertainty. Our study did not explore the impact of these effects on organisational agility. Further studies should analyse negative effects in more detail and could investigate whether promising strategies exist for mitigating negative effects while retaining positive ones.

Third, taking a starting point in individuals acting as knowledge brokers, others have also struggled with the conceptual distinction of knowledge brokering and boundary spanning (Pawlowski & Robey, 2004; Wenger, 2000). As we focus on an organisational unit as a whole, this difficulty, as important as the distinction is for our analysis, does not jeopardise the results of our study. We provide some new insights about the characteristics of knowledge brokers and boundary spanners. The PM Team as a whole performs boundary spanning and knowledge brokering activities. The PM Team's ability to act as a dual bridging unit crucially depends on the different and complementary skills of its members. Reportedly, the PM Team was established in light of the special skills required by its members to fulfill this crucial position within the company. Moreover, both boundary spanning and knowledge brokering are conducted in relation to external partners and customers as well as to other internal units. Thus, the PM Team is neither only conducting internal knowledge brokering nor external boundary spanning. While the literature provides evidence for boundary spanning and brokering as two separate sets of activities, which are performed by individuals (Kimble, Grenier, & Goglio-Primard, 2010; Levina & Vaast, 2005; Marrone et al., 2007; Neal et al., 2023;. Pawlowski & Robey, 2004; Tortoriello et al., 2011; Mahnke, Wareham, & Bjoern-Andersen, 2008), we extend, based on our case, the common body of knowledge and establish that these activities beyond the individual level also take place in an integrated manner on the unit level. Although the individuals are important, it is the unit as a whole, which makes the difference in our case organisation. More work must be done to understand the idiosyncratic and the common characteristics of the involved sets of activities.

We focused on boundary spanning and knowledge brokering from the outset and did not employ other lenses for scrutinising our data, e.g., the role of boundary objects (Star, 2010; Star & Griesemer, 1989) or the role of different technology frames (Leonardi & Barley, 2010; Orlikowski & Gash, 1994) surrounding the various units, which may play a role in this setting as well. These offer a worthy subject for future research.

As a contribution to practice, our findings may help other companies to decide whether having a dedicated knowledge brokering and boundary spanning unit in a second operating system is worthwhile in their context to enable organisational agility. Having a second operating system rooted in a dedicated unit comprising members with complimentary skill sets and knowledge sets that can act as a knowledge reservoir to be able to span or broker, operating alongside a traditional organisational hierarchy, is a promising strategy for balancing flexibility with stability to achieve organisational agility. Our findings indicate that the antecedent conditions for this are to have (a) individuals with skills and competence necessary for knowledge brokering and boundary spanning, and (b) a variety of tasks that are necessary to develop a broad set of knowledge.

However, the role of such a unit comes with a cost. This is mostly observable in terms of the high uncertainty and variability in the PM Team's business processes, which led to bottlenecks and standstills in the software development process. It is also somewhat in conflict with a lean management approach as taken up for software development by the PM Team, which usually has the ultimate goal of eliminating bottlenecks and eliminating waste (Anupindi, Chopra, Deshmukh, Van Mieghem, & Zemel, 2006; Poppendieck & Poppendieck, 2003). Thus, we observe a trade-off: on the one hand, the wide diversity and variety of tasks allows the members of the PM Team to acquire the skill set and the set of broad knowledge through both knowledge brokering and boundary spanning. Without the tasks, the PM Team would not interface as frequent or as much with other units and with customers or partners.

On the other hand, the same variety leads to increased variability and a resulting increased chance for disruptions in the workflow of the PM Team. The task variety leads to coordination problems, for example, visible in the necessity to assign time contingents to tasks or in the fact that it takes quite a while to develop a module for a team working in an agile way. Even though process variation may be related to performance gains or even to organisational survival in dynamic, competitive environments where flexibility is important (Pentland, 2003a), too much emphasis on too many tasks at once may lead to increases in variability and bottlenecks in the process flow (Anupindi et al., 2006), and thus, in turn, may even compromise the intended strategy to use a dedicated team or unit such as the PM Team to operationalise a second operating system be able to react more quickly and be flexible to changes in the environment (Pentland, 2003a, 2003b).

6 References

Ågerfalk, P. J. (2014). Insufficient theoretical contribution: a conclusive rationale for rejection? *European Journal of Information Systems, 23*(6), 593-599.

Agerfalk, P. J., Fitzgerald, B., & Slaughter, S. A. (2009). Introduction to the Special Issue--Flexible and Distributed Information Systems Development: State of the Art and Research Challenges. *Information Systems Research, 20*(3), 317-328. doi.org/10.1287/isre.1090.0244

Aghina, W., De Smet, A., & Weerda, K. (2015). Agility: It rhymes with stability. Retrieved from https://www.mckinsey.com/business-functions/organization/our-insights/agility-it-rhymes-with-stability

Aldrich, H., & Herker, D. (1977). Boundary spanning roles and organization structure. *Academy of Management Review, 2*(2), 217-230.

Alvesson, M., Hardy, C., & Harley, B. (2008). Reflecting on reflexivity: reappraising practice. *Journal of Management Studies* (45). 480–501.

Alvesson, M., & Sandberg, J. (2013). Has Management Studies Lost Its Way? Ideas for More Imaginative and Innovative Research. *Journal of Management Studies* (50), 128-152.

Ancona, D. G., & Caldwell, D. F. (1990). Beyond boundary spanning: Managing external dependence in product development teams. *The Journal of High Technology Management Research, 1*(2), 119-135.

Ancona, D. G., & Caldwell, D. F. (1992). Bridging the Boundary: External Activity and Performance in Organizational Teams. *Administrative Science Quarterly, 37*(4), 634-665. doi:10.2307/2393475

Ancona, D. G., & Caldwell, D. F. (1998). Rethinking team composition from the outside in. In D.H. Grueneld (Ed.), *Research on Managing Groups and Team. Vol 1, Composition,* 21-37.Amsterdam, The Netherlands: Elsevier.

Anupindi, R., Chopra, S., Deshmukh, S. D., Van Mieghem, J. A., & Zemel, E. (2006). *Managing Business Process Flows. Principles of Operations Management* (2 ed.). Upper Saddle River, NJ, USA: Pearson Education.

Argote, L., & Miron-Spektor, E. (2011). Organizational Learning: From Experience to Knowledge. *Organization Science, 22*(5), 1123-1137. doi.org10.1287/orsc.1100.0621

Baham, C., & Hirschheim, R. (2022). Issues, challenges, and a proposed theoretical core of agile software development research. *Information Systems Journal, 32*(1), 103-129. doi.org/10.1111/isj.12336

Baškarada, S., & Koronios, A. (2018). The 5S organizational agility framework: a dynamic capabilities perspective. *International Journal of Organizational Analysis, 26*(2), 331-342. doi.org/10.1108/IJOA-05-2017-1163

Benbya, H., Nan, N., Tanriverdi, H., & Yoo, Y. (2020). Complexity and information systems research in the emerging digital world. *MIS Quarterly, 44*(1), 1-17.

Bharadwaj, A.,. El Sawy, O. A., Pavlou, P. A., & Venkatraman, N. (2013). Digital Business Strategy: Toward a Next Generation of Insights. *MIS Quarterly*, 37(2), 471-482.

Brynjolfsson, E., & McAfee, A. (2014). *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. New York, NY, USA: W.W. Norton & Company.

Burns, T., & Stalker, G. M. (1961). *The Management of Innovation*. Social Science Paperbacks. London, UK: Tavistock.

Cross, R., & Cummings, J. N. (2004). Tie and Network Correlates of Individual Performance in Knowledge-Intensive Work. *Academy of Management Journal, 47*(6), 928-937. doi.org10.2307/20159632

Crupi, A., Del Sarto, N., Di Minin, A., Gregori, G. L., Lepore, D., Marinelli, L., & Spigarelli, F. (2020). The digital transformation of SMEs – a new knowledge broker called the digital innovation hub. *Journal of Knowledge Management, 24*(6), 1263-1288. doi:10.1108/JKM-11-2019-0623

Fleming, L., & Waguespack, D. M. (2007). Brokerage, boundary spanning, and leadership in open innovation communities. *Organization Science, 18*(2), 165-180.

Franco, M., Guimarães, J., & Rodrigues, M. (2023). Organisational agility: systematic literature review and future research agenda. *Knowledge Management Research & Practice, 21*(6), 1021-1038. doi.org/10.1080/14778238.2022.2103048

Gasson, S. (2006). A genealogical study of boundary-spanning IS design. *European Journal of Information Systems, 15*(1), 26-41.

Gopal, A., & Gosain, S. (2009). The Role of Organizational Controls and Boundary Spanning in Software Development Outsourcing: Implications for Project Performance. *Information Systems Research, 20*(8), 1-23.

Haffke, I., Kalgovas, B., & Benlian, A. (2016), Options for Transforming the IT Function Using Bimodal IT, *MIS Quarterly Executive*, 16(2), 101-119.

Harris, M. L., Collins, R. W., & Hevner, A. R. (2009). Control of Flexible Software Development Under Uncertainty. *Information Systems Research, 20*(3), 400-419. doi.org/10.1287/isre.1090.0240

Harter, L., & Krone, K. (2001). The boundary-spanning role of a cooperative support organization: managing the paradox of stability and change in non-traditional organizations. *Journal of Applied Communication Research, 29*(3), 248-277. Retrieved from doi.org/10.1080/00909880128111

Kautz, K., & Nielsen, P. A. (2004). Understanding the implementation of software process improvement innovations in software organizations. *Information Systems Journal, 14*(1), 3-22. Retrieved from doi.org/10.1111/j.1365-2575.2004.00156.x

Keszey, T. (2018). Boundary spanners' knowledge sharing for innovation success in turbulent times. *Journal of Knowledge Management, 22*(5), 1061-1081. doi.org/10.1108/JKM-01-2017-0033

Kimble, C., Grenier, C., & Goglio-Primard, K. (2010). Innovation and knowledge sharing across professional boundaries: Political interplay between boundary objects and brokers. *International Journal of Information Management, 30*(5), 437-444.

Kotter, J. P. (2012). Accelerate! *Harvard Business Review, 90*(11), 43-58.

Kwon, S.-W., Rondi, E., Levin, D. Z., De Massis, A., & Brass, D. J. (2020). Network Brokerage: An Integrative Review and Future Research Agenda. *Journal of Management, 46*(6), 1092-1120. Doi.org/10.1177/0149206320914694

Laato, S., Mäntymäki, M., Islam, A. K. M. N., Hyrynsalmi, S., & Birkstedt, T. (2023). Trends and Trajectories in the Software Industry: implications for the future of work. *Information Systems Frontiers, 25*(2), 929-944. doi:10.1007/s10796-022-10267-4

Leonardi, P. M., & Barley, S. R. (2010). What's under construction here? Social action, materiality, and power in constructivist studies of technology and organizing. *Academy of Management Annals*, *4*(1), 1-51.

Levina, N. (2005). Collaborating on Multiparty Information Systems Development Projects: A Collective Reflection-in-Action View. *Information Systems Research, 16*(2), 109-130. Retrieved from 10.1287/isre.1050.0055 http://search.ebscohost.com/login.aspx?direct=true&db=buh&AN=17527638&amp;lang=de&site=ehost-live

Levina, N., & Vaast, E. (2005). The emergence of boundary spanning competence in practice: implications for implementation and use of information systems. *MIS Quarterly, 29*(2), 335-363.

Lewis, M. W., Andriopoulos, C., & Smith, W. K. (2014). Paradoxical leadership to enable strategic agility. *California Management Review, 56*(3), 58-77.

Lu, Y., & Ramamurthy, K. (2011). Understanding the link between information technology capability and organizational agility: An empirical examination. *MIS Quarterly*, *35*(4), 931-954.

Mahnke, V., Wareham, J., & Bjoern-Andersen, N. (2008) Offshore Middlemen: Transnational Intermediation in Technology Sourcing. *Journal of Information Technology*, 23(1), 18-30. doi: 10.1057/palgrave.jit.2000124

Marrone, J. A., Tesluk, P. E., & Carson, J. B. (2007). A Multilevel Investigation of Antecedents and Consequences of Team Member Boundary-Spanning Behavior. *Academy of Management Journal, 50*(6), 1423-1439. Retrieved from http://amj.aom.org/content/50/6/1423.abstract

Mathiassen, L., & Pries-Heje, J. (2006). Business agility and diffusion of information technology. *European Journal of Information Systems, 15*(2), 116-119. Retrieved from doi.org/10.1057/palgrave.ejis.3000610

Miles, M. B., & Huberman, A. M. (1994). *Qualitative Data Analysis: A Sourcebook of New Methods*. Beverly Hills, CA, USA: Sage.

Miles, M. B., Huberman, A. M., & Saldaña, J. (2014). *Qualitative Data Analysis: A Methods Sourcebook* (3 ed.). London, UK: Sage.

Mintzberg, H. (1979). *Structuring of Organizations.* London: UK: Pearson.

Neal, J. W., Posner, S., & Brutzman, B. (2023). Understanding brokers, intermediaries, and boundary spanners: a multi-sectoral review of strategies, skills, and outcomes. *Evidence & Policy, 19*(1), 95-95–115. doi.org/10.1332/174426421X16328416007542

Nguyen, T., Le, C. V., Nguyen, M., Nguyen, G., Lien, T. T. H., & Nguyen, O. (2024). The organisational impact of agility: a systematic literature review. *Management Review Quarterly*, 1-49. doi.org/10.1007/s11301-024-00446-9

Orlikowski, W. J., & Gash, D. C. (1994). Technological frames: making sense of information technology in organizations. *ACM Transactions on Information Systems (TOIS)*, *12*(2), 174-207.

Overby, E., Bharadwaj, A., & Sambamurthy, V. (2006). Enterprise agility and the enabling role of information technology. *European Journal of Information Systems, 15*(2), 120-131. Retrieved from 10.1057/palgrave.ejis.3000600. http://search.ebscohost.com/login.aspx?direct=true&db=buh&AN=28042003&lang=de&site=ehost-live

Pawlowski, S., & Robey, D. (2004). Bridging User Organizations: Knowledge Brokering and the Work of Information Technology Professionals. *MIS Quarterly, 28*(4), 645-672.

Pawlowski, S. D., Robey, D., & Raven, A. (2000, December 10-13). *Supporting shared information systems: boundary objects, communities, and brokering*. Paper presented at the ICIS 2000 Proceeding, Brisbane, Australia.

Pentland, B. T. (2003a). Conceptualizing and Measuring Variety in the Execution of Organizational Work Processes. *Management Science, 49*(7), 857-870.

Pentland, B. T. (2003b). Sequential Variety in Work Processes. *Organization Science, 14*(5), 528-540. Retrieved from http://orgsci.journal.informs.org/cgi/content/abstract/14/5/528

Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit* (1st ed.). Amsterdam, The Netherlands: Addison-Wesley Longman.

Ravichandran, T. (2018). Exploring the relationships between IT competence, innovation capacity and organizational agility. *The Journal of Strategic Information Systems, 27*(1), 22-42. doi.org/10.1016/j.jsis.2017.07.002

Reagans, R., & McEvily, B. (2003). Network Structure and Knowledge Transfer: The Effects of Cohesion and Range. *Administrative Science Quarterly, 48*(2), 240-267. doi:10.2307/3556658

Sambamurthy, V., Bharadwaj, A., & Grover, V. (2003). Shaping Agility through Digital Options: Reconceptualizing the Role of Information Technology in Contemporary Firms. *MIS Quarterly, 27*(2), 237-263. Retrieved from http://www.jstor.org/stable/30036530

Shi, Y., Cui, T., & Kurnia, S. (2023). Value co-creation for digital innovation: An interorganizational boundary-spanning perspective. *Information & Management, 60*(5), 103817. doi.org/10.1016/j.im.2023.103817

Sommerville, I., Cliff, D., Calinescu, R., Keen, J., Kelly, T., Kwiatkowska, M., Mcdermind, J.,. & Paige, R. (2012). Large-scale complex IT systems. *Communications of the ACM, 55*(7), 71-77. doi:10.1145/2209249.2209268

Star, S. (2010). This is not a boundary object: Reflections of the origin of a concept. S*cience, Technology, & Human Values*, 35, 601-617. doi:10.1177/0162243910377624

Star, S., & Griesemer, J. (1989). Institutional Ecology, 'Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, S*ocial Studies of Science*, 19 (3), 387–420. doi: 10.1177/03063128901900300

Tallon, P. P., & Pinsonneault, A. (2011). Competing perspectives on the link between strategic information technology alignment and organizational agility: insights from a mediation model. *MIS Quarterly*, *35*(2), 463-486.

Tallon, P. P., Queiroz, M., Coltman, T., & Sharma, R. (2019). Information technology and the search for organizational agility: A systematic review with future research possibilities. *The Journal of Strategic Information Systems, 28*(2), 218-237. doi.org/10.1016/j.jsis.2018.12.002

Teece, D., Peteraf, M., & Leih, S. (2016). Dynamic capabilities and organizational agility: Risk, uncertainty, and strategy in the innovation economy. *California Management Review*, 58(4), 13-35. doi.org/10.1525/cmr.2016.58.4.13

Tortoriello, M., Reagans, R., & McEvily, B. (2011). Bridging the Knowledge Gap: The Influence of Strong Ties, Network Cohesion, and Network Range on the Transfer of Knowledge Between Organizational Units. *Organization Science, published online before print* doi:10.1287/orsc.1110.0688

Tushman, M. L. (1977). Special Boundary Roles in the Innovation Process. *Administrative Science Quarterly, 22*(4), 587-605.

Tushman, M. L. (1979). Work Characteristics and Subunit Communication Structure: A Contingency Analysis. *Administrative Science Quarterly, 24*(1), 82-98.

Vial, G. (2019). Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems, 28*(2), 118-144. doi.org/10.1016/j.jsis.2019.01.003

Walter, A.-T. (2021). Organizational agility: ill-defined and somewhat confusing? A systematic literature review and conceptualization. *Management Review Quarterly, 71*(2), 343-391. doi:10.1007/s11301-020-00186-6

Wenger, E. (1998). *Communities of Practice: Learning, meaning, and identity.* Cambridge, UK: Cambridge University Press. doi.org/10.1017/CBO9780511803932

Wenger, E. (2000). Communities of Practice and Social Learning Systems. *Organization, 7*(2), 225-246. Retrieved from http://org.sagepub.com/content/7/2/225.abstract

Wessel, L., Baiyere, A., Ologeanu-Taddei, R., Cha, J., & Jensen, T. B. (2021). Unpacking the Difference Between Digital Transformation and IT-Enabled Organizational Transformation. *Journal of the Association for Information Systems, 22*(1), 6. doi:10.17705/1jais.00655

Copyright

doi: https://doi.org/10.3127/ajis.v29.5095