Structural and Behavioral Fit in Software Sourcing Alignment

Marko Nöhren
University of Mannheim
Business School
noehren@uni-mannheim.de

Armin Heinzl
University of Mannheim
Business School
heinzl@uni-mannheim.de

Thomas Kude
University of Mannheim
Business School
kude@uni-mannheim.de

Abstract

In recent years, the corporate software ecosystem has been subject to tremendous changes. Vendors are increasingly offering enterprise systems in a Software-as-a-Service setting. For most of their applications, client companies can now choose between developing software in-house or sourcing packaged software in an on-premises or on-demand mode. However, no attempt has been made so far to explicitly compare these three sourcing arrangements with each other in terms of alignment and performance. The presented paper makes a contribution to fill this research gap. Drawing upon the ontological view of information systems, we develop a novel understanding of software sourcing arrangements. Based on exploratory case studies that are analyzed in the light of extant literature, we examine different types of fit arising from a structural and behavioral dimension of software sourcing. We further show how these types of fit are interrelated and discuss their performance implications.

1. Introduction

Looking at recent market data, decision makers across the globe trust that IT is a major strategic success factor for their company [1]. As a consequence, global IT expenditures are increasing year by year [2]. At the same time, decision makers are increasingly confronted with the challenge of improving efficiency and effectiveness of their IT portfolio by focusing on IT and business alignment, process re-engineering, and cost reduction [3, 4]. Despite a huge body of research on the value of IT, the debate on how IT contributes to performance of firms and thus how IT investments can be justified still persists [5]. One of the main drivers of the growth of global IT expenditures are software purchases [2]. Companies are increasingly moving from traditional in-house application development to sourcing of standard packaged software [6]. Against this background, the fit between enterprise software and organizations [7] and how it impacts usage and performance [8, 9] has been investigated. Furthermore, previous contributions revealed how to cope with misfits of standard applications [10, 11] as well as how such misfits emerge and develop over time and among organizational subunits [12, 13].

In recent years, the global software market has been subject to tremendous changes. Considering that companies are increasingly shifting from on-premises to on-demand software (software-as-a-service) [3, 14], a huge body of research addresses performance issues related to this novel software sourcing arrangements. For instance, studies in the area of on-demand adoption reveal how institutional influences increase perceived benefits or perceived disadvantages of cloud computing [15], what constitutes to quality of on-demand service delivery and how this impacts customer satisfaction and the intention to continue an ongoing relationship (e.g. [16, 17, 18]). Furthermore, the role of relational norms and knowledge interdependencies between providers and clients on contract design [19] and how contract mechanisms impact performance outcomes has been examined [20].

From a client perspective, the adoption of new software constitutes an innovation that needs to be carefully addressed [21, 22]. Therefore, the selection of an adequate software sourcing mode is a major challenge for client firms [23]. Drawing upon our review of literature we identified two significant shortcomings in previous research. First, studies on fit of enterprise systems enhanced our knowledge on the differences in utilizing standard software packages and in-house developed applications. However, research in this area does not explicitly address whether the software is sourced via an on-premises or an on-demand mode. Second, previous contribution in the area of cloud computing enriched our understanding of on-demand service-related issues and how to manage software-as-a-service relationships. However, the question of how on-demand solutions fit with organizational reality compared to on-premises or in-house software has not been addressed yet. Against this background, we aim at answering the following research question: *How do application software*

978-1-4799-2504-9/14 $31.00 © 2014 IEEE
DOI 10.1109/HICSS.2014.489

sourcing modes impact software alignment and performance?

The paper proceeds as follows. In the second chapter, we present the theoretical foundation of our work. Drawing on the ontological view of information systems, we discuss the duality of software sourcing modes and position our research in the area of IT alignment. Subsequently, we briefly describe our research design. In the fourth chapter, we present our conceptual framework derived from exploratory case studies and discussed in light of existing literature. We conclude with the theoretical and practical importance of our findings and by discussing implications for future research.

2. Theoretical Foundation

Before we investigate how software sourcing arrangements impact alignment and performance, we first need to define the software artifact and have a closer look at the distinct characteristics of in-house, on-premises, and on-demand applications. We start with a description of the ontological perspective of enterprise systems and continue with a discussion of the duality of software from a sourcing perspective.

2.1. Ontological View on Enterprise Software

Ontology is a philosophical domain dealing with models of reality in terms of assumptions about how the world is made up and what the nature of things is [13, 24]. It defines how to describe the structure of the world in general [25]. In information system research, the Bunge-Wand-Weber (BWW) ontology has been applied to define the IT artifact [13]. Instead of focusing on the way IT is managed, used, and implemented in organizations, and how these factors impact quality, performance, or adoption, BWW ontology views "*information systems as independent artifacts that bear certain relationships to the real-world system they are intended to model*" [26]. Information systems are representation of an organization and its reality [26]. It is distinguished between a deep and a surface structure of IT representing organizational reality [26]. Deep structure refers to the core of the real-world system that an IT artifact is designed to model [13]. An IT artifact is made up of things (e.g., an ordering processing software) with inherent properties attached (e.g., an inventory number) existing at certain states (e.g., waiting) that are changed through transformations (i.e., rules for product delivery) [13, 27]. The surface structure refers to the interface between users and the IT artifact [27]. It describes how real-world meanings are delivered by a system [13].

2.2. The Duality of Software Sourcing Modes

If software systems, such as enterprise software, are intended to be a representation of a real-world system in terms of an organization, its design must be capable to reflect the structure and behavior of the real world [26]. In this study, we look at the three sourcing options for enterprise applications, in-house, on-premises, and on-demand software. These souring arrangements differ with respect to their structural and behavioral dimension.

The first dimension refers to the degree of structural resource control. We distinguish between two control functions, deployment and development. In-house software is defined as a customized application developed internally by a corporate IT unit or externally by a contracted software vendor. The application is operated within a users' firm. Consequently, development and deployment control are within customers' hierarchy. On-premises software refers to off-the-shelf software, developed by an external vendor for multiple clients. The application is installed and operated within a users' firm. Therefore, only the deployment control function is held internally. Finally, on-demand software constitutes a sourcing arrangement, where an application is developed and operated outside a clients' firm [28, 29, 30]. Hence, both control functions are outside a customer's hierarchy.

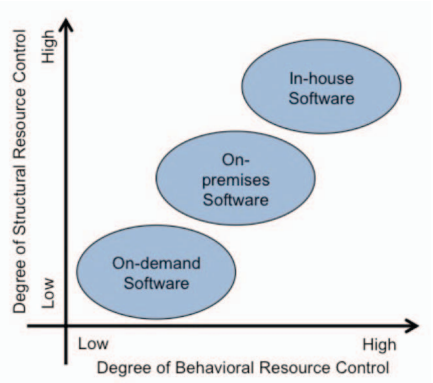

Figure 1. Duality of Software Sourcing

Greater structural resource control leads to a higher authority on deep and surface configuration. In an in-house setting, an organizational planner designs deep and surface structure as a representation of a company's reality. In an on-premises mode, the design of the system is made outside a firm. Typically, organizations can set up own interfaces (surface structure) and make certain changes to the software core (deep structure) due to an internal deployment of the system. The opportunity to adapt the surface

structure of an application in terms of building own interfaces that link the software and the users also exists in an on-demand setting. However, due to an external deployment, an organizational planner has no impact on the deep structure. Thus, a lower degree of structural resource control characterizes a higher standardization of software.

The second dimension of software sourcing refers to the degree of behavioral resource control. We differentiate between a transformation and timing function. In-house software constitutes a rather static sourcing arrangement. It is developed for the needs of a specific firm and is only transformed and updated if a certain improvement is desirable or necessary. Thus, transformation and timing function are largely within a firm's hierarchy, which constitutes a high behavioral resource control. In contrast, a vendor adapts on-premises software on the basis of the needs of multiple customers. A client decides when to update to a newer version. Hence, the behavioral control is lower than in an in-house setting. Finally, when sourcing software on-demand, both, the transformation and the timing function are outside a client's hierarchy. A vendor adapts the software on a regular basis and pushes modifications to all customers simultaneously [30]. Thus, the behavioral control from a client perspective is low.

The duality of software sourcing places particular demands on the alignment of enterprise application and organizational reality. A lower degree of structural control leads to a less customizable application. A lower degree of behavioral control leads to a lower discretion over how and when software representing an organization is changed.

2.3 Alignment and Enterprise Application Fit

Alignment is a "*(...) state in which the goals and activities of a business are in harmony with the information systems that support them*" [31]. For several years, alignment of IT and business has been among the top five management concerns [4]. Numerous studies have highlighted positive outcomes of alignment such as an increased competitive advantage (e.g. [32, 33]) as well as organizational and business process performance (e.g. [34, 35]). Furthermore, several antecedences of alignment (e.g. [33, 36]) and how alignment and related capabilities evolve over time [37] have been studied.

Prior research on IT alignment can be classified into five dimensions [38]. On an informal structure dimension, the differences of informal and formal alignment are compared [39]. On a social dimension, shared understanding and communication among people involved is analyzed [36]. The third dimension focuses on the cultural component of alignment by studying the cultural fit between business and IT [38, 40]. In previous research, particular attention has been paid to issues of structural and strategic alignment [38]. Strategic alignment describes the extent to which a certain business strategy is in line with a corporate IT strategy [33]. Structural alignment refers to structural fit between IT and business [41]. These two dimensions are closely linked to each other [42].

Drawing on the ontological view of information systems, we see software as a representation of an organizations reality [26]. Therefore, we focus on the structural dimension of alignment. Alignment research differs with respect to the level of analysis. Chan and Reich [38] distinguish between organizational level [33], systems, and project level [32, 35], as well as individual level [43] studies. In this study, we investigate software that is sourced in different ways and that provides a representation of organizational reality on a system's level. We focused on this level due to the fact that software sourcing decisions take place on application level. When sourcing software, the core and the surface structure needs to be in line with the business functionality it is supposed to support [13]. Hence, we study enterprise application fit and misfit.

Enterprise application misfits occur as "*(...) the result of differences between the structures embedded in the [software] and those embedded in the organization*" [10]. From an ontological perspective, misfits are cases where aspects of the real world are not adequately represented by the application [13]. In previous research, two major typologies of fit between software and organizational structure have emerged [12]. The first one is a typology by Sia and Soh [13]. In their view, misfits arise from differences in the actual organizational context and the context assumed by a developer. Drawing upon institutional theory, the authors develop four types of misfits with respect to these two dimensions. The first dimension is related to the system's ontology discussed in the second chapter. The second dimension focuses on the source of misfits and whether they arise from external (imposed structures necessary for the survival of a firm) or internal (voluntary structures developed by an organization) contingencies [13]. The second typology was proposed by Strong and Volkoff [7]. Based on grounded theory, the authors identified sources of misfits (functionality, data, usability, role, control, and organizational culture) embedded in the software application and differentiate them into two dimensions, deficiencies and impositions. The former refers to problems that arise when software features are needed but missing [7]. The latter one specifies those problems that arise from the inherent structure of the software

that requires ways of working conflicting with the organizational reality [7]. Furthermore, the authors develop two fit constructs. Coverage fit means that an application is free from deficiencies [7]. Enablement fit is related to the absence of impositions and describes a situation where the organization is better off with a new application in comparison to the legacy system [7]. In this study, we rely on the fit constructs of Strong and Volkoff [7]. These constructs expand the typology by Sia and Soh [13] with respect to latent structure deficiencies and impositions in terms of role, control, and organizational culture misfits and thus enable us to incorporate additional sources of misfit. Moreover, enablement fit gives us the opportunity to directly compare a current application with a replaced system.

3. Research Design

Due to the fact that little is known about impacts of in-house, on-premises, and on-demand software on IT alignment and sourcing performance, this study is exploratory in nature. It involves expert interviews with CIOs and senior IT management staff of large and medium-sized client firms. Our goal is to learn how companies align their enterprise applications with their businesses objectives and to identify how different application sourcing arrangements shape the way clients achieve operational and strategic benefits from their sourcing modes. The findings are linked to related literature in developing a conceptual framework.

In order to analyze the performance of software sourcing modes on business processes, we investigated customer relationship management (CRM) and enterprise resource planning (ERP) applications. We conducted six exploratory semi-structured expert interviews within six case companies (see Table 1) [44, 45]. To obtain a comprehensive picture of software sourcing mode, IT alignment, and performance implications, we interviewed senior IT managers at each case site. Prior to the implementation of the on-demand system at ALPHA, BETA, GAMMA, and DELTA and the on-premises software at EPSILON and ZETA, all of the six case companies utilized heterogeneous in-house applications, developed by external service providers. This allowed us to analyze in-house software sourcing in a retrospective manner.

We then transcribed and coded the interviews. This processes was initiated by relating text fragments to concepts identified in previous literature. That is, we initially coded for software sourcing outcomes, coverage fit, and enablement fit.

Generally, the term outcome is rather vague and a wide dispersion of dependent variables have been adopted [46]. For instance, studies in the area of IT value rely on measures such as benefits in terms of cost and time saving or increased flexibility (e.g. [47, 48]). In the area of IT sourcing, Lacity et al. [46] identified 36 different dependent variables such as success (e.g. [49, 50, 51]) or cost advantages (e.g. [52, 53]). Dibbern et al. [54] synthesized the myriad of outcome variables and classified them into the three categories satisfaction, realization of expectations, and performance.

Case Company and Application	Business	Interviewee
ALPHA (CRM)	Hygienic Products Manufacturer and Service Provider	CIO
BETA (CRM)	Flooring Material Manufacturer and Service Provider	IT Manager
GAMMA (CRM)	IT Infrastructure Service Provider	CIO
DELTA (CRM)	Consulting Company	CIO
EPSILON (ERP)	Civil and Industrial Construction and Services	CIO
ZETA (ERP)	Auditor Company	IT Manager

Table 1. Case Companies and Interviewees

An extensive discussion of the dependent variable is beyond the scope of this paper. Given that the primary aim of our study is to investigate fit and alignment of software sourcing modes in an exploratory manner, we rely on a broad notion of sourcing outcomes as classified by Dibbern et al. [54]. Looking at the realization of expectation supports us in assessing whether the current software sourcing mode outperforms a legacy system. We gather information on software performance in terms of perceived contribution to efficiency and effectiveness of the firm and investigate whether the case companies are satisfied with their sourcing results.

In addition, we related text fragments to a system's coverage fit, that is if the system "*includes the features that the organization needs to operate and that users need to do their work*", as well as to enablement fit, defined as the ability of the system "*to operate more effectively, and users to do their work more efficiently than was the case without [the system] even after accounting for negative effects of imposition*" [7]. In the process of theoretically linking sourcing mode, IT alignment, and performance, dynamic fit emerged as a new construct that reflects the behavioral dimension of software sourcing introduced above. We see software

systems to have a dynamic fit if the software is changed over time in accordance with the organization's requirements. The following section presents the results of our exploration.

4. Conceptual Framework

The results of our empirical analysis are illustrated in Figure 2 and presented subsequently. We identified three fit constructs that emerge from the duality of software sourcing. We found that the distinct characteristics of in-house, on-premises, and on-demand software enable two patterns of alignment that lead to varying degrees of sourcing outcome.

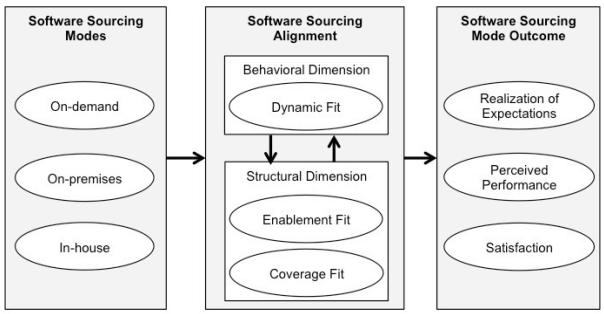

Figure 2. Conceptual Model

4.1. Structural Dimension

In line with previous research, the results of our study suggest that sourcing outcomes are particularly positive if the sourced software system does not result in deficiency misfits, and thus increase coverage fit [7]. For instance, the interviewee at ZETA stated that *"time is a crucial success factor for our firm. We need to make service offerings fast and don't waste time with long-lasting billing processes. (...) We save a lot of time in the whole process now."* Indeed, the empirical analysis suggests that this positive perception of ZETA's on-premises ERP system can be traced back to the absence of deficiency misfits *"The system works predictable and stable (...). The system shows only such data to the user that is relevant to that person."*

Similarly, the CIO of EPSILON commented that *"Information is structured, evaluable and accessible to everyone"* As a consequence, according to the CIO, the ERP system did not lack any needed feature, contributing to the positive software sourcing outcome (SSO). *"One advantage is speed. The software connects all subsidiaries with each other so that all can access the same data pool simultaneously."* Taken together, we posit

Proposition 1a: Coverage fit positively impacts SSO.

Similar to previous studies, our results showed varying degrees of imposition misfits that may lower the effectiveness and efficiency of the sourced software solution. As such, the CIO of GAMMA commented on the on-demand CRM solution that its company is applying: *"We see no opportunity to adapt the software to our processes."* As a direct result, the CIO stated that *"we are not satisfied with the CRM system. (...) Our hardware costs decreased, however other and unexpected costs increased significantly. (...) Overall, we don't experience any efficiency gains."* The situation was different at BETA, where *"One of our big goals with respect to the new CRM system was that work is globally executed in the same way. (...) If a sales person in Helsinki takes the order of a Finish architect, he/she needs to process this order in the same way as his counterpart in L.A."*. This company experienced positive outcomes. *"(...) we were able to increase the efficiency gains with the new system in terms of cost reduction and increasing connectedness of our global workforce."* Thus, enablement fit was found to improve sourcing outcomes.

Proposition 1b: Enablement fit positively impacts SSO.

Interestingly, our findings provide evidence that in case of enablement fit, a software system may result in positive outcomes even though deficiencies exist. This can be illustrated by the case of ALPHA, where no coverage fit could be observed: *"The software lacks a lot of features. E.g. our sales representatives need to participate in special training activities that we need to monitor with a CRM system"* However, the system enables ALPHA to perform sales-related processes more effectively and efficiently than was possible before. *"(It) is a product that can be adapted to our processes not vice versa. (...) The software enables us to introduce a customer self service system (...)"* According to ALPHA's CIO, this benefit outweighed the issues related to deficiency misfits and resulted in a positive evaluation of the software *"It helped us to differentiate from competitors and increased our timeliness significantly."* This leads us to suggest the following proposition:

Proposition 1c: Enablement fit potentially outweighs the negative impact of deficiency misfits on SSO.

4.2. Behavioral Dimension

In addition to coverage and enablement fit, the behavioral dimension of software sourcing identified above points out the role of software changes over time when studying enterprise application fit. In general, change encompasses the three components magnitude, direction, and timing [55]. Magnitude describes the extent to which software is changed, and whether it is a small or a large adoption. Direction expresses whether an application is adapted towards an organization's reality. Timing refers to the date software is changed.

If software is transformed according to a firms requirements (direction), to the right extend (magnitude), and at the right time (timing), a dynamic fit is established. To illustrate this, we adapt the classes of dynamic fit by Zodiac et al. [55]. As illustrated in Table 2, there are two ways to realize dynamic fit.

		Software change occurs: yes	Software change occurs: no
Organizational reality requires software change	yes	**Quadrant II** Beneficial Agitation	**Quadrant I** Harmful Inertia
	no	**Quadrant III** Harmful Agitation	**Quadrant IV** Beneficial Inertia

Table 2. Dynamic Fit

First, if an organization's reality has changed and if this change leads to a situation in which the current application or parts of it becomes inappropriate to deal with this new situation, a company is likely to experience a downturn in structural fit (coverage and enablement fit). In this situation, the firm requires a software change as a necessary condition for dynamic fit. If this change happens according to the requirements of a firm with respect to magnitude, direction, and timing (sufficient condition), dynamic fit referred to as beneficial agitation is established (quadrant II). In our study, companies experiencing deficiencies and enablement fit at the same time were very satisfied with the behavioral dimension of their CRM systems. For instance, the CIO at DELTA stated that *"we want to have a software that is constantly developed. (...) We don't need to take care of software updates any more"* Similarly, the CIO at ALPHA commented that *"six times a year, we get an update of the software with new functionalities without doing anything for it. That means, we always have the latest technology, which is very important for us."*. Those firms realized dynamic fit through beneficial agitation. This agitation constantly adds improvements to the system that were advantageous for the company and contribute to establishing enablement fit over time.

Proposition 2a: Dynamic fit in terms of beneficial agitation positively impacts SSO.

Second, if a company demands for persistence and the current enterprise software is in line with the organizational reality (necessary condition), the sufficient condition for dynamic fit is a stable software environment. We label this type of fit beneficial inertia (quadrant IV). In our study, dynamic fit was also present at EPSILON and ZETA. For instance, the CIO of EPSILON commented on the inertia of the system: *"There has been a presentation by the software vendor how to use the program and adapt it to our needs. (...) The implementation and adaptation was a very time consuming step. Fortunately the software is running now and we don't have to adapt it any time soon"* Both companies strived for a stable software environment and achieved dynamic fit by beneficial inertia.

Proposition 2b: Dynamic fit in terms of beneficial inertia positively impacts SSO.

In addition, there are two ways to suffer from dynamic misfit. The first one is referred to as harmful agitation. It describes a situation where a company demands for persistence and software change occurs (quadrant III). In our study, GAMMA suffered from dynamic misfit. At the time of the data collection, the implementation has been three years ago. However, the CIO stated that the implementation has not been completed yet. *"The implementation was a letdown. It took way longer than expected and it is not completed, yet. (...) So far, software updates were not helpful for us. Quite the opposite was the case."* This was primarily due to the fact that the updates were not valuable to the firm. In fact, quite the contrary was the case. The updates extended the implementation time and thereby add imposition misfits for the company.

Proposition 2c: Dynamic misfit in terms of harmful agitation negatively impacts SSO.

The second way to establish dynamic misfit is if an organization's reality has changed and the software remains static. This situation is labeled harmful inertia (quadrant I). No case company in our study suffered from this type of dynamic misfit with its current software application. However, the CIO at ALPHA commented on the inertia of the replaced system: *"Our legacy system has been out there for decades. (...) The system has not been improved in the last years."* This situation resulted in negative outcomes for the firm *"It was no tool to react to changes in the market."* We propose:

Proposition 2d: Dynamic misfit in terms of harmful inertia negatively impacts SSO.

4.3. Role of Software Sourcing Mode

Despite the fact that we discovered a positive connection between the three fit constructs and sourcing outcomes, the question of how the structural and behavioral dimension are linked to each other as well as the role of software sourcing modes remains. Drawing upon our findings discussed above, we derived two favorable patterns. The first one is labeled

evolution. We found that three out of six firms (ALPHA, BETA, and DELTA) realized positive sourcing outcomes even though deficiencies exist. These companies requested their software to be highly volatile, which was necessary to establish enablement fit over time. Thus, a certain degree of deficiency misfits was accepted in order to establish dynamic fit in terms of beneficial agitation. The second pattern is labeled ***maturation***. In our study, the companies EPSILON and ZETA focused on realizing coverage and enablement fit. A constantly changing software environment was not desired. Therefore, these companies aimed at realizing dynamic fit in terms of beneficial inertia.

In an in-house setting, structural and behavioral resource control is held internally. This type of software is developed with respect to a particular firm's requirements. Therefore, structural misfits in terms of deficiencies and impositions are unlikely to occur as long as the organizational reality has not changed. The interviewee at BETA commented on the replaced in-house software that *"Our legacy CRM system was developed by an external service provider. It was customized to our needs. It was old-fashioned, but it did what it was supposed to do."* Moreover, in-house software constitutes a very stable sourcing arrangement. *"At the time we shifted to the new system, (vendor that developed and customized the legacy in-house software) had been out of market for several years" (BETA)*. This sourcing mode appears to be adequate if a firm demands for persistence in their software portfolio.

Proposition 3a: Sourcing software in-house positively impacts maturation.

However, if dynamic fit can only be established by change, the stability of in-house software increases the risk of harmful inertia. This risk is especially high when change in organization's reality occurs frequently. Therefore, we propose:

Proposition 3b: Sourcing software in-house negatively impacts evolution.

When implementing a standard software package in terms of an on-demand or on-premises application, rival forces between software and organization leads to structural misfits [12]. Companies strive for solving structural misfits through organizational adaption or package customization [13]. We found that all four companies ALPHA, BETA, GAMMA, and DELTA sourcing software on-demand were unable to realize coverage fit. As such, we found that constant software changes multiple times a year results in a highly disruptive nature for customers. The interviewee at BETA commented that *"The integration is an ongoing story. It is never completed."* While ongoing coverage fit is difficult to achieve in an on-demand setting, our findings suggest that this sourcing mode may be particularly appropriate when companies strive for constant software evolution. In our study, the three companies ALPHA, BETA, and DELTA experiencing positive outcomes demanded for agitation. Therefore, they were willing to accept certain deficiencies. In contrast, the disruptive nature of the on-demand application was not beneficial for GAMMA. The firm suffered from harmful agitation.

Proposition 3c: Sourcing software on-demand positively impacts evolution.

Even though, software maturation might be realizable with on-demand software, the fact that the behavioral control is outside a firm's hierarchy would result in high efforts in terms of continuous surface adaptation. Therefore, a focus on maturation is not appropriate with on-demand software.

Proposition 3d: Sourcing software on-demand negatively impacts maturation.

Two companies (EPSILON and ZETA) engaged in on-premises sourcing. We found that both companies were satisfied with the outcomes of their system. With respect to the structural dimension they focused on realizing coverage and enablement fit. For instance the CIO of EPSILON stated: *"A good software needs to represent our processes and not change them. It needs to adapt to our processes and not to require us to change."* The realization of coverage fit is possible due to fact that the timing function of the behavioral dimension is held within a firm's hierarchy. Even though there might be deficiencies, when software is implemented, the internal deployment provides a company with the opportunity to realize coverage fit over time. The CIO of EPSILON commented that *"we have no `take it or leave it' mentality. Things don't work that way in our firm. We listened to each individual's complains and tried to adapt the interface as much as possible."* Therefore, we propose:

Proposition 3e: Sourcing software on-premises positively impacts maturation.

As opposed to in-house settings, on-premises software is constantly developed by vendors based on industry best practices reflecting the external transformation control function of the behavioral dimension. This makes on-premises software as vital as on-demand applications. Even though EPSILON and ZETA focused on realizing dynamic fit in terms of inertia, it is also possible to update on-premises software frequently.

Proposition 3f: Sourcing software on-premises positively impacts evolution.

To sum up, the behavioral dimension of software sourcing plays a key role regarding application fit. It

drives structural fit in terms of enablement and coverage over time. On one hand, the internally held timing function of in-house and on-premises software enables a company to focus on maturation. On the other hand, the external transformation of standard software such as on-demand and on-premises applications enables a company to focus on evolution. However, the more frequently software needs to be adapted to establish enablement fit, the less appropriate are on-premises applications. For instance, the CIO of ALPHA stated: *"We get new releases with new features six time a year. If I imagine, we would have to send installer packages to more than 2000 users, I have no glue how this should work effectively."*

Based on these findings, we assert that the software sourcing modes are directly linked to the two patterns. This choice needs to be made with a close consideration of the organizational reality. Drawing on the ontological view, this reality is created based on internal and external contingencies. Accordingly, we reject the assumption of "one size fits all" software sourcing mode strategy [42]. Instead, we suppose that contextual and organizational factors needs to fit together to facilitate alignment [38]. We found that the three companies engaged in an evolution strategy are faced with high and turbulent competitive pressures in terms of industry rivalry (BETA and DELTA) and threat of substitute goods (ALPHA). Those companies have subsidiaries at globally dispersed locations that closely cooperate with each other. Their processes are rather loosely structured. By contrast, the two companies following the maturity strategy (EPSILON and ZETA) are facing stable competitive pressures. Their processes are well structured with little room for alteration.

5. Discussion and Conclusion

In this paper, we studied in-house, on-premises, and on-demand software from an ontological perspective. More specifically, we aimed at answering the research question of how different sourcing modes impact alignment and performance. Our findings suggest that the sourcing mode does not only influence coverage and enablement fit, but also dynamic fit in terms of when software changes are made. Moreover, we found that companies can be successful through evolution (facilitated by on-demand and also on-premises software) as well as maturation (enabled through in-house and on-premises software).

Our results provide a number of contributions to prior literature. First, our study is a first step to complement existing work on enterprise application fit. Previous work has mostly focused on a structural dimension and distinguished between coverage and enablement fit. By adding the behavioral dimension that incorporates software changes over time, our conceptual framework helps to analyze different sourcing modes in terms of their implications on fit on a system level. As such, our findings suggest that by ensuring enablement fit over time, on-demand software may be beneficial even if certain deficiency misfits arise.

Second, this study contributes to previous work on on-demand software sourcing. While prior literature provided important insights on the benefits and risks associated with sourcing software-as-a-service, the question of how on-demand sourcing as opposed to on-premises or in-house software influence enterprise-application fit was still missing. The findings of our study show that when adopting on-demand software, deficiency misfits are usually more likely due to the standardized nature of these applications. At the same time, the continuous update of these solutions in terms of technological advancements and industry best practices may also entail benefits for organizations in that they may facilitate dynamic fit. Our findings suggest that in some cases, the merits of dynamic fit may outweigh the lack of perfect coverage fit.

Our study has several managerial implications. First, practitioners learn about the inherent and distinct characteristics of software sourcing. These individual characteristics pose different challenges to alignment of application and business. Second, due to the relationship between coverage and dynamic fit, we revealed a trade-off between a high structural fit and beneficial agitation. Practitioners need to be aware of the different benefits before making a sourcing decision. Third, we identified two promising alignment patterns realizable with in-house, on-premises, and on-demand software. Our findings suggest that the feasibility of these patterns depends on the external and internal environment.

Our study has three important limitations that need to be taken into account when interpreting our findings. First, even though our data hints to certain context factors our small data set did not allow an in-depth investigation of external and internal contingencies. Future research may therefore rely on a larger data set that systematically accounts for context factors influencing the feasibility of the identified maturation and evolution patterns. Second, whereas previous studied accounted for the multi-level nature of the enterprise application fit construct, our findings rely on statements from CIOs and senior IT management staff. Future research may therefore study coverage, enablement, and dynamic fit both, by interviewing high-level IT executives, and by observing the occurrence of misfits on a process or individual level. Such a multi-level analysis of sourcing modes and

enterprise application fit may also account for the inseparability of human agency and technology, which may influence misfit perceptions on a process or individual level [56]. Third, we were only able to investigate in-house software sourcing in retrospective. Future research may rely on data that includes firms, which currently rely on in-house software solutions.

6. References

[1] D. Q. Chen, "Information Systems Strategy: Reconceptualization, Measurement, and Implications", MIS Quartely, vol. 34, no. 2, pp. 233-259, 2010.

[2] A. H. Bartels, E. Daley, C. Mines, and C. Mute, "US and Global IT Market Outlook: Q4 2009", Forrester Research, Inc., Cambridge, 2010.

[3] J. Bélissent, H. Shey, E. Daley, and Z. Reiss-Davis, "The State of Global Enterprise IT Budgets: 2009 to 2010," Forrester Research, Inc., Cambridge, 2009.

[4] J. Luftman and T. Ben-Zvi, "Key Issues for IT Executives 2009: Difficult Economy's Impact on IT", MIS Quarterly Executive, vol. 9, no. 1, pp. 49-59, 2010.

[5] N. Melville, K. Kraemer, and V. Gurbaxani, V. "Information Technology and Organizational Performance: An Integrative Model of It Business Value", MIS Quarterly, vol. 28, no. 2, pp 283-322, 2004.

[6] S. Sawyer, "Packaged Software: Implications of the Differences from Custom Approaches to Software Development", European Journal of Information Systems, vol. 9, no. 1, pp. 47-58, 2000.

[7] D. M. Strong, and O. Volkoff, "Understanding Organization–Enterprise System Fit: A Path to Theorizing the Information Technology Artifact", MIS Quarterly, vol. 34, no. 4, pp. 731-756, 2010.

[8] D. L. Goodhue, and R. L. Thompson, "Task-Technology Fit and Individual Performance", MIS Quarterly, vol. 19, no. 2, pp. 213-236, 1995.

[9] H. van der Heijden, "Measuring IT Core Capabilities for Electronic Commerce: Results from a Confirmatory Factor Analysis," Proceedings of the International Conference on Information Systems, 2000.

[10] C. Soh and S. K. Sia, "An Institutional Perspective on sources of ERP Package–Organisation Misalignments", Journal of Strategic Information Systems, vol. 13, pp. 375–397, 2004.

[11] J. Ward, C. Hemingway, and E. Daniel, "A Framework for addressing the Organisational Issues of Enterprise Systems Implementation", Journal of Strategic Information Systems, vol. 14, pp. 97–119, 2005.

[12] C. Maurer, N. Berente, and D. Goddhue, "Are Enterprise System Related Misfits Always a Bad Thing?," Proceedings of the 45th Hawaii International Conference on System Sciences (HICSS), 2012.

[13] S. K. Sia, and C. Soh, "An Assessment of Package–Organisation Misalignment: Institutional and Ontological Structures", European Journal of Information Systems, vol. 16, pp. 568–583, 2007.

[14] S. Merz, C. Eschinger, T. Eid, C. Pang, and L. Wurster,. "Forecast: Software as a Service, Worldwide, 2010-2015, 1h11 Update", Gartner Research, 2011.

[15] S. Saya, L. G. Pee, and A. Kankanhalli, "The Impact of Institutional Influences on Perceived Technological Characteristics and Real Options in Cloud Computing Adoption", Proceedings of the International Conference on Information Systems, 2010.

[16] W. Hong, Y. Y. M. Thong, L. C. Chasalow, and G. Dhillon, "User Acceptance of Agile Information Systems: A Model and Empirical Test", Journal of Management Information Systems, vol. 28, no. 1, pp. 235–272, 2011.

[17] S. Lee, B. Shin, and H. G. Lee, "Understanding Post-adoption Usage of Mobile Data Services: The Role of Supplier-side Variables", Journal of the Association of Information Systems, vol. 10, no. 12, pp. 860-888, 2009.

[18] A. Susarla, A. Barua, and A. B. Whinston, "Understanding the Service Component of Application Service Provision: An Empirical Analysis of Satisfaction with ASP Services," MIS Quarterly, vol. 27, no. 1, pp. 91-123, 2003.

[19] A. Susarla, A. Barua, and A. B. Whinston, "Multitask Agency, Modular Architecture, and Task Disaggregation in SaaS", Journal of Management Information Systems, vol. 26, no. 4, pp. 87–117, 2010.

[20] A. Susarla, A. Barua, and A. B. Whinston, "A Transaction Cost Perspective of the "Software as a Service" Business Model," Journal of Management Information Systems, vol. 26, no. 2, pp. 205–240, 2009.

[21] E. B. Swanson, "Information Systems Innovation Among Organizations", Management Science, vol. 40, no. 9, pp. 1069-1092, 1994.

[22] E. B. Swanson, and N. C. Ramiller, "Innovating Mindfully with Information Technology", MIS Quarterly, vol. 28, no. 4, pp. 553-583, 2004.

[23] A. Schwarz, R. Hirschheim, B. Jayatilaka, and T. Goles, "A Conjoint Approach to Understanding IT Application Services Outsourcing", Journal of the Association of Information Systems, vol. 10, no. 10, pp. 748-780, 2009.

[24] E. G. Guba, and Y. S. Lincoln, "Competing Paradigms in Qualitative Research", in: Denzin N.K. and Lincoln Y.S. (eds.): Handbook of Qualitative Research, Thousand Oaks, California (USA), pp. 105-117, 1994.

[25] Y. Wand, and R. Weber, "Research Commentary: Information Systems and Conceptual Modeling - A Research Agenda", Information Systems Research, vol. 13, no. 4, pp. 363–376, 2002.

[26] Y. Wand, and R. Weber, "Towards a Theory of the Deep Structure of Information Systems", Proceedings of the 11th International Conference on Information Systems (ICIS), 1990.

[27] S. K. Sia, and C. Soh, "Severity Assessment of ERP-Organization Misalignment: Honing in on Ontological Structure and Context Specificity", Proceedings of the 23th International Conference on Information Systems, 2002.

[28] A. Benlian, and T. Hess, "Opportunities and Risks of Software-as-a-Service: Findings from a Survey of IT Executives", Decision Support Systems, vol. 52, no. 1, pp. 232-246, 2011.

[29] S. Stuckenberg, E. Fielt, and T. Loser, "The Impact Of Software-As-A-Service On Business Models Of Leading Software Vendors: Experiences From Three Exploratory Case Studies", Proceedings of the Pacific Asia Conference on Information Systems (PACIS), 2011.

[30] M. Xin, and N. Levina, “Software-as-a Service Model: Elaborating Client- Side Adoption Factors”, Proceedings of the 29th International Conference on Information Systems (ICIS), 2008.
[31] J. D. McKeen, and H. Smith, “Making IT Happen: Critical Issues in IT Management”, Chichester, Hoboken, NJ: Wiley, 2006.
[32] S. W. Floyd, and B. Woolridge, “Path Analysis of the Relationship between Competitive Strategy, Information Technology, and Financial Performance”, Journal of Management Information Systems, vol. 7, no. 1, pp. 47–64, 1990.
[33] G. S. Kearns, and A. L. Lederer, “The Effect of Strategic Alignment on the use of IS-Based Resources for Competitive Advantage”, Journal of Strategic Information Systems, vol. 9, no. 4, pp. 265–293, 2000.
[34] F. Bergeron, L. Raymond and S. Rivard, “Ideal Patterns of Strategic Alignment and Business Performance”, Information & Management, vol. 41, no. 8, pp. 1003–1020, 2004.
[35] P. P. Tallon, “A Process-Oriented Perspective on the Alignment of Information Technology and Business Strategy”, Journal of Management Information Systems, vol. 24, no. 3, pp. 227–268, 2007.
[36] D. S. Preston, and E. Karhanna, “Antecedents of IS Strategic Alignment: A Nomological Network”, Information Systems Research, vol. 20, no. 2, pp. 159–179, 2009.
[37] R. Sabherwal, R. Hirschheim, and T. Goles, “The Dynamics of Alignment: Insights from Punctuated Equilibrium Model”, Organization Science, vol. 12, no. 2, pp. 179-197, 2001.
[38] Y. E. Chan, and B. H. Reich, “IT Alignment: What have we learned?”, Journal of Information Technology, vol. 22, no. 4, pp. 297–315, 2007.
[39] Y. E. Chan, “Information Systems Strategy, Structure and Alignment”, in R. Papp (ed.): Strategic Information Technology: Opportunities for competitive advantage, 1st edn, Hershey, PA: Idea Group Publishing, pp. 56–81, 2001.
[40] P. J. Pyburn, “Linking the MIS Plan with Corporate Strategy: An exploratory study”, MIS Quarterly, vol. 7, no. 2, pp. 1–14, 1983.
[41] Y. E. Chan, “Why Haven’t we Mastered Alignment?: The Importance of the Informal Organization Structure”, MIS Quarterly Executive, vol. 1, no. 2, pp. 97–112, 2002.
[42] J. C. Henderson, and N. Venkatraman, “Strategic Alignment: Leveraging information technology for transforming organizations”, IBM Systems Journal, vol. 32, no. 1, pp. 4–16, 1993.
[43] F. B. Tan, and B. Gallupe, “Aligning Business and Information Systems Thinking: A cognitive approach”, Engineering Management, IEEE Transactions, vol. 53, no. 2, pp. 223–237, 2006.
[44] K. M. Eisenhardt, “Building Theories from Case Study Research”, The Academy of Management Review, vol. 14, no. 4, pp. 532-550, 1989.
[45] R. K. Yin, “Case Study Research - Design and Methods”, 4th ed.: Sage Publications, 2009.
[46] M. C. Lacity, S. Khan, A. Yan, and L. P. Willcocks, “A review of the IT outsourcing empirical literature and future research directions”, Journal of Information Technology, vol. 25, pp. 395–433, 2010.
[47] W. H. DeLone, and E. R. McLean, “The DeLone and McLean Model of Information Systems Success: A Ten-Year Update”, Journal of Management Information Systems, vol. 19, no. 4, pp. 9-30, 2003.
[48] C. Soh, and M. L. Markus, "How IT Creates Business Value: A Process Theory Synthesis", Proceedings of the Sixteenth International Conference on Information Systems, 1995.
[49] J. N. Lee, S. M. Miranda, and Y. M. Kim, "IT Outsourcing Strategies: Universalistic, Contingency, and Configurational Explanations of Success", Information Systems Research, vol. 15, no. 2, pp. 110–131, 2004.
[50] V. Grover, M. J. Cheon, and J. T. C. Teng, "The Effect of Service Quality and Partnership on the Outsourcing of Information Systems Functions", Journal of Management Information Systems, vol. 12, no. 4, pp. 89-116, 1996.
[51] J. Goo, C. D. Huang, and P. Hart, “A Path to Successful IT Outsourcing: Interaction Between Service-Level Agreements and Commitment. Decision Sciences”, vol. 39, no. 3, pp. 469-506, 2008.
[52] J. Dibbern, J. Winkler, and A. Heinzl, “Explaining Variations in Client Extra Costs Between Software Projects Offshored to India”, MIS Quarterly, vol. 32, no. 2, pp. 333-366, 2008.
[53] M. C. Lacity, and L. P. Willcocks, “An Empirical Investigation of Information Technology Sourcing Practices: Lessons from Experience”, MIS Quarterly, vol. 22, no. 3, pp. 363-408, 1998.
[54] J. Dibbern, T. Goles, R. Hirschheim, and B. Jayatilaka, “Information Systems Outsourcing: A Survey and Analysis of the Literature”, The DATA BASE for Advances in Information Systems, vol. 35, no. 4, pp. 6-102, 2004.
[55] E. J. Zajac, M. S. Kraatz, and R. K. F. Bresser, “Modeling the Dynamics of Strategic Fit: A Normative Approach to Strategic Change”, Strategic Management Journal, vol. 21, no. 4, pp. 429-453, 2000.
[56] W. J. Orlikowski, “Sociomaterial Practices: Exploring Technology at Work”, Organization Studies, vol. 28, pp. 1435-1448, 2007.