Version Transitioning of Enterprise Systems in Software Ecosystems: A Grounded Theory Approach

Philip Holst Riis

Ph.d. stipendiat Phillip Holst Riis works at the Center for Applied ICT, Copenhagen Business School, Denmark. His research interests are the development and the quality of Enterprise systems.

This paper presents an emerging theory of version transitioning from an old to a new version of a pre-packaged enterprise system among consultant companies in a software ecosystem. The emerging theory proposes the key categories of Perceiving, Pushing, Implementing, and Increased experience as stages in the transition process, and the categories of Technology impact, Supplier impact, Customer impact, Strategy impact, and Market impact as key contextual categories impacting the transition process. The emerging theory proposes an iterative nature of the transition process in which each stage in the process is undergone multiple times by the consultant companies. The integration of the emerging theory with existing adoption and diffusion theories provides an initial step towards a formal theory of version transitioning in software ecosystems.

1. Introduction

While early implementations of enterprise systems in the ‘80s and ‘90s relied on development by a software vendor to fit the individual company, pre-packaged enterprise systems have now become dominant within the past decade [1]. In tandem, the delivery model of enterprise systems is increasingly evolving from two-party (vendor-customer) configurations to loosely coupled networks [2], also referred to as software ecosystems [3].

These ecosystems typically consist of a vendor, also referred to as a keystone [4] or a hub [5], which develops the core of the enterprise system, and a number of partners, also referred to as niche players [4], or spokes [5], who deliver a range of products and services complementing the core system delivered by the vendor [6]. Among the services delivered by the players in an ecosystem is consultancy on the implementation of the enterprise system at the customer organisation which includes solving problems, offering related and required knowledge, assisting with configuration, and deriving value from the enterprise system package [7]. The implementation consultants performing these services are thus an important part of the ecosystem, and previous research suggests that having competent implementation consultants is among the critical success factors for successful implementation of enterprise systems [8, 9].

Furthermore, the inter-linked nature of ecosystems suggests that the success of adoption of innovations in the ecosystems is dependent on adoption of all actors in the ecosystem rather than adoption at any single actor [4]. Previous research has addressed multiple perspectives of enterprise software ecosystems, including the motivation for forming the partnerships [2], coupling and control [5, 10], value creation [6], and competitive advantage [11; Anonymous, 2011].

However, not much research has addressed the process of adoption of new versions of enterprise systems packages released by the vendor into the ecosystems, which precedes the implementation of enterprise systems in customer organisations. Therefore, this paper investigates the transition to a new version of a pre-packaged enterprise system in an ecosystem of a large software vendor for the purpose of uncovering the paths in the transition process from the perspective of the implementation consultants.

The paper is structured as follows: 1) background presentation of the research setting; 2) methodology of the research; 3) the emerging theory; 4) the emerging theory in the context of the research; 5) discussion of the findings and theoretical integration; 6) conclusions; and 7) implications for practice and future research.

2. Background of the research setting

The enterprise system vendor in the study is a major global player in the market for enterprise systems. The vendor followed the consolidation of the enterprise systems market in the early 2000’s [12] and acquired a number of enterprise system solutions resulting in a portfolio of systems primarily targeted at small and medium enterprises (SMEs). The vendor releases a new major version of its enterprise systems approx. every 2-3 years, and so-called service packs with bug fixes and other improvements are sometimes released in-between the major releases. The particular enterprise system in vendor's portfolio included in this study has gone through six major releases.

The vendor sells and implements the enterprise system only through an ecosystem of partner companies, and the partner companies thus handle all implementations in customer organisations. The partner companies can broadly be categorised into two different types: Independent Software Vendors (ISVs) and Value Added Resellers (VARs).

The ISVs develop reusable software modules for the enterprise system, called 'add-ons'. There are several hundred add-ons available that complement the core enterprise system in areas ranging from generic horizontal functions such as payroll, online banking, and shipping to specialized vertical solutions such as education, veterinary medicine, legal companies, and furniture manufacturing.

Any individual or community with a developer license can extend the enterprise system and develop add-ons, but only add-ons that are developed by certified partners and have undergone quality assurance are listed as official add-ons on the vendor's website. The vast majority of add-ons are thus developed by certified ISV partners. Nearly all implementations in customer organisations include one or several add-ons to complement the core enterprise system package. The business model of the ISVs is thus to sell licenses for the add-ons to customers through the VARs, who in turn get a share of the license fee.

The consultants at the VAR companies take on the implementation of the pre-packaged enterprise system at the customers. The consultants make customisations to the enterprise systems by request from the customers but, unlike the ISVs, the customisations are customer specific and seldom reused across different customers. The VARs generate the majority of their revenue from invoicing the time spent on implementation and customisation, and only a smaller part of their revenue is generated from getting a part of the license fee. On a typical implementation of the enterprise system only 1-2 consultants are involved, depending on the amount of customisation needed. Some of the partner companies have characteristics of both an ISV and a VAR, meaning that they develop reusable add-ons which they sell to VARs, and they have a staff of consultants implementing the enterprise system together with the add-ons from themselves. Figure 1 illustrates the different value chain paths of the players in the ecosystem.

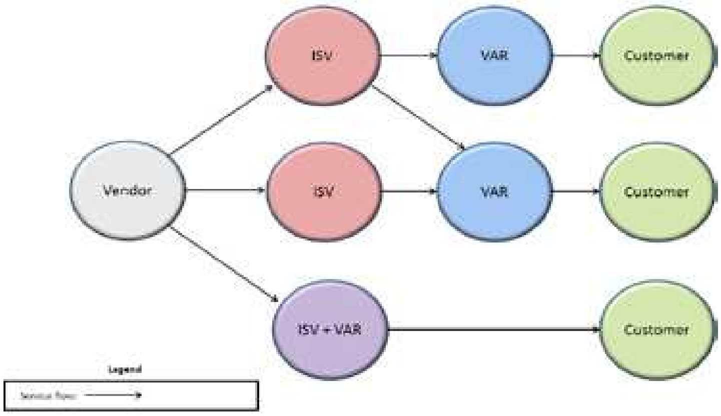

Figure 1 - Value chain of the software ecosystem

3. Methodology

The study was carried out using a Grounded Theory approach [13] as the frame for data collection and analysis. Grounded Theory is a 'data centric' inductive methodology for analysing (primarily qualitative) data for the purpose of building or extending theory [14], and the method has been evolved and applied to multiple research studies in the field of information systems [15].

The method stands out from many other research methods by emphasising that researchers rid themselves of theoretical pre-conceptions about the area of inquiry and that theory should emerge from the data – not through deduction or hypothesis testing [16]. The substance of this tenet has fuelled debate, not only among researchers using the method, but also between the two founders of the method, concerning the risk of forcing theory from the data instead of allowing the theory to emerge [17]. The details of this debate is beyond the scope of this paper, but the implications forces a stance on the use of existing theoretical literature in the study. The approach to existing literature in this study was a 'middle of the road' approach, where a general orientation within the literature of adoption of technology and diffusion of innovations was present prior to the analysis of the data, but no pre-existing theoretical constructs were forced on the data. A detailed comparison with existing literature was not conducted until after the emerging theory was present.

Urquhart et al. [18] provides five guidelines for conducting Grounded Theory in the IS field: Constant comparison; Iterative conceptualisation; Theoretical sampling; Scaling up; and Theoretical integration. Besides providing a guide and support for IS researchers embarking on conducting Grounded Theory, the five guidelines also explicate the essence of the method.

Constant comparison is the process of constantly comparing instances of data to a particular concept or category for the purpose of exposing theoretical properties of the concepts and categories. This guideline was followed by constantly comparing all the coded instances of data to other coded instances of data.

Iterative conceptualisation suggests that researchers should increase the level of abstraction and relate categories to each other to expose the different relationships between theoretical constructs. This should be done through the process of theoretical coding [19], or axial coding [14]. This guideline was followed by going through several iterations of the coding process, resulting in the same instance of data being re-coded several times in the iterative process of splitting and merging codes. Furthermore, theoretical memos were written as the analysis progressed and the memos were used for generating theoretical codes used for coding the data and for relating the codes to each other.

Theoretical sampling stresses the importance of deciding on analytical grounds where to sample from as the research progresses [20]. This approach helps saturate the categories of the emerging theory and ensures that the theory is actually grounded in the data [21]. This guideline had a significant impact on the research, as agreements with interviewees and consulting companies could not be made prior to initiating the research study, but had to be established as the data analysis played out. Furthermore, the data for the study was collected from respondents in companies of various roles in the ecosystem, different sizes, and with various degrees of experience with the new version of the enterprise system.

The guideline of Scaling up proposes the grouping of higher level concepts into broader themes to help escape the descriptive level of analysis and help contributing to the generalizability of the emerging theory. This process was aided by extensive use of the theoretical memos and by iteratively visualising the emerging theory through the use of diagrams in order to reach a substantive theory rather than mere description.

Theoretical integration calls for integration of the developed substantive theory with other theories in the same or similar fields in order to create a formal theory [22] that extends beyond the substantive area in which the theory originally emerged. In this study the substantive theory was related to other theories within and outside the IS field by reviewing literature on theory addressing adoption of technology and diffusion of innovations.

3.1. Data collection

Three types of data were collected and analysed as part of the research: Documents; observations; and interviews. Documents, primarily from the vendor, were used in the beginning of the study for gaining background information about the new version and to gain insight into the documented differences between the old and the new version.

Two types of observations were made during the study. The first type consisted of participatory observations [23] where the observing researcher participated in three presentations and four workshops with consultants concerning the new version. The second type of observations came from in-depth experimenting with a demo version of the new version of the enterprise system, provided by the vendor.

All interviews conducted in the research were semi-structured [24] with the initial interview guides being explorative and open-ended, but as the research progressed, the interview guides became more focused on saturating the emerging categories, and thus varied significantly from the initial interview guides. 12 interviews with consultants and managers in the partner companies in the ecosystem were carried out as part of the research. Additionally, two interviews with representatives from the vendor were conducted for three reasons: First, to provide the background information on the ecosystem; second, to saturate concepts and categories based on the principle of theoretical sampling; and finally, to triangulate statements from the interviews with the consultants. A total of 14 face-to-face interviews were carried out between December 2008 and March 2011. Each interview lasted approx. one hour on average, and all interviews were recorded and fully transcribed to allow detailed coding of the data. An overview of the conducted interviews is shown in Table 1. Due to reasons of non-disclosure agreements, the country in which the study was conducted is not revealed, and the names of the vendor, partner companies, and respondents are replaced by aliases.

Company alias	No. of employees	Company type	Interviewee title
Partner 1	28	ISV + VAR	CIO
Partner 2	1100 global/250 local	VAR	Unit Manager
Partner 3	50	VAR	Consultant
Partner 4	14	VAR	Chief Consultant
Partner 5	1	VAR	Consultant
Partner 6	39000 global/250 local	ISV + VAR	Product Manager
Partner 7	50	VAR	Chief Consultant
Partner 8	180	ISV + VAR	Consultant
Partner 9	1800 global/80 local	VAR	Product Manager Consultant
Partner 10	23	ISV	CEO Product Manager
Vendor	90000 global/1000 local	Vendor	Product Marketing Manager Partner Technology Advisor

Table 1 – Participating companies in the study

3.2. Data analysis

In following the guideline of iterative conceptualisation, the analysis of the data began after the first two interviews were conducted with the consultant in Partner 3 and the CEO of Partner 10. The interviews were analysed using open, axial, and selective coding [14] and the coding process was aided by the use of the ATLAS.ti software [25]. Open coding consisted of conceptualising the text in the 246 pages of transcripts of the interviews on a line-by-line basis by marking each line, or occasionally a few words, and assigning a particular concept to that piece of data. While during the stage of open coding, theoretical memos were written to stimulate theoretical sensitivity. The process proceeded to the phase of axial coding in which the concepts were grouped into categories and the concepts and categories were related to each other, resulting in a total of 41 concepts in three categories. Finally, the phase of selective coding entailed the selection of core categories to which other categories and concepts were related. After the first iteration of coding, the concepts and categories were far from saturated and many new questions arose.

The collection and analysis of the remaining 12 interviews focused on saturating and extending the concepts and categories by selecting companies and interviewees based on the guideline of theoretical sampling. A non-sequential iteration of open, axial, and selective coding continued through the remaining analysis, and by the end of the final iteration of coding, more than a thousand instances of data had been coded into 22 overall concepts in 9 categories, and numerous theoretical memos of various lengths had been written through the coding process. The final concepts and categories included in the emerging theory were discussed with other researchers to improve reliability of the study [26]. The appendix shows the distribution of concepts across categories along with examples of coded data that led to the concepts.

4. The emerging theory

The theory emerging from the analysis of the study revolves around the version transitioning that the consultants go through, as illustrated Figure 2. The figure shows the categories and concepts emerging through the analysis of the study and how they interact with each other, and depicts the paths through the transition process that the consultants go through

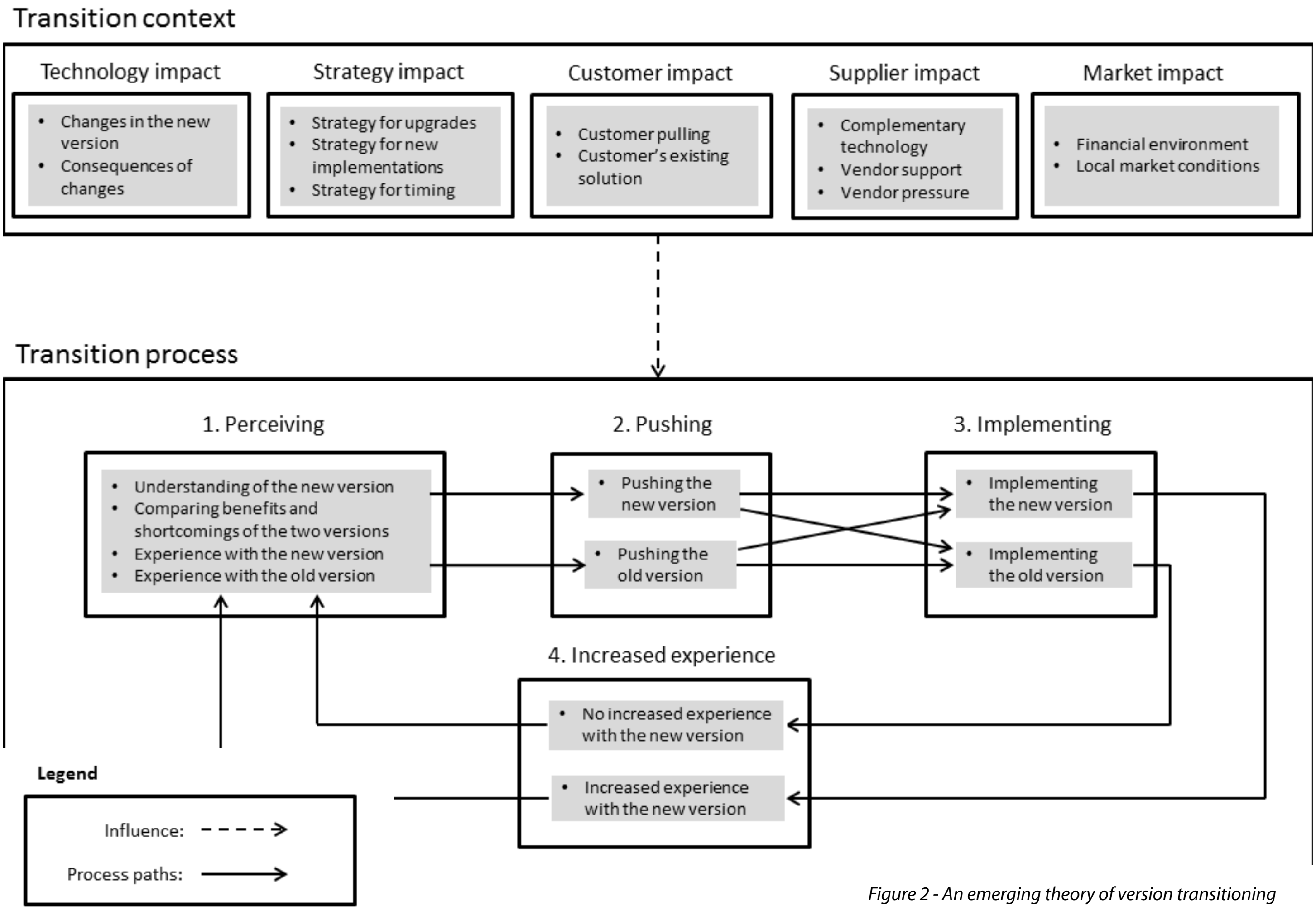

Figure 2 - An emerging theory of version transitioning

every time they are faced with the prospect of selling an implementation of the pre-packaged enterprise system (lower part of Figure 2), and the transition context that influences the process, (upper part of Figure 2). The presented categories and concepts are not proposed as being exhaustive, and only the most central and saturated concepts are presented. In the text describing the emerging theory, both concepts and categories are typeset using italics but only categories have their first letter capitalised.

4.1. The transition process

The process of transitioning to implement a new version begins with the category of Perceiving (stage 1). The category includes the concept of an understanding of the new version in which the consultants attempt to understand the changes that have been made in the new version of the pre-packaged enterprise system as compared to the old version. The concept of understanding of the new version is closely tied to the concept of comparing benefits and shortcomings of the two versions in which the consultants compare advantages of one version over the other in different areas. The concepts of experience with the old version and experience with the new version conceptualise the consultants' experience with implementing the two versions respectively.

When the consultants face the prospect of selling an upgrade or a new implementation to a customer, the Pushing (stage 2) is initiated. At this stage the consultants are either pushing the new version or pushing the old version when discussing implementation with the customer, depending on the outcome of the Perceiving stage.

Once the customer has decided which of the two versions to buy, the process moves to Implementing (stage 3) in which the consultants are either implementing the new version or implementing the old version for the customer. Even though the consultants push one of the two versions at the Pushing stage, the customer may still decide not to follow the push from the consultants. The paths from the Pushing stage to the Implementing stage may thus cross, as illustrated by the crossing of the paths in Figure 2.

Once the implementation is carried out, the consultants go through the stage of Increased experience (stage 4). If the consultants were implementing the new version in the Implementing stage, increased experience with the new version is gained, which in turn influences the Perceiving stage at the concept of experience with the new version. If the old version is implemented, no increased experience with the new version is gained and no influence is exercised on the Perceiving stage. On the other hand, if the consultants were implementing the old version in the Implementing stage, experience with the old version is gained and the Perceiving stage is influenced at the level of experience with the old version, causing pushing the old version at the Pushing stage.

4.2. The transition context

The transition process is influenced by a number of contextual categories. The category Technology impact contains concepts related to the impact of the technology of the new and the old version on the transition process. The concept of changes in new version refers to the changes in the technology of the new version in itself, such as architecture and hardware requirements compared to the old version. The consequences of changes refer to the derived consequences of the technological changes, such as increased cost of implementation or speed of implementation.

The category of Supplier impact reflects influences from the other players in the ecosystem, the vendor and the ISVs, on the transition process of the consultants in the VAR companies. Complementary technology conceptualises the impact relating to the dependence on compatible add-ons of the core enterprise system package. The category also includes the concept of vendor support, such as providing formal training for the consultants, service packs, and documentation of the new version. The concept of vendor pressure reflects the pressure communicated by the vendor in an effort to persuade the consultants to start selling the new version.

Strategy impact includes the concepts related to the strategies applied by the consultants, which influences the transition process. The concept of strategy for upgrades refers to the strategy imposed by the consultants when selling to existing customers that already have a previous version of the enterprise system, and the strategy for new implementations refers to the strategy for selling to new customers with another enterprise system or no enterprise system at all. Another central concept of the Strategy impact is the strategy for timing concerning at what point in time, after a new version is released, the consultants will initially consider selling it to customers.

The Customer impact category groups concepts relating to the customers' influence on the transition process. The concept of the customer's existing solution denotes any existing solution that a customer may have. The concept influences the transition process, e.g. through the Pushing category by determining which of the two versions the consultants try to push. The customers also form and express perceptions of the new and the old version conceptualised as customer pulling for one of the two versions, potentially influencing the paths of the transition process from the Pushing stage to the Implementing stage, as previously explained in the section on the transition process.

The final category influencing the transition process is Market impact containing the concepts financial environment and local market. Financial environment reflects influence of the financial climate at any time of the transition process, and local market conceptualises conditions in the local market that may impact the transition process.

5. The emerging theory in the research context

In the following section, the categories of the emerging theory and their interaction are discussed in detail in the context of the research from which they emerged. In order to provide insight into the context for the transition process, the categories of the transition context (upper part of Figure 2) are addressed first and second the categories of the transition process (lower part of Figure 2).

5.1. Technology impact

5.1.1. Changes in the new version The new major version of the pre-packaged enterprise system studied here was launched in late 2008. The changes and additions in the new version included, among other things, a new the front-end client with a new user interface, a change in the keyboard shortcuts, a change in the way of generating and developing customised reports, and the possibility of using a different software development tool compared to the old version. Closely linked to the new front-end client was a change from a two-tier to a three-tier architecture, entailing a requirement for a new database server if the new front-end client was going to be implemented. The new version maintained the possibility of running the old front-end client from the previous version on the new version alongside the new front-end client, although the vendor announced that from the next version this possibility would be discontinued. The first release of the new version had a number of stability issues and lacked some keyboard shortcuts. To remedy these shortcomings the vendor released a service pack in the autumn of 2009.

5.1.2. Consequences of changes When addressing the consequences of the changes in the new version, some consultants suggested that the new version was more expensive to implement due the higher license fees and higher hardware requirements of the new architecture: "The new server requirement is probably one of the biggest barriers for the new version, because the old server was free." (CIO – Partner 1).

The change in shortcuts was also pointed out as a major change between the two versions by many consultants: "From the very first versions I have known, postings have always been control-F5. It has never been otherwise in any version. Now it is suddenly completely different, so the change in the shortcuts is major", said Consultant – Partner 9.

However, the largest consequence of the change between the two versions was attributed to the new front-end client. Many consultants even pointed out that the change to the new front-end client was one of the largest between any two versions in the history of the enterprise system: "It was a shift in paradigm when we went from DOS to Windows. This is a bigger change", said Product Manager – Partner 9.

5.2. Strategy impact

5.2.1. Strategy for upgrades The partners in the ecosystem expressed different transition strategies as being suitable for selling a new implementation to a new customer respectively selling an upgrade to an existing customer. Some of the consultants feared the new front-end client would be difficult for existing customers and end-users to adjust to: "Unless they were new customers we didn't recommend [the new front-end client]. We did implement the new version but not with [the new front-end client]." (Consultants – Partner 8). Others saw the new front-end client as an opportunity for the existing users to replace previous customisations of the interface, made by the consultants, with the users' own personalisation. Some of the consultants also emphasised the importance of the first implementation of the new version being at an existing customer: "Know your customer. It is very important when you make a transition of technology at this level that you know your customer" (Unit Manager – Partner 2).

5.2.2. Strategy for new implementations The new front-end client was generally perceived as easier for new customers to adjust to: "[The new customers] are ready for change. They know that they have to adjust to a new user interface", said Consultant – Partner 8, and Consultant – Partner 9 added that: "Many new users think [the new front-end client] looks good".

The issue of new versus existing customers was intensified by the vendor advising that the new front-end client should only be sold to new customers while existing customers should keep the old front-end client when upgrading to the new version. "When you as a consultant hear that they [the vendor] only recommend it to new customers how much do you really believe in it then? […] I think that announcement has pushed the whole thing by a full year." says CIO – Partner 1.

5.2.3. Strategy for timing "Every consultant says "no thanks" every time something new comes along […]. Very few [of our consultants] go with the first release of a new version. Let the others take the beating first and then we join in later", says Chief Consultant - Partner 7, as an example of a strategy of beginning to sell the new version to customers late. The vendor's Product Marketing Manager confirms that this is a strategy of many consultants: "[The consultants] are very conservative. They stick to what they know", and elaborates that many of the owners of the smaller consulting companies are close to retirement and do not want to make the investments to carry out the version transitioning. Other consultants had a transition strategy of making the version transitioning as early as possible: "I am always in favour of implementing the newest version, if it makes sense for the customer" (Consultant – Partner 5).

The issues with the first release of the new version were also frequently mentioned as a reason for late transition timing: "We said, we don't want to touch [the first release] and so we waited for the first service pack. When that came we evaluated it and found that now it was working and then we could begin to move existing customers [to the new version]", said Consultant – Partner 8. Finally, the difficulties of understanding the technological changes in the new version were perceived as a cause for late transition timing by some respondents.

5.3. Customer impact

5.3.1. Customers pulling Even when the consultants did not feel completely ready for implementing the new version, some of the customers still had a positive impression of it, and asked that the consultants implemented the new version instead of the old: "It was actually the customer that asked for [the new front-end client]. I was not ready to implement it yet because i did not feel I had a complete overview of how to do it, so I just had to catch up" (Consultant – Partner 8). At other times the customer chose the old version over the new, even when the consultants were pushing for the new version.

5.3.2. Customer's existing solution As described above in the section about Strategies for upgrades, the strategies deployed by the partners were different when selling a solution to an old compared to a new customer. This entailed that the customer's existing solution became an import concept in the transition process of the new version, especially since most customers already had an existing solution: "They always have something", said Product Manager – Partner 9.The partners also explained that the existing solution was also generally used as reference when implementing a new version: "[The customer's] existing solution fulfils an existing need that we also fulfil with the new version. You cannot implement a new version that does not fulfil that need", said CEO – Partner 10. Moreover, the frequent occurrence of customized implementations entailed that upgrading from previous versions to the new version of the system required considerable consultant resources to ensure that customer specific customizations would be compatible with the new version.

5.4. Supplier impact

5.4.1. Complementary technology As the new version of the core enterprise system package in the study included substantial changes to the architecture and a new front-end client, some of the frequently used add-ons were not fully upgraded to work with all aspects of the new version before late 2010, nearly two years after the new version was released.

The vendor's Product Marketing Manager and many of the consultants explained that regardless of the customer type nearly all implementations included one or more add-ons to complement the core package: "I cannot imagine carrying out an implementation without any add-ons" (Unit Manager – Partner 2). This was especially the case for vertically specialised customers but also more horizontally oriented customers, such as small trade companies, required a number of add-ons, such as payroll and online banking, in order for the solution to meet their requirements. This entailed that the consultants were dependent on the ISVs to deliver new versions of the add-ons that were compatible with the new version of the core package: "One of the major factors in this has been that some of the add-ons we always implement when we are selling have not been ready for [the new front-end client]. And many of the add-ons have only been ready within the past three months so we have not been able to deliver the solutions we wanted", said Product Manager – Partner 6.

The ISVs in turn were depending on the vendor to deliver documentation for the code and executable code before being able to upgrade the add-ons: "[The ISVs] have been waiting for some fundamental elements from [the vendor]" says Unit Manager – Partner 2, linking the concept of complementary technology to the concept of vendor support.

The ISVs also appeared to be driven by a demand from the VARs before they began to upgrade their solutions: "There is no doubt that the ISVs have massive expenses associated with this transition […] they are very demand driven, so when we ask for [an upgrade of an add-on] they evaluate it carefully if they haven't already [upgraded it]" (Unit Manager – Partner 2).

5.4.2. Vendor support The vendor supported the transition from the old to the new version in a number of ways. First, the vendor provided service packs which included updates and technical fixes for the new version. Second, the vendor offered a vast amount of documentation in the form of white papers, web casts, blogs, and implementation guidance for supporting the various steps in the implementation process of the enterprise system. The vendor also provided formal training and certification for the consultants, aimed at explaining the new features and underlying technology of the new version.

Finally, the vendor ran a number of projects together with key ISVs and VARs prior to the release of every major version. The projects were primarily aimed at testing the new version in a real-world customer company. However, for the partner companies it also served as an opportunity for testing the new version before it was released, while simultaneously getting special support from the vendor.

In addition to the regular projects, the vendor also organised a special workshop for six selected consulting companies 14 months after the initial release of the new version, specifically aimed at explaining the potential benefits of the new front-end client: "Then we participated in [the workshop]where we went more in-depth with the ideas and that was really an eye-opener. The ideas are extremely well-thought, but extremely poorly communicated to the consultants." says CIO – Partner 1.

5.4.3. Vendor pressure The vendor applied a lot of pressure on the consultants to make the transition to the new version: "[We] push a lot for things to change – perhaps too much. They feel stressed and then they rely on what they know." says the vendor's Product Marketing Manager. However, some of the consultants also indicated that the pressure from the vendor was necessary in order for the ecosystem to speed up the transition.

Contextual categories	Barriers	Enablers
Technology impact	Poor stability Changed keyboard shortcuts Higher license fees Increased hardware requirements Poor fit between existing users and new front-end client	New front-end client was "future proof" New front-end client appeals to new customers and users Less need for customisation of user interface
Supplier impact	Pressure from the vendor Lack of add-on compatibility	Pressure from the vendor Support from the vendor
Customer impact	Pull for old version Pull for new version with old client	Pull for new version
Market conditions	Financial crisis Saturated market	Market leadership

Table 2 – Barriers and enablers of transition to the new version

5.5. Market impact

5.5.1. Financial environment Some respondents pointed out that the financial environment had a substantial impact on the transition from the old to the new version: "There is no doubt that the timing has been bad, because right after the release, the financial crisis came crashing down and that means that none of the consultant companies has been willing to make the required investments in training and so they cling to the old version because they know they can make some money on that […] I don't think we would have made the investment [in upgrading the add-ons] if we had begun half a year later.", said CEO – Partner 10, referring to their participation in one of the vendors projects prior to the initial release.

5.5.2. Local market conditions "[In other countries] the product does not have the same market share as it does here. [In our local market] any company that considers acquiring an enterprise system will consider [our products]. They may not end up buying them but the will consider them. So we do not have to put up big posters in the airport like many others have to", said the vendor's Product Marketing Manager, indicating a market leadership in the local market, which was also confirmed by documentation. The consultants also suggested that the local market was somewhat saturated, meaning that most implementations were either upgrades of existing customers with an older version or customers that had another enterprise system.

In summarising the contextual impact on the transition process of the consultants in the study, Table 2 illustrates the distribution of the expressed barriers and enablers. Note that pressure from the vendor is categorised as both a barrier and an enabler, as findings from the study indicated this as both hindering and enabling the transition process.

5.6. Perceiving

This section describes the stages of the transition process of the emerging theory in the context of the research study and exemplifies the contextual impact on the transition process.

5.6.1. Understanding the new version The initial understanding of the new version was hard for some of the consultants: "It is rather complicated to get [the new version] running and it is something we have never done before, because the whole technology is different." says Product Manager – Partner 9. Especially the changes in the new front-end client caused a great deal of difficulties in understanding: "It is a new technology and a new way of thinking" (Product Manager – Partner10).

5.6.2. Comparing benefits and shortcomings of the two versions The benefits expressed by the consultants were primarily related to the increased usability of the new front-end client in terms of possibilities of personalisation for the individual user: "The users can put their personal touch on [the new front-end client] to achieve the approach that is best for them and that part is really cool", explained CEO – Partner 10, and the consultants generally perceived the new front-end client as more "future-proof" that the old client: "It is the only way to go. The [old front-end client] is old in the worst kind of way. It is just not up to date on how you do things today. That goes for the technical aspects as well as the usability. You cannot display a graph in the old client. I mean, we are talking 2010 and you cannot display a graph. What is going on?" said Product Manager – Partner 9.

Many respondents pointed out that the development of reports was easier in the old version: "We have had a report generator that all consultants are world champions in using. Then [the vendor] decided that when you run the [new front-end client] you have to use this new technology […] and that part should have been done differently" (Product Manager – Partner 6). The conversion of old reports to fit the new version was also perceived as a challenge: "One thing is that it takes a long time but is also extremely boring. Nobody wants to do it. It really has to be an emergency before I do it", said Product Manager – Partner 9.

Finally, all the consultants explained that the stability issues and bugs in the first release of the new version had significant negative impact on the transition: "The first release should never have been released because it was straight out unusable." (Product Manager – Partner 6).

5.6.3. Experience with the new version The experience with the new version was limited for many of the consultants: "Even though we have been working with the new version since 2007, we still have more experience with the old version" explained Product Manager – Partner 10. The consultants also pointed out that experience with implementing the new version entailed a more positive perception of it: "Once I get [the new version] under my skin then I think it will be fantastic. So if the customers are buying from me, then they will get [the new version]" (Consultant – Partner 5).

5.6.4. Experience with the old version Many of the consultants in the ecosystem had substantial experience with implementing previous versions of the pre-packaged enterprise system: "Many of the consultants have been in the business for 20-25 years", explained the vendor's Product Marketing Manager. The consultants pointed out that regardless of which version was implemented, it typically took in excess of two years before a new consultant had in depth experience of how to implement the system. The extensive experience with the old version and the lack of experience with the new version caused many of the consultants to push for implementing the old version at the customers.

5.7. Pushing

5.7.1. Pushing the new version During the study, several examples were found of partners pushing the new version to the customer: "So we asked [the customer] if they were interested in [the new version]. […] So I would not say it was the customer that initiated it. We initiated it and convinced them", said Consultant – Partner 3.

5.7.2. Pushing the old version The respondents explained that when the customers ended up choosing the old version it was often due to a push from the consultants: "I don't believe it is the customers that choose the [old front-end client]. It is the partners. And when we are under pressure we do the same thing. We say, let us start out with [the old front-end client] and then we can switch over to [the new front-end client] later[…] If [the customers] had a 100% free choice then I think they would always choose [the new front-end client]. It is definitely the partners that push the old one to the customers and then promise them that they can upgrade later. And we all know that is probably not going to happen once you have begun the implementation", said CEO – Partner 10. The Product manager of Partner 9 also confirmed that they were driven by a demand for the new version rather than pushing it: "We are driven by customers asking for [the new version]" (Product Manager – Partner 9).

5.8. Implementing

5.8.1. Implementing the new version Some of the implementations did result in a the new version with the new front-end client being implemented: "[The customer] was in the process of implementing the new version with the old client but then they saw [the new front-end client] and did not want to have the old one implemented" said Chief Consultant – Partner 4.

5.8.2. Implementing the old version The partners explained that the push for the new version did not always result in the new version being implemented and when it did, it often did not include the new front-end client. The vendor's Product Marketing Manager supported this impression by explaining that one year after the new version was released, only very few customers' had purchased a license for the new front-end client.

5.9. Increased experience

5.9.1. No increased experience with the new version The respondents stressed that if the consultants did not implement the new version they could not gain any experience with it: "They are not world champions when they are done with [the training courses] because you only become that through working with practical cases and it is only customer implementations which gives that" (Product Manager – Partner 6). Due to various contextual factors, little new experience was gained when the old version was implemented at a customer: "[…] when you have done 50 implementations [of the old version] then there is not much new" (Chief Consultant – Partner 7).

5.9.2. Increased experience with the new version The consultants explained that the first couple of implementations with the new version gave them a significant increase in experience: "We knew this was new territory but also that this is the way the wind is blowing. So it was an option for us for getting to know [the new version]. And we succeed with it through blood, sweat, and tears and gained experience", says Chief Consultant – Partner 3, when referring to his first participation in an implementation of the new version.

6. Discussion of findings

The findings from the study indicates that poor stability of the first release of the new version, and other barriers associated with the technology of the new version, were some of the main barriers for transition of the new version among the implementation consultants in the ecosystem. Many consultants considered the first release too unstable to implement in customer organisations, and thus the ISVs had little incentive to upgrade their add-ons to be compatible with the new version. When the service pack was released by the vendor and the new version was considered mature enough to implement, the lack of upgraded add-ons was evidently perceived as a barrier, causing inertia in the version transitioning. The study thus illuminates some of the challenges of software ecosystems in respect to transitioning to a new version of a pre-packaged enterprise system by highlighting the dependence on complementary technology, in the form of add-ons, in order for the consultants to deliver a complete solution of the enterprise system package to the customer. The findings thus support the importance of addressing business strategies from a network perspective rather that looking at individual companies in isolation [27].

The influence of increased experience on the Perceiving stage of the transition process suggests a reinforcing effect in the transition process once initial experience is gained with implementing the new version. The crossing paths in the transition process between the stages of Pushing and Implementing (see Figure 2) further indicate that the customer's pull for one of the two versions can change the pursued transition paths of the consultants, hence enabling or hindering the transition to the new version of an enterprise system. The findings are thus consistent with previous suggestions that neither a technology-push nor a customer-pull perspective in isolation is sufficient for understanding adoption and diffusion of innovations [28]. Instead, a more integrated perspective is needed. To reach such an integrated perspective the guideline of theoretical integration in the Grounded Theory methodology may help integrating the emerging substantive theory with existing diffusion theories as an initial step towards creating formal theory [13].

6.1. Integrating the emerging theory

Previous research on adoption of innovations has addressed the stages in the adoption process of innovations. The adoption process in diffusion theories has been conceptualised differently by different researchers, but a particularly useful approach for integrating the transition process may be the two-stage adoption process of Initiation and Implementation as suggested by various authors [29-31]. In this view, the Initiation stage consists of activities related to perception, information gathering, and attitude formation leading to the decision to adopt, and the Implementation stage consists of events and actions pertaining to modifications in both the innovation itself and the organisation and utilisation of the innovation [29]. The emerging theory of version transitioning from the research thus resembles both of these aspects, in that the categories of Perceiving and Pushing are comparable to the Initiation stage and the Implementing category is comparable to the Implementation stage.

Integrating the transition process part of the emerging theory with the stages of Initiation and Implementation may thus provide an appropriate lens through which to scale up the emerging theory and reach a higher level of generalisation. As described above, the first three of the four stages in the emerging theory are readily comparable to the stages of Initiation and Implementation. However, the stage of Increased experience in the emerging theory falls between the categories in the two-stage conceptualisation. Preserving the relationship between Increased experience and the grounded categories of Implementing and Perceiving, the integrated theory suggests iteration between the categories of Initiation and Implementation, as illustrated in Figure 3.

While the transition process stages of the emerging theory may thus be integrated with innovation adoption stages in existing diffusion theories, the context categories of the emerging theory should be compared to contextual, rather than processl, factors from existing theories. Existing diffusion theories suggest a number of contextual factors that may enable or inhibit the diffusion process. While several of these contextual factors may be comparable to the contextual factors of the emerging theory of version transitioning, Orlikowski's [32] study of adoption of CASE tools as a process of organisational change may be particularly suited for theoretical integration with the contextual factors of the emerging theory.

Orlikowski proposes three contextual categories which influences adoption and use: IS Context, Organisational Context, and Environmental Context. Integrating the contextual categories of the emerging theory with these contextual categories provides a suitable foundation for integrating the theory and generalising the context categories. In this perspective the categories of Supplier impact, Customer impact and Market impact can be compared with the Environmental Context, the category of Strategy impact with the Organisational Context, and Technology impact with the IS Context, all of which influence the adoption stages. Furthermore, Orlikowski (ibid.) also proposes that the contextual categories themselves are influenced by the adoption process as it progresses as depicted in Figure 3.

Although the purpose of theoretical integration in the Grounded Theory methodology is not to apply the integrated theory back to the data set from which parts of the theory emerged, the integrated theory of version transitioning has more explanatory power compared to the emerging theory. First, the division of the transition process categories into Initiation and Implementation provides distinction between the "planning" activities (Initiation) in which the consultants, often prior to the release of the new version, would form a perception and strategize about the transition to the new version and the "action" activities (Implementation) in which the strategy for transition to the new version would be executed and subsequently revised based on increased experience. Second, extension of the emerging theory with the reciprocal relationship between process and context fits and extends the emerging theory to assist in understanding of the mutual influence on the players in the ecosystem, including the push/pull configuration between the VARs and the ISVs in regards to development of compatible add-ons and the mutual influence between VARs and their customers in regards to selection of the new or the old version. Finally, the division of the contextual categories of the emerging theory into Environmental, Organisational, and IS context provides a clearer view of which overall areas the contextual categories of the emerging theory are attributable to, which, in turn, provides general indications for if and how the categories can be influenced by the actors in the ecosystem.

7. Conclusions

The study of transition from an old to a new version of an enterprise system in an ecosystem context has provided an opportunity for theorizing about the transition process that partner companies undergo, and the contextual factors that influence and are influenced by the transition process. The emerging theory thus provides us with initial understanding of how actors in software ecosystems experience enterprise system version transitioning, and also illustrates the substantial effect the phenomenon has on the consultant companies in the ecosystem. The emerging theory suggests the transition process is an iterative process in which the actors repeat each stage in the process multiple times before the transition is complete, as opposed to traditional adoption theory in which the stages are only undergone once by each adopter for a particular innovation [30]. Although the introduction of a new version of a an enterprise system in the ecosystem will eventually lead to the discontinuation of the old version, the process resembles that of a gradual transition rather than adoption at one particular point in time, and aligns with the perspective that "as innovation develops and diffuses, learning occurs; the old and the new exist concurrently, and over time these are linked together" [33].

8. Implications for practice and future research

The research presented in this paper suggests that managers in software vendor companies orchestrating ecosystems indeed need to pay close attention to the dependencies on complementary technology in software ecosystems. Just as important, the interconnectedness of players in the ecosystem also entails that there is little gain in releasing inferior or unstable releases of new versions in the expectation that bugs and shortcomings can be fixed along the way, as rejection in any part of the ecosystem causes a barrier for transition in other parts. Finally, managers and consultants should consider the reinforcing effect of experience gained from implementing new versions of pre-packaged enterprise systems as indication of the value of facilitating trial of implementations through,

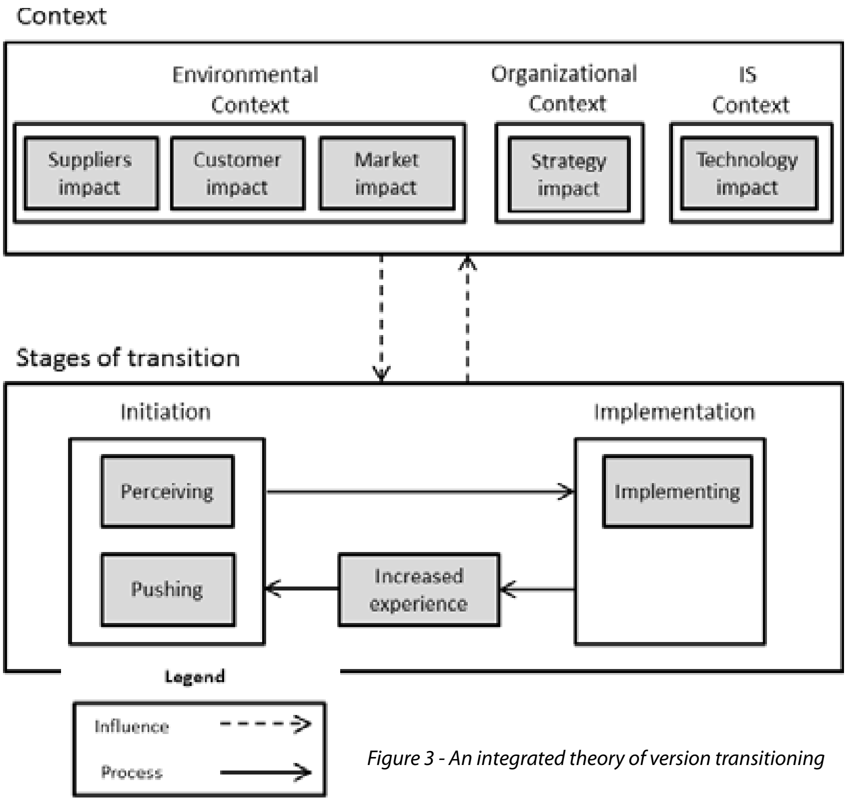

Figure 3 - An integrated theory of version transitioning

e.g. wider investment in formal adoption programs and influencing of potential early adopters among customers.

The inherent limitations of building theory from the study of transition of a single new version in a single ecosystem suggest that future research should look into version transitioning and adoption in other software ecosystems. Version transitioning in other types of ecosystems with different configurations of actors should be investigated to further extend the current integration of the emerging theory into a more generalizable formal theory. Furthermore, the research presented in this paper leaves room for future studies of the effects of supporting the simultaneous use of two different front-end clients on the same version of an enterprise system as a means of allowing partial and even more gradual transition to a new version. Finally, the ambiguous findings of the effects of vendor pressure on the transition process suggest further research in this area. Future studies may thus benefit from a holistic network perspective on the influence applied by the different actors in software ecosystems.

References

[1] Campbell-Kelly, M., From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry. 2003: The MIT Press, Cambridge, USA.

[2] Arndt, J., T. Kude, and J. Dibbern, The Emergence of Partnership Networks in the Enterprise Application Development Industry: A Global Corporation Perspective, in Advances in Information Systems Research, Education and Practice, D. Avison, et al., Editors. 2008, Springer: Boston. p. 77-88.

[3] Messerschmitt, D. and C. Szyperski, Software ecosystem: understanding an indispensable technology and industry. 2005: MIT Press Books.

[4] Adner, R., Match your innovation strategy to your innovation ecosystem. Harvard Business Review, 2006. 84(4): p. 98-107.

[5] Huber, T., T. Kude, and J. Dibbern. Resolving Tensions in Hub-and-Spoke Networks of the Enterprise Application Software Industry–An Exploratory Case Study. in AMCIS 2010. 2010.

[6] Fox, P., J. Wareham, and J.L. Cano. Designing a Software Ecosystem for Co-Creation: An Empirical Study of the Plaftform Infrastructure of an Enterprise Software Channel. in ECIS 2009. 2009.

[7] Volkoff, O. and S. Sawyer, ERP implementation teams, consultants, and information sharing, in AMCIS 2001 Proceedings. 2001.

[8]] Bingi, P., M. Sharma, and J. Godla, Critical Issues Affecting an ERP Implementation. Information Systems Management, 2001. 16(3): p. 7-14.

[9] Gable, G.G., Consultant engagement for computer system selection : A pro-active client role in small businesses. Information & Management, 1991. 20(2): p. 83-93.

[10] Kude, T. and J. Dibbern, Tight versus Loose Organizational Coupling within Inter-Firm Networks in the Enterprise Software Industry-The Perspective of Complementors, in AMCIS 2009. 2009. p. 666-675.

[11] Johansson, B. and M. Newman, Competitive advantage in the ERP system's value-chain and its influence on future development. Enterprise Information Systems, 2010. 4(1): p. 79-93.

[12] Jacobs, F.R. and F.C.T. Weston Jr., Enterprise resource planning (ERP) - A brief history. Journal of Operations Management, 2006. 25(2): p. 357-363.

[13] Glaser, B.G. and A.L. Strauss, The discovery of grounded theory: Strategies for qualitative research. 1967: Aldine Transaction.

[14] Strauss, A. and J.M. Corbin, Basics of qualitative research: Grounded theory procedures and techniques. 1990: Sage Publications, Inc.

[15] Urquhart, C., The evolving nature of grounded theory method: The case of the information systems discipline, in The Sage handbook of grounded theory, A. Bryant and K. Charmaz, Editors. 2007, Sage. p. 339–359.

[16] Holton, J.A., The coding process and its challenges, in The Sage handbook of grounded theory, A. Bryant and K. Charmaz, Editors. 2007, Sage. p. 265-289.

[17] Glaser, B.G., Basics of grounded theory analysis: emergence vs forcing. 1992: Sociology Press.

[18] Urquhart, C., H. Lehmann, and M.D. Myers, Putting the 'theory' back into grounded theory: Guidelines for grounded theory studies in information systems. Information Systems Journal, 2010. 20(4): p. 357-381.

[19] Glaser, B.G., The grounded theory perspective III: Theoretical coding. 2005: Sociology Press.

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

[21] Stern, P.N., On solid ground: Essential properties for growing grounded theory, in The Sage handbook of grounded theory, A. Bryant and K. Charmaz, Editors. 2007, Sage. p. 114-126.

[22] Glaser, B.G., Doing formal theory, in The Sage handbook of grounded theory, A. Bryant and K. Charmaz, Editors. 2007, Sage. p. 97-113.

[23] Angrosino, M.V., Recontextualizing Observation, in The Sage Handbook of Qualitative Research, N.K. Denzin and Y.S. Lincoln, Editors. 2005, Thousand Oaks Calif. : Sage Publications.

[24] Kvale, S. and S. Brinkmann, Interviews: An introduction to qualitative research interviewing. 2008: Sage.

[25] Muhr, T., ATLAS/ti - A Prototype for the Support of Text Interpretation. Qualitative Sociology, 1991. 14(4): p. 349-371.

[26] Miles, M.B. and A.M. Huberman, Qualitative Data Analysis: An expanded sourcebook. 2nd ed. 1994: Sage Publications.

[27] Iansiti, M. and R. Levien, Strategy as ecology. Harvard Business Review, 2004. 82(3): p. 68-81.

[28] Rothwell, R., Successful industrial innovation: critical factors for the 1990s. R&D Management, 1992. 22(3): p. 221-240.

[29] Damanpour, F., Innovation type, radicalness, and the adoption process. Communication research, 1988. 15(5): p. 545-567.

[30] Rogers, E., Diffusion of innovations. 5th ed. 2003, New York: Free Press.

[31] Zaltman, G., R. Duncan, and J. Holbek, Innovations and organizations. 1973: John Wiley & Sons.

[32] Orlikowski, W.J., CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development. MIS Quarterly, 1993. 17(3): p. 309-340.

[33] Baskerville, R. and J. Pries-Heje, Diversity in modeling diffusion of information technology. The Journal of Technology Transfer, 2003. 28(3): p. 251-264.

10. Appendix

Transition process		
Categories	Concepts	Examples of data from the study
Perceiving	Understanding of the new version	“It is seriously a different way of thinking”(Product Manager – Partner 10) “You have to understand the concept of [the new version] to see the point” (CIO – Partner 1)
	Comparing benefits and shortcomings of the new version	“Much of the key functionality from [the old version] was not there” (Product Manager – Partner 6) “[The new reporting tool] has some tools that are much smarter than the old reports” (Consultant – Partner 3)
	Experience with the new version	“I only have experience from one implementation” (Consultant – Partner 3) “It was very new to me” (Chief Consultant – Partner 7)
	Experience with the old version	“[…] and I had much experience with the old version […]” (Consultant – Partner 5) “[…] the classic version that we are used to […]” (Product Manager – Partner 6)
Pushing	Pushing the new version	“So we asked [the customer] if they felt like trying out [the new version]” (Consultant – Partner 3) “[…] and that convinced them” (Unit Manager – Partner 2)
	Pushing the old version	“The are many that offer the old version” (Product Manager – Partner 6) “[The new version] was not interesting for us to try to push […]” (Consultant – Partner 8)
Implementing	Implementing the new version	“We have actually carried out a relatively large project of [the new version] where 30 users got [the new version]”(Chief Consultant – Partner 4) “The is not doubt that when you are implementing [the new version] then […] ”(CEO – Partner 10)
	Implementing the old version	“I was once in an implementation of [the old version]…” (Consultant - Partner 5)
Increased experience	Increased experience with the new version	“So we got our pilot project and a lot of experience” (Chief consultant – Partner 4) “Part of implementing [the new version] at the customer is also a matter of training for us […]” (Unit Manager – Partner 2)
	No increased experience with the new version	“[…] when you have done 50 implementations [of the old version] before, then there is not much new” (Chief Consultants – Partner 7) “[…] most of it you do not get “into the spine” unless you do implementations [of the new version]” (Consultant – Partner 9).
Context		
Categories	Concepts	Examples of data from the study
Technology impact	Changes in the new version	“The change in the keyboard shortcuts is huge” (Product Manager – Partner 9 “[The vendor] chose to use a new technology for the reports in the new version” (Product Manager – Partner 6)
	Consequences of changes	“Developing a report [in the new version] takes longer than in the old version” (CEO – Partner 10) “It takes half a day to install the old version in the new it takes at least three days” (Consultant – Partner 9)
Supplier impact	Complementary technology	“One of the major factors in this is the [compatibility] of the add-ons we always offer in the implementation” (Product Manager – Partner 6) “That is a little special about our business because we nearly always use add-ons for both payroll and online banking” (Consultant – Partner 8)
	Vendor support	“I think the information [the vendor] provided was OK. They put up some good examples on blogs…” (Chief Consultant – Partner 4) “[…] also in relation to the attention we get from [the vendor]” (CIO – Partner 1)

	Vendor pressure	"We pressure, pressure, pressure the partners" (Product Marketing Manager – Vendor) "[…] in order to keep a certain status with [the vendor]" Product Manager – Partner 6)
Customer impact	Customers pull	"[…] so it was actually the customer that asked for [the new version]" (Consultant – Partner 8) "The customer would not implement the old version" (Chief Consultant – Partner 4)
	Customer's existing solution	"Their current system[…]" (CEO – Partner 10) "[…] and because the system they had was out dated […]" (Consultant - Partner 8)
Strategy impact	Strategy for timing	"We want to be on the newest technology" (Unit Manager – Partner 2) "Only very few go with the first release" (Chief Consultant – Partner 7)
	Strategy for new implementations	"We have had the approach with selling to the new customers" (CIO – Partner 1) "All new implementations are [the new version]" Unit Manager (Partner 3)
	Strategy for upgrades	"Most of the times where we implement the new version are new implementations" (Product Manager – Partner 9) "Whether we recommend existing customers to upgrade is a totally different matter" (Chief Consultant – Partner 7)
Market impact	Financial environment	"There is no doubt that the timing in the market has been very unfortunate" (CEO – Partner 10) "[…] but then the financial crisis struck and now it is on hold" (Chief Consultant – Partner 7)
	Local market conditions	"Because [the local market] is so small […]" (CIO – Partner 1) "[…] and perhaps that is because of [the local market] and the wide spread of [the enterprise system]" (Unit Manager – Partner 2)

Keywords:
Grounded theory, enterprise systems, software ecosystems, implementation consultants, adoption, diffusion of innovations

Philip Holst Riis
Center for Applied ICT
Copenhagen Business School
Copenhagen
Denmark

E-Mail: ph.caict@cbs.dk
Phone: +45 21 81 18 80