

AMBIDEXTROUS ENTERPRISE ARCHITECTING: BETTING ON THE FUTURE AND HACKING PATH-DEPENDENCIES

Complete research

Knut H. Rolland, Westerdals Oslo School of Arts, Communication and Technology and SINTEF ICT

Gheorghita Ghinea, Brunel University and Westerdals Oslo School of Arts, Communication and Technology

Tor-Morten Grønli, Westerdals Oslo School of Arts, Communication and Technology

Abstract

Much existing research on IS and enterprise architecture focuses on suggesting different representations describing a ‘target architecture’ for processes, information and information systems at different levels across the organization. Less emphasis has been put on the actual evolution of the current enterprise architecture towards this envisioned ‘target architecture’. In this paper we look at enterprise architecture from a process perspective and explore how actual practices of architecting unfold and how they are mutually shaped by context. In doing so, we offer two contributions. First, we contribute empirically showing how enterprise architecting in practice is an ambidextrous activity focusing on both fixing the limitations of historically entrenched architectures and establishing new architecture by betting on what needs are going to give the most flexibility and value in the future. Secondly, we contribute by conceptualizing how enterprise architecture evolves over time, and propose the concepts of ‘architectural path-dependency’ and ‘enterprise architecting’. Architectural path-dependencies is the effect of existing architectures that become economically, technically and organizationally hard to change – at least in a radical manner. Enterprise architecting, on the other hand, is the intentional acts to circumvent path-dependencies and evolve towards an envisioned architecture. Henceforth, we argue that enterprise architecture transition is shaped by multiple architectural path-dependencies as well as various acts of enterprise architecting to establish new paths. The paper discusses these concepts in relation to existing literature on software and enterprise architecture.

Keywords: Enterprise architecture, Enterprise architecture as a process, Path-dependency, Case study.

1 Introduction: Current Perspectives And Blind Spots

Today's organizations – more often than not – have a complex archipelago of different interconnected information systems (Hepsø et al., 2009; Tilson et al. 2010). One of the consequences of this is that organizations struggle to substitute existing legacy systems, develop new digital services, and re-organize their IS portfolios in order to align with new business needs and strategies. In the presence of such problems, a focus on different levels and aspects of information systems architecture has emerged in recent years both among practitioners and researchers (e.g. Bruls et al., 2010; Ren and Lyytinen, 2008; Ross et al., 2006; Smolander et al., 2005, 2008; Simon et al., 2013; Tiwana and Konsynski, 2010; Versteeg and Bouwman, 2006; Zackman, 1987). In this body of research, the notion of 'enterprise architecture' is frequently used to underscore that the phenomena under scrutiny is largely an enterprise-wide and socio-technical issue, distinguishes itself from the more technically oriented – but related research on software architectures (e.g. Baragry and Reed, 2001; Gacek et al., 1995; Jansen and Bosch, 2005).

As noted by Tamm et al. (2011: p. 142) the primary aim of an enterprise architecture is “to define the desirable future state of the organization's business processes and IT systems (often referred to as the “to-be” or target architecture) and to provide a roadmap for achieving this target from the current state (“as-is” or baseline architecture)”. Interestingly, much existing work is concerned with the former - defining a target architecture, rather than the latter transformation process from the baseline architecture. A target architecture in this literature often entails different ways of representing a *blueprint* for an envisioned future architecture. Recent contributions have proposed various concepts for making more nuanced and layered blueprints – such as for instance 'business architecture' (Bouwman et al., 2011), 'domain architecture' and 'solution architecture' (Bruls et al., 2010), as well as suggesting business architecture as a new way of linking strategy and IS (Versteeg and Bouwman, 2006).

As much for its merits, the predominant focus on blueprints of future architectures leaves a potential *blind spot* for how organizations should deal with current heterogeneous architectures of legacy systems, work processes and often fragmented collections of digital content towards envisioned future enterprise architecture. Arguably, this perspective is important in several ways. First, as illuminated by Smolander and colleagues (2008), in practice, a substantial amount of architectural work involves understanding past architectural decisions and ways of representing these for use in future developments and maintenance. This shows that enterprise architecture also is directed towards the historical legacy of IS, processes and digital content, and without a firm understanding of this – further evolution (i.e. a roadmap) would become difficult. There is an increasing awareness of the challenges involved in radically transforming existing legacy of IS embedded in organizations (e.g. Aanestad and Jensen, 2011; Hepsø et al., 2009; Rolland, 2000) due to path-dependencies and lock-in effects (e.g. Arthur, 1989; Ciborra, 2000). Keeping this in mind, a viable way for transforming the current enterprise architecture to a future target architecture is to proceed through small incremental steps and to bootstrap the new architecture by providing gateways and mobilizing stakeholders (Hanseth and Lyytinen, 2010). Secondly, this blind spot also illuminate the importance of architecture modularization as recently noted in the literature (Ren and Lyytinen, 2008; Ross et al. 2006). However, path-dependencies and lock-in effects (Arthur, 1989; David, 1985) also make a transition towards a more modularized enterprise architecture challenging for organizations, and without recognizing this phenomena organizations risk investing too much in strategies that turn unrealistic and that might undermine fruitful transitions to take place. Keeping this in mind, this paper aims at empirically describing and theorizing how enterprise architectures change over time in interaction with new information systems, business needs and a wider context of the organization.

This paper attempts at shedding new light on the current 'blind spot' in enterprise architecture and offers two contributions. Firstly, the paper contributes empirically by providing a longitudinal case study of enterprise architecture evolution in a global company. The case shows how existing architectures on different levels need to be incessantly modified and adjusted in order to meet emergent

changes in the company's environment, organizing structures and digital infrastructure. At the same time, new architectures are established involving 'betting on' the future needs of the company – especially because of the unpredictability of software platforms' evolution and shifting requirements for products and services provided by the company.

Secondly, building on empirical insights, we contribute by conceptualizing enterprise architecture as an on-going and ambidextrous process involving managing both past architectural decisions and future prospects for enterprise architecture. In particular we propose the constructs of *architectural path-dependency* and *enterprise architecting* in order to denote how past decisions about architecture are influencing future trajectories for enterprise architecture and how actors try to work-around these limitations respectively. Subsequently, we argue that conceptualizing enterprise architecture as a process – rather than a radical shift or just as a blueprint of a envisioned future, is important both for explaining the outcomes of enterprise architecture initiatives and also for doing more realistic architecting in practice. In this respect, we see a process perspective as an important addition to existing perspectives on enterprise architecture (e.g. Bouwman et al., 2011; Bruls et al., 2010; Ren and Lyytinen, 2008; Tamm et al., 2011).

The paper is structured as follows. The next section reviews literature that we draw from in order to set up our multidisciplinary theoretical lens for studying enterprise architecture and IS architecture in general. Then in section three, we present the case study of Bergen Drilling, a global oil and gas company, and their struggle to establish an enterprise architecture in an turbulent environments. In the following sections we focus the case and analyse how ambidextrous architecting was practiced. In section six discuss the findings from our study in light of relevant literature. Finally, we conclude the article by discussing some future possibilities for IS research on enterprise architecture.

2 Theory

2.1 Transition of software architecture

The focus on enterprise architecture has in many ways grown out of a more specific focus on software architectures to also cover many more aspects. This is helpful for both conceptualizing the interdependencies across systems and organizing, but also what often drives practitioners interests in the topic (Tamm et al., 2011). However, at the same time, it is important that the discussion on enterprise architecture does not loose track of the role and specifics of software technology. Certainly, when studying how enterprise architectures evolve, existing installed base of legacy systems and databases would play a role. More over, architectural plans and visions need to consider not only the possibility of new technologies and systems being implemented in the future – but more appropriately, how to take in to the account new technologies and systems that will be implemented.

The field of software architecture is a classic of the computer science literature and, particularly of software engineering. Although highly technical in its purest form, it is important for understanding the complexity of software systems and why they become hard to change. The reasons underlying this difficulty are numerous, and have been studied in depth by both the software systems as well as information systems literature. On the one hand, is the perceived gap between the technically-oriented software architecture and the IS-oriented enterprise architectures (Chung and Subramanian, 2007). On the other, a focus on enterprise architectures signifies that many problems facing contemporary business and organizations are related to the interconnectedness of different IS – and ways of organizing, or indeed, the more general problem of IS mis-alignment (Chan, 2002).

Of course, designing software architectures, which are resilient to change, is one way what narrows the software/enterprise architecture gap, and a considerable amount of work has been devoted to this aim. This comprises a wide spectrum of research, ranging from the design and adoption of modular software architectures with connectors (Concha et al, 2010; Lizcano et al., 2014; McArthur et al., 2002; Velasco-Elizondo and Lau, 2010), to a call for better design patterns through appropriate blueprinting (Šaša and Krisper, 2011; Smolander et al, 2008), and a push towards a better understanding of

the ecosystem in which a particular software architecture exists and the imperatives driving change within (Christensen et al., 2014; Lagerström et al., 2010).

However, change can also be thought of as a gap in itself, between a past situation and the current (or desired future), and it is this gap that architecting research leaves mainly unexplored – and forms one of the cornerstones of the research reported in the current paper.

2.2 Path-dependency and enterprise architecture transition

The concept of path-dependency derives from evolutionary economics and is often used to explain why some technologies seem to outlive newer, seemingly more appropriate and qualitatively ‘better’ technologies. As an effect of increasing returns (Arthur, 1989), path-dependency implies that historical choices of technologies and number of users greatly influences future developments. Events and designs that at first appear as insignificant can have large-scale effects in the long run (Arthur, 1989). One such example is the QWERTY keyboard design that still is the de-facto standard for keyboards although the basic design constraint of jamming hammers of the old typewriter is long gone (David, 1985).

The concept of path-dependency has more recently also been adopted in information systems research in relation to corporate-wide IT infrastructures (Ciborra, 2000; Rolland, 2000), inter-organizational systems (Zhu et al., 2006), and the diffusion of mobile services (Yoo et al., 2005). Common to these studies is that path-dependency is used to explain the socio-technical challenges and costs involved in migrating from an ‘old’ technological platform to a ‘new’ one. Organizations and users are not totally free to select whatever new technology they prefer, but are more or less locked-in by current designs. In the context of corporate-wide IT infrastructures, path-dependencies and lock-ins are common effects of interconnected information systems and ways of organizing. For example, switching to a Microsoft SharePoint platform for collaboration and document management from an existing Lotus Notes platform can be extremely challenging in large corporations (Hepsø et al., 2009).

Also digital content and metadata structures are prone to path-dependencies and lock-in effects. Shapiro and Varian (1999) note that digital content in itself can lead to lock-in because of the specific ways large amounts of data is structured in databases and Information Systems. Especially, they argue that information-based path-dependency and lock-in seems to be more durable than lock-ins based on physical equipment. For example, in a case study by Rolland (2000) a Ship Classification Company experienced path-dependency and lock-in based on the particular structure of the information of a large database of ship information. This had multiple consequences for establishing a new information system based on a very different information architecture referred to as a product model.

Finally, path-dependency is also linked to innovation and learning. In this stream of research, path-dependency denotes how past focus on certain types of architectural knowledge (Henderson and Clark, 1990) and innovation processes seem to narrow the scope for future innovation processes and possibilities. Therefore, companies might be better off by simply “avoid being “stuck” to a particular path, and remain open to different (and competing) ideas/solutions” (Fagerberg et al., 2005: p. 10). This reminds us that path-dependency is closely connected to competence (e.g. Zhu et al., 2006) and learning – and can in this respect also lead to ‘learning failure’ (Lyytinen and Robey, 1999).

Taken together, we see that path-dependency and lock-in effects are likely to be found on different levels of enterprise architectures. Distinguishing between process architecture, information architecture and application architecture as often done in the literature (e.g. Bouwman et al., 2011; Bruls et al., 2010) we can identify architectural path-dependency and lock-ins that stem from existing work process, digital information and the existing application portfolio respectively. Additionally, we can also anticipate path-dependencies related to the ‘business architecture’ (Versteeg and Bouwman, 2006) or the underlying assumptions about what processes, information and systems are needed in order to pursue specific strategies. Thus, the phenomenon of path-dependency is not only related to the technical

aspects of architecture *per se*, but should be understood as a socio-technical construct operating on different levels of an enterprise architecture.

<i>Enterprise architecture layer</i>	<i>Type and source of path-dependency</i>	<i>Relevant reference</i>
Business architecture	Lack of learning and trapped in existing architectural knowledge making it difficult to proceed to different types of architectures.	Fagerberg (2005); Henderson and Clark (1990); Robey and Lyytinen (1999).
Process architecture	Interdependencies between existing work processes and local Information Systems make it challenging and costly to re-organize the process architecture of a company.	Hepsø et al. (2009).
Information architecture	Digital content in databases or structures (e.g. file formats) in various Information Systems can comprise severe path-dependencies.	Rolland (2000); Shapiro and Varian (1999).
Application architecture	Interdependencies between different Information Systems make modifications or substitutions of one particular Information System challenging.	Ciborra (2000).

Table 1. Path-dependency is likely to occur at different levels of an enterprise architecture.

The emphasis on path-dependencies does not rule out strategic agency in dealing with complex technologies (e.g. Garud et al., 2010; Meyer and Schubert, 2007). Enterprise architects can obviously also – at least over time – based on new blueprints as working implementations (Smolander et al., 2008) break out of lock-ins and establish new paths. In this respect, we see enterprise architecture paths as both evolutionary emergent and strategic deliberate – at least before a certain path leads to lock-in (Meyer and Schubert, 2007).

3 Case and Methodology

3.1 Case description: Bergen Drilling

Bergen Drilling is a privately held Norway-based oilfield service company located in nine different countries globally. Bergen Drilling produces advanced drilling equipment and supports oil companies in installing and operating these on oil wells globally. The company has a history dating back to the 1980s growing out of a cluster of Norwegian oil service companies supplying the government owned Statoil. Since then, Bergen Drilling has gone through various splits and mergers, and was in 2012 bought and split into three different companies. Bergen Drilling, has at the time of writing 300 employees most of whom are located in the larger offices in Norway, Houston, Baku and Perth.

In terms of enterprise architecture Bergen Drilling has a portfolio of different types of integrated and semi-integrated Information Systems, different work processes, important stores of digital content as well as global IT services and infrastructure.

In recent years, the enterprise architecture of Bergen Drilling has been attempted to be radically transformed according to changing business needs as well as an attempt at making it more modularized and standardized across at least three times. Firstly, as part of a strategy for increased global growth and collaboration across locations, a new Information System based on Microsoft SharePoint 2007 software platform was developed in 2009. This involved not only technical implementation and addition

of one more component in the enterprise architecture but a radical transformation of the architecture, as the main plan was to implement a global document management system on SharePoint. Additionally, it was planned to partly substitute and change the role of an advanced CAD tool SolidWorks used by engineers for designing new drilling equipment and the associated PDM-system which was the main archive for technical drawings and documents.

Secondly, prior to the split in 2012, Bergen Drilling had a fairly large IT-department of 30 IT-experts responsible for operating the IT-infrastructure as well as maintaining and developing specialized Information Systems for oil and gas equipment maintenance. However, Bergen Drilling was sold without the IT-department and was then forced to outsource its IT-operations. In 2012 a total of 26 different IT-systems that was therefore attempted transplanted from the old company into a new network domain. Thirdly, after five years of using SharePoint 2007/2010 it was decided to migrate to version 2013.

3.2 Research method

The research reported in this paper is a result of a longitudinal case study of Bergen Drilling. The case of Bergen Drilling was chosen among several alternatives because it was expected to be particularly suited for shedding new light on the complexity of evolving enterprise IT architecture in a global organization. Since Bergen Drilling had been part of a larger corporation with an aligned enterprise IT architecture, it was an opportunity to study the transition towards a smaller company, and then next, a transition towards an increasingly global and complex business. In this respect, the particular choice of the case study as an approach to study enterprise IT architecture evolution and the selection of Bergen Drilling was done with an intention to empirically describe as well as theoretically explain a relatively under-research area. As underlined by Eisenhardt and Graebner (2007), this makes the particular case and approach well suited for generating new theory. Flyvbjerg (2006: p.13) denotes that “atypical or extreme cases often reveal more information because they activate more actors and more mechanisms in the situation studied”. We argue, that this is exactly the case with Bergen Drilling.

The case study of Bergen Drilling is longitudinal both in terms of historically tracing the process of an evolving enterprise IT architecture over a period of 7 years, as well as following the process closely over the last 1,5 years. At the time of writing, we are still doing fieldwork in the company. An important perspective in our research has been to conduct especially longitudinal studies, as the phenomena being studied requires this in order to empirically illuminate shifting challenges and conditions – and theorize the interaction between content and context over time (Pettigrew et al., 2001). In general, there is also an increasing reflection around the need for studying IS over longer periods of time as IS can have different organizational consequences over time (Williams and Pollock, 2012).

Data collection and analysis was conducted in an iterative manner as suggested by many others (e.g. Eisenhardt and Graebner, 2007; Miles and Huberman, 1994) relating to qualitative inquiries. The data collection started off with an initial round of interviews and visits during the autumn of 2013. Interviews were both in-depth qualitative interviews based on an interview guide, as well as more informal talks. Interviews of users of the main IS used, such as PDM and SharePoint, were interviewed at their normal workplace while actually using the systems. Most interviews were tape recorded, and when not, this was based on the researchers own expectations that the interview would risk to give less value and nuances because of the informant’s awareness that tape-recording was done (Walsham, 2006). At the time of writing, a total of 16 formal in-depth interviews have been conducted which has transformed to approximately 150 pages of transcribed material. The informants interviewed could be classified as general managers, members of top management, project engineers, system developers, IT architects, Quality and safety managers, and project managers for operations. The analysis was conducted by organizing all documents and transcribed interviews in the hyperRESEARCH tool. The material was coded according to concepts drawn both from current literature on enterprise architecture and information infrastructure evolution (Hanseth and Lyytinen, 2010). As the analysis progressed, new categories and concepts were added to the initial codes, and some of the original codes were dis-

missed. In this way, the analysis tended to become a mixture of both deductive and inductive reasoning where themes, concepts and patterns emerged over time as more data was collected and analysed in an iterative manner. Thus, as pointed out by Eisenhardt and Graebner (2007: p. 25), “The theory-building process occurs via recursive cycling among the case data, emerging theory, and later, extant literature”.

4 Themes Illustrating Attempts at Radically Changing the Existing Enterprise Architecture

4.1 Theme 1: implementing global document management

In 2009 Bergen Drilling recognized the need for improving the quality of document management and increased standardization across different locations and practices. The main strategy for realizing this was to come up with a comprehensive taxonomy for documents in customer projects and to implement Microsoft SharePoint as a common system for document management. However, this shift required a relatively radical change in the enterprise architecture that became far more complex than anticipated.

First, this strategy required that the existing PDM-system used by engineers for managing technical drawings and documents regarding a product¹ to be replaced by Microsoft SharePoint 2007/2010. However, this turned out more complicated than first envisioned:

“[We] tried to collect all documents in SharePoint – that was our aim. Then, we realized that SharePoint did not tackle SolidWorks-drawings. So the drawings had to continue to be stored in PDM. And next, we had a big controversy around whether the technical documentation that we get from our suppliers should still remain in PDM or not. We lost that battle too. So this documentation is also still in PDM. “ (Manager, DCC)

Thus, not only is the PDM-system path-dependent because of its specific CAD file formats that are not compatible with the current configuration of SharePoint, but also because the existing work processes of engineers required that all technical documentation regarding a project should be categorized and stored in PDM. Engineers pointed out that designing new products and configurations need to always have an overview of related testing procedures and installation procedures. In the end of 2010, they ended up with a “hybrid solution” for document management combining PDM and SharePoint:

“Nobody could foresee that we would need this more comprehensive solution when we started. Obviously we were still thinking along with the folder-structure as the only way of organizing documents. So we ended up with developing a quasi-solution” (Manager and former product owner)

A different strategy attempted on a later stage in 2012-13 was to provide a plug-in for PDM in SharePoint 2010 in order to make it possible to access PDM documents through SharePoint. This solution was eventually turned down because it was expected to become too expensive.

Second, the document management provided by the PDM also involved verification of documents and various reports exchanged with external suppliers and customers. This exchange was partly automated in the PDM-system through sending out emails and for converting specific documents to pdf-format so that they could be sent off to suppliers and customers. As this is not trivial to implement in SharePoint, communication with external actors were partly handled manually by the Document Control Centre.

¹ Bergen Drilling offer several standardized products used on offshore rigs in order to improve safety and efficiency in the drilling of wells. Most projects involve customizing these products to the customer’s specific requirements. This involves small – but crucial differences in various documents and drawings used by the customer for installing, testing and operating the equipment.

In this way the new information architecture and policies for managing documents are somewhat implemented. However, in contrast to the vision, the path-dependencies in the existing information architecture requires lots of manual extra-work in order to comply with the policies defined. Furthermore, the interweaving of the previous information architecture and the PDM-system – that is, lack of modularization, give a strong path-dependency prohibiting a clean implementation of new enterprise architecture. Another example of this is how individuals tend to update excel sheets for keeping track of the status of different documents and how project managers upload a pdf-version of the SolidWorks drawing in SharePoint. Additionally, Microsoft SharePoint comes with a specific architecture that has to combine on all layers of the enterprise architecture.

4.2 Theme 2: re-establishing IT-services and infrastructure

After splitting the original company in three different parts in 2012, the part that at the time of writing comprises Bergen Drilling had to establish new IT services and infrastructure on a separate network domain. The challenge here was first and foremost that the IT-department was left behind in a different company, so Bergen Drilling was more or less forced to outsource IT operations and competence. Consequently an IT service company, here referred to as the ITCompany got the deal and initiated the process of transferring IT services and infrastructure to a new network domain:

”This was a project run by the ITCompany. However, they had a hard time figuring out what to transfer (*to the new domain*). And no one from Bergen Drilling was involved” (IT Director)

The complexity of the existing IT architecture the ITCompany found it a more of a daunting task than they had expected. A Senior Architect that worked in both companies also confirms this. He explains:

“[You] move a lot of garbage with you because it is always something there that you need. But there is usually not possible to determine exactly what we need and what we can leave behind... [...] In this case, it was so complex and a lot of things had dependencies so it became hard to manage. For example, if I move this service what would happen? Nobody could answer that question. We had some blueprints and I think these terrified people so much that no one dared to do anything radical.” (Senior Architect)

The architect interviewed had detailed knowledge about the ‘old’ IT architecture and refers back to his experiences and with doing major changes. The main problem here is complexity and even with fairly detailed blueprints architects are hesitating to conduct radical changes. In other words, the current IT architecture with numerous interdependencies between Information Systems had path-dependency effects.

A further aspect of re-establishing the IT services and infrastructure for the new organization was that outsourcing the project and operations to a third party was not unproblematic. This was because of the long-built competence and detailed architectural knowledge of the internal developers and architects. In this sense, there is a path-dependency related to knowledge about the IT architecture as well as the architecture in itself.

However, in order to deal with such path-dependencies, the architecting practices do not amount to blindly following blueprints, but rather to conduct small experiments in order to get hunches about what the consequences of a certain change would be:

“The consequence is that you have 10 engineers who cannot do their work for about a week. So this is not something you want to do. I would rather try with one user and PC and then see what is happening. Does it still work? Can I change this and this? But, then there are other aspects also. There is something called Managed Client that I still do not fully understand. This is an application used by the ITCompany.” (Senior Architect)

4.3 Theme 3: migrating to new SharePoint versions

A third theme that implied a relatively comprehensive change in the current enterprise IT architecture was the introduction of Microsoft SharePoint software platform in 2009. After initially configuring

and implementing SharePoint 2007 across the organization in late 2009, Bergen Drilling decided to migrate to a new version of the software platform. Version 2010 of SharePoint had obvious advantages with enhanced search facilities based on Microsoft's acquisition of the FAST search engine as well as improved ways of managing meta-data. On the other hand, this required migrating from SharePoint 2007 to SharePoint 2010 – not a simple task at all:

“I did a migration. First, I installed a test version of SharePoint 2010 to just take a look at it. When, I found a tutorial on the Internet for how to upgrade from 2007 to 2010. A late Friday night I started the procedure and copied the SQL database from 2007 to 2010 SQL database, among other things – it was a long shot. Eventually, I found new versions of nearly all the web parts [software modules] we were using. However, there was two web parts that could not be substituted, and those we just had to scrap in order to proceed with the migration. Then, we discovered a rather funny problem when I first launched the new version 2010. Suddenly the main page got all white. In the end we found out that this was because Microsoft had substituted content types (*definitions describing what content to be shown in different containers on a web page*) on the main page, so that the page specified in the 2007 version did not exist in the 2010 version. So then I had to develop the front page all over again. But in the meantime we hacked a version partly on 2007 and partly on 2010” (Developer)

Although the SharePoint software platform was a fairly modularized component of the enterprise IT architecture, it had path-dependencies. As noted by the developer above, he had trouble in getting rid of all parts of the old platform and actually ended up continuing with implementing a part of the solution on version 2007 and different parts on 2010. This went on for practically a year. Moreover, the switch also required changes in the AD.

In 2014 it was decided to migrate again. This time the plan was to switch from SharePoint 2010 to SharePoint 2013 and also at the same time improve document management that still did not work as expected. Now the situation was different since SharePoint had been used for five years across the organization. The platform had also been extended with additional customized applications. Consequently, this new situation also involves path-dependencies.

Architecting requires mobilizing different stakeholders and establishing a ‘common enough’ solution. Finding a way of making a smooth transition often involves keeping the previous versions for some parts while substituting it with the new version for other parts. Cannot make new blueprints except from the obvious – before stakeholders agree.

5 Analysis: Ambidextrous Enterprise Architecting

In order to analyse the different architectural path-dependencies over time, and how they evolve we now synthesize the scenarios above. What we see is a pattern where all three themes over a time period of 5 years are fraught with challenges of path-dependencies. Furthermore, nearly all themes attempt changes that are constrained by interweaving path-dependencies across different architectural layers and components. In this sense, path-dependencies of this type seems to be *architectural* in that they involve more than one component, one type of content and cut across several work processes.

In the case it becomes evident that a lack of modularization and standardization increases path-dependencies, and thus in turn make it even more difficult to modularize the architecture. The introduction of the SharePoint software platform is especially interesting in this regard. At the one hand, SharePoint increases path-dependencies because it for example requires a certain structure of AD and develops path-dependencies as content is uploaded. On the other hand, however, SharePoint also provides a more modularized architecture. The applications built on top of SharePoint are seldom a problem.

Another pattern is the tendency that the complexity of the enterprise architecture and the architectural path-dependencies related to information and process architecture seems to increase over time. More-

over, the path-dependencies related to the basic IT services and infrastructure when at last established throughout the global organization on one common domain, seem to be reduced.

A characteristic is that although new plans and visions are made for describing a future state of the enterprise architecture, *enterprise architecting in practice* is more concerned with solving issues of path-dependency. Hence, we argue that enterprise architecting need to focus just as much on the historical legacy aspects of the different layers of the architecture as on blueprints for representing a future state.

<i>Architectural layer</i>	<i>Theme 1(2009-2012) path-dependencies</i>	<i>Theme 2 (2012-14) path-dependencies</i>	<i>Theme 3 (2010 & 2014) path-dependencies</i>
<i>Work processes</i>	Path-dependencies related to existing work practices and competence of engineers and project managers.	No direct path-dependencies found.	Path-dependencies related to the work practices of categorizing documents. Path-dependencies related to existing work practices and competence of engineers and project managers.
<i>Digital content</i>	The large archive of existing digital drawings and project information stored in PDM and file servers.	Path-dependencies related to information about users in the AD and the HR-system.	Increasing path-dependencies related to the information architecture implemented in SharePoint 2010 platform. And still path-dependencies related to digital content in PDM and file servers.
<i>Information Systems</i>	The PDM-system has path-dependencies.	Path-dependencies related to the existing AD, SharePoint and HR-system and various integrations.	Path-dependencies related to the existing AD, SharePoint and HR-system and various integrations.
<i>IT services and infrastructure</i>	Existing global IT-infrastructure have path-dependencies	Path-dependencies related to the network domain and basic services.	No direct path-dependencies found.

Table 2. Summary of different architectural path-dependencies across themes as the enterprise architecture is transformed over time.

Simultaneously, different actors in Bergen Drilling also perform various intentional acts in order to circumvent the inertia of path-dependencies and in order to bet on future needs of the organization. Henceforth, although architectural path-dependencies are shaping the enterprise architecture over time, it is not a deterministic force as Bergen Drilling finds clever ways – not in radically breaking away from them, but in partly working around their limitations (Table 4 below). In this way, one can argue that enterprise architecture evolution is a highly ambidextrous activity involving both the inertia of existing architecture and visions of the future.

<i>Architectural layer</i>	<i>Theme 1(2009-2012): hacking path dependencies and betting on the future</i>	<i>Theme 2 (2012-14): hacking path dependencies and betting on the future</i>	<i>Theme 3 (2010 & 2014): hacking path dependencies and betting on the future</i>
<i>Work processes</i>	Establishing a Document Control Centre and changing document routines. A strategic initiative. Manages over time to curb the path-dependencies of PDM and DocuShare.	Not applicable.	Developers bet on what work processes would be helpful for various users of SharePoint. Users ‘hack’ the solution and develop their own workflows.
<i>Digital content</i>	Attempting to reduce the complexity in existing taxonomy for documents. Users ‘hacking’ exiting taxonomies by tagging documents.	Architects must bet on what information is needed in the new solutions for the new company.	Developers bet on what information architecture would be needed for handling future information needs.
<i>Information Systems</i>	Adopting Microsoft SharePoint 2010 and a new taxonomy. Involves betting on what kind of information architecture and work processes are needed in the future.	Architects find smart ways of curbing the limitations of the existing architecture of information systems.	Architects find smart ways of connecting SharePoint to existing HR-system.
<i>IT services and infrastructure</i>	Develops new integrations between SharePoint, Active Directory, and the HR-system. Involves betting on what integrations are needed.	Architects bet on future needs by establishing new common services across the global organization. Architects experiments to uncover the consequences of changing the architecture.	Architects struggle to configure SharePoint to fit with global IT infrastructure.

Table 3. Summary of different acts of hacking existing path-dependencies and betting on future architectural needs across themes and layers.

6 Discussion

This paper aims at increasing the understanding of enterprise architecture from a process perspective. In so doing, we have argued that enterprise architecting in organizations is constituted through a combination of an emergent process with path-dependencies largely influencing the process, as well as more intended practices and projects.

Firstly, this paper contributes empirically by illuminating how an enterprise architecture emerges in three major change processes over a time period of nearly six years. In all three themes: establishing company-wide document management, substituting underlying IT infrastructure, and migrating to new versions of the SharePoint software platform, existing solutions greatly influenced the transition towards a ‘target’ architecture. This made the initial blueprints and visions more challenging and sometimes impossible to implement. For example, the initial idea to have SharePoint as the only IS for document management was not possible due to existing PDM-system and complexity of existing work practices. This is an issue that has been given much attention in current research on enterprise architecture. According to their thorough review of the current literature, Simon et al. (2013: p. 16) find that the issue of transition “receive only minor attention”. Interestingly, transition is not an issue the

authors discuss any further in their paper. As for the more technically oriented literature on software architecture, our findings partly correspond to Jansen and Bosch (2005) perspective on ‘architectural design decisions’. Here the authors identify that ‘obsolete design decisions are not removed’ because of the unexpected consequences of removing them as a major problem. This is exactly the kind of complexity we identified in the Bergen Drilling case, where architects were reluctant to radically change the architectures according to the visions because of unknown consequences. However, in our study they usually also find a way of working around the issues through small hacks and experiments, and thereby manages to slowly cultivate the architecture in an incremental step-by-step fashion towards a different direction. On the other hand, the architects also tend to ‘bet on’ what the future needs would be. In this sense, we find *enterprise architecting to be ambidextrous: it needs both an attention to the past and the future at the same time.*

As evident in current literature, enterprise architecture tends to be split into different layers representing different domains and/or granularities of the architecture (e.g. Bruls et al., 2010; Simon et al., 2013). In practice, however, as shown in the case study, attempting to treat the enterprise architecture as consisting of self-contained layers, is indeed highly problematic. Blindness for the interdependencies between different layers can potentially have catastrophic consequences for the architecture and hence the organization as a whole. An archetype of this is how different versions of the SharePoint software platforms have features that make slightly different information architectures preferable. Hence, one of the most challenging issues in enterprise architecture is the unexpected dependency across different layers. This is, at least, in the case of Bergen Drilling hard to avoid due to the reliance upon Microsoft SharePoint that comes with its own architectural logic.

Secondly, the paper contributes by suggesting some constructs for understanding enterprise architecting as process and the challenges involved in implementing a new enterprise architecture. The influence of the existing enterprise architecture across multiple layers can in some instances be referred to as *architectural path-dependencies*. Architectural path-dependency implies that the existing architecture (or layer) has developed inertia that makes it economically, technically and organizationally challenging to choose a different path. Arguably, this is an important insight for understanding the implementation process of enterprise architecture. In the literature, Ross et al. (2006) propose a stage model for architecture maturity in order to progress across four different stages towards ‘business modularity’. In their model, Ross et al. (2006) argue that different competence and learning is important at different stages so that skipping a stage is not recommended. Based on our study, we agree on this point; however, we also expect transitions from one stage to the next to be hampered by architectural path-dependencies and lock-in. Skipping a stage is likely to involve even stronger path-dependencies and lock-ins. Moreover, Ross et al. (2006) expect a larger amount of enterprise systems and shared data in the latter stages. Our study shows that it is exactly this transition that involves architectural path-dependencies that are hard to turn around and strategically select a different path. The attempts in Bergen Drilling to establish common document manages and to migrate from one SharePoint version to another illustrates how such transitions are packed with architectural path-dependencies both within and across layers of the enterprise architecture.

On the other hand, enterprise architectures are of course not entirely determined by architectural path-dependencies (cf. Meyer and Schubert, 2007). From a process perspective, we recognize how enterprise architecture is not only a static blueprint or a vision for future ‘target’ architecture, but *continuous conducts* by architects, developers, and managers. Hence, we propose the term *enterprise architecting* to denote how architectures are intentionally cultivated over time through small incremental steps. Enterprise architecting comes across as both directed towards the *past* – solving the problems of architectural path-dependencies and lock-ins and also as directed towards the *future*. Enterprise architecting directed towards the past is often concerned with mitigating the unwanted effects of path-dependencies and to adjust the architecture to align with ‘new’ business needs or technologies. In this respect, Smolander et al. (2008) note that one category of architectural work is oriented towards past decisions, and describes this as ‘architecture as literature’. Likewise, in the case of Bergen Drilling there blueprints of existing solutions are important for understanding the past decisions and present

architecture. Secondly, and also mentioned by Smolander et al. (2008) enterprise architecting is also directed towards the future. Predominantly, however, in the case of Bergen Drilling this is done through having different sketches of how the existing architecture looks like, and from there the architects are in a sense ‘betting on’ what kind of architectural solutions a future organization would require. Hence, the architects seem to work with an understanding that a radical break with the present situation is somewhat unrealistic – and in some instances considered involving high risk. On the other hand, though, the case also reveals that strategic initiatives *can* curb the inertia of architectural path-dependencies. One example of this from the Bergen Drilling case is the top-down initiative to establish a Document Control Centre. In this case, Bergen Drilling managed to increase business modularization (e.g. Ross et al., 2006), and thus establish not only more flexible enterprise architecture for future changes, but also one that combined different aspects of the existing architecture. In this way, more flexible and loosely coupled architectures are not necessarily implemented exclusively through sophisticated technologies like SOA (Ren and Lyytinen, 2008), but also by new organizational units and work processes.

Our analysis also has potential implications for furthering the conceptualizing how enterprise architecture evolves. Based on their comprehensive review, Simon and colleagues (2013: p. 20) note that studies of enterprise architecture need to attain a closer “integration with related research fields”. In this paper we have explicitly drawn on IS literature concerning information infrastructures (e.g. Hanseth and Lyytinen, 2010; Hepsø et al., 2009), software architecture (e.g. Smolander et al., 2008), network economics (e.g. Arthur 1989; David, 1985), as well as science and technology studies (e.g. Meyer and Schubert, 2007). Arguably, this is helpful in illuminating the many different aspects of the enterprise architecture phenomena – without necessarily dismissing many existing studies focusing on frameworks, representations and meta-representations. However, we do argue that current literature needs to be further extended with new and refreshing perspectives outside the ‘traditional’ paths. Furthermore, we argue that a process perspective is a potential way of closing the gap between enterprise architecting practice and theory. A process perspective takes the specific context and temporal aspect of enterprise architecting into account. As seen in the Bergen Drilling the specific context in terms of splitting the company in smaller units, developing more advanced products needing more comprehensive document management and changes in the SharePoint architecture highly influenced the enterprise architecture of the company, and the practical solutions also need to deal with these issues. In this sense, a process perspective can be seen as a way to ‘innovate mindfully’ (Swanson and Ramillier, 2004) with enterprise architecture.

7 Conclusion

In this paper we have investigated enterprise architecture from a process perspective. Taking into account that evolvement of architecture is about the transition from a given state to a desired end state, we argue that enterprise architecting needs to focus just as much on the historical legacy aspects of the different layers of the architecture as on blueprints for representing a future state. The main contribution is given through an explorative case study and analysis of Bergen Drilling's enterprise architecting in a global business. The case study reveals that enterprise architecting is also about fixing issues with transitions from the past in terms of legacy architecture as well as betting on the future needs of architecture. Secondly, by drawing upon the theoretical framing from path dependency, and suggesting the term architectural path dependency and enterprise architecting, we provide novel insight into how decisions of the past form the decisions of the future. The tendency is that the complexity of the enterprise architecture and the architectural path-dependencies related to information and process architecture seems to increase over time. Path-dependencies provide a link related to the IT services and infrastructure throughout the global organization, and thereby highlight the connection between previous and future IT architectures of the company. Further research should aim at further investigating different choices and kinds of path-dependencies in enterprise architecture evolution.

References

- Aanestad, M. and Jensen, T.B. (2011). Building nation-wide information infrastructures in healthcare through modular implementation strategies, *Journal of Strategic Information Systems* 20(2), 161-176.
- Arthur, W. B. (1989). Competing technologies, increasing returns, and lock-in by historical events. *The economic journal*, 116-131.
- Baragry, J., & Reed, K. (2001). Why we need a different view of software architecture. In *Software Architecture, 2001. Proceedings. Working IEEE/IFIP Conference on* (pp. 125-134). IEEE.
- Bruls, W. A., van Steenberg, M., Foorthuis, R. M., Bos, R., & Brinkkemper, S. (2010). Domain architectures as an instrument to refine enterprise architecture. *Communications of the Association for Information Systems*, 27(1), 27.
- Bosch, J. (2004). Software architecture: The next step. In *Software architecture*. Berlin: Springer, 1994-199.
- Bouwman, H., van Houtum, H., Janssen, M., & Versteeg, G. (2011). Business architectures in the public sector: experiences from practice. *Communications of the Association for Information Systems* 29(1), 411-426.
- Bruls, W. A., van Steenberg, M., Foorthuis, R. M., Bos, R., & Brinkkemper, S. (2010). Domain architectures as an instrument to refine enterprise architecture. *Communications of the Association for Information Systems* 27(1), 517-540.
- Chan, Y. E. (2002). Why haven't we mastered alignment? The importance of the informal organization structure. *MIS Quarterly executive*, 1(2), 97-112.
- Chung, L., and Subramanian, N. (2007). Bridging the gap between enterprise architectures and software architectures. *Science of Computer Programming*, 66(1), 1-3.
- Ciborra, C. and associates. (2000) *From Control to Drift – The dynamics of corporate information infrastructures*, Oxford University Press: Oxford, England.
- Concha, D., Espadas, J., Romero, D., and Molina, A. (2010). The e-HUB evolution: From a Custom Software Architecture to a Software-as-a-Service implementation. *Computers in Industry* 61(2), 145-151.
- Christensen, H. B., Hansen, K. M., Kyng, M., and Manikas, K. (2014). Analysis and design of software ecosystem architectures—Towards the 4S telemedicine ecosystem. *Information and Software Technology*, 56, 1476–1492.
- David, P. A. (1985). Clio and the Economics of QWERTY. *The American economic review*, 75(2), 332-337.
- Eisenhardt, K. M., & Graebner, M. E. (2007). Theory building from cases: opportunities and challenges. *Academy of management journal* 50(1), 25-32.
- Fagerberg, J., Mowery, D. C., & Nelson, R. R. (Eds.). (2005) *The Oxford handbook of innovation*. Oxford: Oxford University Press.
- Flyvbjerg, B. (2006) Five misunderstandings about case-study research. *Qualitative inquiry* 12 (2), 219-245.
- Gacek, C., Abd-Allah, A., Clark, B., & Boehm, B. (1995). On the definition of software system architecture. In *Proceedings of the First International Workshop on Architectures for Software Systems*, Seattle, Wa., 85-94.
- Garud, R., Kumaraswamy, A., & Karnøe, P. (2010). Path dependence or path creation?. *Journal of Management Studies* 47(4), 760-774.
- Henderson, R. M., & Clark, K. B. (1990). Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. *Administrative science quarterly* 35(1), 9-30.
- Hanseth, O and Lyytinen, K. (2010) Design theory for dynamic complexity in information infrastructures: the case of building Internet, *Journal of Information Technology* 25(1), 1–19.

- Hepsø, V., Monteiro, E. and Rolland, K. (2009) Ecologies of e-infrastructures *Journal of the Association for Information Systems* 10(5), 430–446.
- Jansen, A., & Bosch, J. (2005). Software architecture as a set of architectural design decisions. In *Software Architecture, 2005. WICSA 2005. 5th Working IEEE/IFIP Conference on. IEEE*. Chicago, 109-120.
- Lagerström, R., Johnson, P., and Höök, D. (2010). Architecture analysis of enterprise systems modifiability—models, analysis, and validation. *Journal of Systems and Software* 83(8), 1387-1403.
- Lizcano, D., Alonso, F., Soriano, J., and López, G. (2014). A component-and connector-based approach for end-user composite web applications development. *Journal of Systems and Software* 108–128.
- Lyytinen, K., & Robey, D. (1999). Learning failure in information systems development. *Information Systems Journal* 9(2), 85-101.
- McArthur, K., Saiedian, H., and Zand, M. (2002). An evaluation of the impact of component-based architectures on software reusability. *Information and Software Technology*,44(6), 351-359.
- Meyer, U., & Schubert, C. (2007). Integrating path dependency and path creation in a general understanding of path constitution. The role of agency and institutions in the stabilisation of technological innovations. *Science, Technology & Innovation Studies*, 3(1), 23-44.
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook*. London: Sage Publishing.
- Pettigrew, A. M., Woodman, R. W., & Cameron, K. S. (2001). Studying organizational change and development: Challenges for future research. *Academy of management journal*, 44(4), 697-713.
- Ren, M., & Lyytinen, K. J. (2008). Building enterprise architecture agility and sustenance with SOA. *Communications of the Association for Information Systems* 22(1), 75-86.
- Rolland, K. H. (2000) Challenging the Installed Base: Deploying a Large Scale IS in a Global Organization. In: Hansen, HR., Bichler, M and Mahrer, H (eds.) *Proceedings of the Eight European Conference of Information Systems*, Vienna, 583-590.
- Ross, J.W., Weill, P. & Robertson, D.C. (2006). *Enterprise Architecture as Strategy: Creating a Foundation for Business Execution*. Boston, Massachusetts: Harvard Business School Press.
- Šaša, A., and Krisper, M. (2011). Enterprise architecture patterns for business process support analysis. *Journal of Systems and Software*, 84(9), 1480-1506.
- Shapiro, C., & Varian, H. R. (2013). *Information rules: a strategic guide to the network economy*. Boston, Massachusetts: Harvard Business Press.
- Simon, D., Fischbach, K., & Schoder, D. (2013). An exploration of enterprise architecture research. *Communications of the Association for Information Systems*, 32(1), 1-77.
- Smolander, K., Rossi, M., & Purao, S. (2005). Going Beyond the Blueprint: Unravelling the Complex Reality of Software Architectures. *ECIS 2005 Proceedings*, 95.
- Smolander, K., Rossi, M., and Purao, S. (2008). Software architectures: Blueprint, literature, language or decision&quest. *European Journal of Information Systems* 17(6), 575-588.
- Swanson, E. B., & Ramiller, N. C. (2004). Innovating mindfully with information technology. *MIS quarterly*, 553-583.
- Tamm, T., Seddon, P. B., Shanks, G., & Reynolds, P. (2011). How does enterprise architecture add value to organisations. *Communications of the Association for Information Systems* 28(1), 141-168.
- Tilson, D., Lyytinen, K., & Sørensen, C. (2010). Research commentary-digital infrastructures: the missing IS research agenda. *Information Systems Research* 21(4), 748-759.
- Tiwana, A., & Konsynski, B. (2010). Complementarities between organizational IT architecture and governance structure. *Information Systems Research* 21(2), 288-304.
- Velasco Elizondo, P., and Lau, K. K. (2010). A catalogue of component connectors to support development with reuse. *Journal of Systems and Software* 83(7), 1165-1178.
- Versteeg, G., & Bouwman, H. (2006). Business architecture: A new paradigm to relate business strategy to ICT. *Information Systems Frontiers* 8(2), 91-102.

- Walsham, G. (2006). Doing interpretive research. *European journal of information systems* 15(3), 320-330.
- Williams, R., & Pollock, N. (2012). Research Commentary-Moving Beyond the Single Site Implementation Study: How (and Why) We Should Study the Biography of Packaged Enterprise Solutions. *Information Systems Research* 23(1), 1-22.
- Yoo, Y., Lyytinen, K., & Yang, H. (2005). The role of standards in innovation and diffusion of broadband mobile services: the case of South Korea. *The Journal of Strategic Information Systems*, 14(3), 323-353.
- Zachman, J. A. (1987). A framework for information systems architecture. *IBM systems journal* 26(3), 276-292.
- Zhu, K., Kraemer, K. L., Gurbaxani, V., & Xu, S. X. (2006). Migration to open-standard interorganizational systems: network effects, switching costs, and path dependency. *MIS Quarterly* 30 (Special Issue on Standard Making), 515-539.