Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 16 (2014) 684 – 693

Procedia Technology

CENTERIS 2014 - Conference on ENTERprise Information Systems / ProjMAN 2014 - International Conference on Project MANagement / HCIST 2014 - International Conference on Health and Social Care Information Systems and Technologies

Service-oriented architecture projects in practice: A study of a shared document service implementation

Eli Hustad[a]*, Christer de Lange[b]

[a]*University of Agder, Gimlemoen 25, 4630 Kristiansand, Norway*
[b]*Norwegian Defence, Kjevik airport, Norway*

Abstract

This paper reports on a case study focusing on service-oriented architecture (SOA) projects in practice. The aim of these projects has been to develop a shared document service supporting different business divisions in a large Norwegian governmental institution. The research question guiding this research is: *How are SOA projects carried out in practice? What are the challenges of developing shared services in service-oriented architecture?* SOA projects are accompanied by a complex socio-technical system development environment. In the present study we followed parallel system development processes and identified several issues associated with competence requirements, distributed coordination principles and control, lack of communication, tuning of parallel projects, and selection of appropriate project management approaches and system development methods. Clearly, there was a difference between undertaking a system development project within one organizational unit (silo) compared to the development of shared services to provide support across a large organization. Findings demonstrate that the complexity of SOA projects was underestimated by the project managers, and ad hoc governance was practiced in terms of control, coordination, and communication. To capture the entire system development context of an SOA project, a holistic approach and mind-set comprising time management and fine tuning of all parallel SD activities is necessary. Organizational maturity to carry out SOA projects is also of significance. The study has implications for SOA adopters in general and for system developers and project managers working in an SOA context in particular.

* Corresponding author. Tel.: +047 38141000; fax: +047 38141001.
E-mail address: eli.hustad@uia.no

doi:10.1016/j.protcy.2014.10.017

Keywords: Service-oriented architecture; SOA projects in practice; SOA governance, SOA implementation, Scrum

1. Introduction

Recently, service-oriented architecture (SOA) has received increased attention as a platform from which to approach challenges related to the development and maintenance of heterogeneous information technology (IT) portfolios [1-3]. Several organizations face architectural challenges arising from complex mixes of different legacy systems, enterprise systems, platforms, and applications. To achieve the amalgamation of these components, point-to-point integration and implementation of middleware solutions (e.g., enterprise application integration [EAI]) has increased [4]. The resources required for the development and maintenance of these compound solutions, however, have become extensive. SOA can meet these challenges, and its module-based architecture and the services it offers promise increased flexibility and reusability [5]. The focus on services makes SOA unique, because it provides transparency across multiple (legacy) applications and data sources that are black boxed. Because the services are defined by open standards, SOA makes available a common pool of IT resources despite the presence of different IT systems, functionalities, language codes, and platforms.

The implementation of SOA, however, requires long-term projects that involve comprehensive organizational changes in terms of new approaches to system development and different IT governance mechanisms, as well as changes in the roles and responsibilities of employees and system developers in particular [1, 5]. SOA should be utilized as a business transformation tool for solving larger business needs, rather than strictly as an IT architectural initiative [3, 5, 6]. Moreover, SOA should be seen as a means to drive organizational strategy that focuses on the alignment of business and technology for agility [1]. So far, SOA research has been mainly technology-oriented, and there is a need to study the socio-technical issues associated with SOA. There is an increasing interest in how to approach SOA governance, and several frameworks and models have been developed by both vendors and academics [2, 7, 8]. However, few empirical studies have focused on SOA projects in practice and the challenges SOA adopters encounter. This study seeks to bridge this gap. SOA projects demand a different system development (SD) approach, and knowledge about how to accomplish SOA governance in practice.

For the purpose of this paper, we go beyond a technical perspective on SOA to view this architectural infrastructure as socio-technical system that affects the enterprise at different levels: business processes, system development practices, and IT governance mechanisms.

We conducted a case study in the IT and Service Department (ITSD) of NORDIC PI (pseudonym) which is a Norwegian governmental institution. The organization began an SOA program in 2008. During the study, we followed system developers, system architects, and project managers across different functions working on specific SOA projects. Our aim was to understand how such projects are carried out and managed in practice. The system developers in a SOA program work across different functions (silos) and seek to develop shared IT services. The following research questions guided our study:

How are SOA projects carried out in practice? What are the challenges of developing shared services in service-oriented architecture?

To explore these issues, we followed SOA projects in practice to identify the specificities of SOA projects and the particular challenges SOA adopters and system developers run into when working in an SOA environment.

The paper is organized as follows. Section 2 conceptualizes SOA, presents some relevant research, and discusses SD challenges in an SOA context. Section 3 introduces the research site and method. Section 4 provides an overview of the SOA program and SOA projects under study, while Section 5 provides a discussion and the implications of our research. In Section 6, we provide some concluding remarks.

2. SOA concepts and related research

SOA definitions. There are several conceptualizations of SOA in the literature. SOA is technically defined as follows: “Service-Oriented Architecture is an IT architecture where data and logic functionality are ‘black boxed’ or encapsulated with only their input and output exposed for others to use" [5]. The concept of SOA also combines business, technology, and IT management perspectives [9]. First, from a business perspective, SOA represents a set of services used to improve the capability of organizations to do business with their customers. Second, from a technological point of view, SOA is characterized by modularity, service reuse, new programming methods based on standards, and tools involving web services. Third, from an IT management perspective, SOA provides a new method for designing IT application portfolios. In addition, the implementation of SOA brings about new challenges related to organizing, IT governance, and change management. The alignment of technology and business is critical; without it, the potential of SOA will not be optimized [10].

Ross [11], who has investigated enterprise architectures, explains how IT architecture develops through four maturity levels over time: application silo, standardized technology, rationalized data, and modular architecture. Modular architecture is enterprise-wide architecture (EA) based on standards and loosely coupled applications, data, and technology components. The advantage of this kind of architecture is that it is possible to quickly adapt solutions and reuse code in SD. SOA typically represents modular architecture. The largest risks of modular architecture are (1) that enterprises introduce modules before they have established rationalized data architecture, and (2) that it lacks governance mechanisms [11].

Hirschheim et al. [12] have developed an SOA maturity model based on a study of ten companies and their experiences with SOA. The maturity model is divided into five different levels to illustrate the organizations' progress regarding effective utilization of SOA. Several of the companies in the survey were still at early maturity levels. To reach a higher level of maturity, the organizations must succeed in applying SOA as an integrator to align business strategy and IT strategy in order to create adaptive architecture.

Baskerville et al. [1] have conducted a comparative case study of the implementation of SOA in the banking sector, concluding that the adoption of SOA may provide greater organizational agility and competitiveness. However, SOA is a means to drive organizational strategy that focuses on the alignment of business and technology for agility. SOA is thus becoming more than IT architecture with services defined by the IT department; it enables business architecture to support a service-oriented enterprise [5].

SOA governance and SD approaches. SOA requires new approaches to system development and governance mechanisms. There are several frameworks presented in the literature focusing on SOA governance principles, and a key insight is that SOA should be centrally controlled by developing centralized governance architecture [5, 13-15]. Erl et al. [16] define SOA governance metaphorically as four pillars that must be stabilized and focused on when managing an SOA program. These pillars are teamwork, education, discipline, and a balanced scope between them. To obtain *a balanced scope,* the different project groups must collaborate and have a common knowledge base regarding how shared services are developed, giving consistency and discipline to the development process. SOA governance principles govern how SD activities are approached by SOA adopters [17, 18].

SOA requires new roles for developing and maintaining SOA applications, which may differ from SD in silo structures. It is important to plan for and tackle interoperability challenges at an early stage. However, due to the complexity of SOA and the nature of the system development tasks, it is beneficial to apply different system development approaches [17]. Building an SOA infrastructure requires a different approach than using the services for composite applications [17]. The former requires a more rigorous system development approach (e.g., waterfall), while the latter would benefit from a more agile and flexible approach. The planning phase requires a holistic mindset and a global perspective to leverage enterprise services across the organization, the analysis requires rigor and discipline, and traditional SD knowledge is important. An SOA context is distributed, and the design phase may be the most critical phase with respect to changes from former SD practices in silos. A high level of coordination is required during implementation and deployment, because many components across functions are difficult to control and must be considered to avoid changes at later stages.

Scrum is the *agile method* that was applied of the system developers in this study. The point of departure for agile methods is that the SD process is an unpredictable, creative, and complicated process that, like incremental SD, does not follow specific steps [19]. The work of the SD team is based on autonomy. Scrum reduces the complexity and increases the flexibility of an SD project by splitting it into several iterative circles to solve tasks gradually as well as by continuously adjusting. The Scrum team consists of system developers and a Scrum master [20]. The latter is responsible for organizing the meetings, but decision making is a democratic team process. Customers attend meetings regularly to present their functionality needs, and these "user stories" and "use cases" serve as the foundation for SD. One of the key principles of Scrum is that written requirement specifications can suffer from inaccuracy and be subject to misunderstanding. Therefore it is better to discuss and communicate directly with the end users [21].

Coordination across different Scrum teams, however, is necessary in larger projects; otherwise the project can easily fail. This is a challenging situation for the Scrum masters, who should organize "Scrum of Scrums" meetings, that is, cross-team interaction events in which members of different Scrum teams participate [20].

3. Research site and method

The site of this study is the IT and Service Department (ITSD) of NORDIC PI (pseudonym) which is a Norwegian governmental institution with approximately 7,000 employees working in different business units. ITSD, which is one of these units, has approximately 900 employees and is responsible for developing, operating, and maintaining Nordic PI's IT portfolio. ITSD has offices in three different locations of Norway.

Nordic PI and ITSD in particular have an ongoing, long-term program that focuses on a transformation and renewal of the IT function. This program is rooted at the highest level of the organization, and the top manager is the leader of the steering committee. The focus of this overarching IT strategy is to create modern and flexible IT solutions, effective integrations, and service-oriented architecture. The unit working on architecture has a mandate to be part of all SD projects and exert an influence on how different SD projects are carried out. The aim is to obtain unitary integrations and standardized execution of applications. These guidelines are intended to be common knowledge for all system developers working at ITSD.

In 2008, NORDIC PI underwent organizational development, which included a review of existing business processes and work routines. This move resulted in comprehensive changes in business processes, establishment of a new organizational structure, changes in collaboration models, and assumption of new tasks for employees. This reorganization project was a means of preparing the organization for the implementation of SOA. NORDIC PI introduced the concept of services and analyzed the value chain to identify the most important business tasks. In 2009, an enterprise architecture project began. The aim of this project is to establish an overarching EA in connection to an IT regulation plan. Along with these changes, ITSD developed a new IT strategy that emphasizes the importance of integrated system architecture based on shared components, reusability, and service orientation. An organizational development (OD) project has also been initiated at ITSD and was implemented in 2011. Different sections were established, one of which involves common components (building blocks) and is responsible for architectural issues related to the integration platform (SOA bus), the new system development platform (JAVA), and other shared components and services. Other sections deal with tax collections and legal issues, infrastructure, and master data. The reorganizing has led to new roles and responsibilities for employees working within ITSD.

NORDIC PI has not completed its program for renewal of the IT function, and there are still several ongoing projects that intend to modernize services and functionalities. The aim of these projects is to gradually simplify the information systems of NORDIC PI by identifying and distinguishing common functionalities in order to develop shared components and services. In this study, four parallel SOA projects and an overbuilding project (document project [DP]) have been studied. The aim of these projects is to develop a shared document service and a shared print solution working from and across different legacy systems (silos). The project is large and complex and involves several stakeholders across NORDIC PI working at different geographical locations of the institution. The project has high priority and is central in the renewal of the IT function.

In addition to 15 initial interviews conducted when the SOA program started in 2008-2009, 10 semi-structured, face-to-face interviews were conducted during 2012 to understand how SOA projects were carried out in practice. The participants had roles as system developers, system architects, and project managers in the projects under investigation. The interviews lasted from one to two hours. Project documents were applied as secondary material to clarify details and contextual information related to the SOA projects and the organization. Data collection and analysis proceeded iteratively, which allowed themes to emerge and to be examined deeply. The analytical process utilized the principles of Miles and Huberman [22], which comprise an interaction between four components of data analysis: data collection, data reduction, data display, and emerging conclusions. The interviews were taped and transcribed, and the interview records were listened to several times to identify voice details (tone, atmosphere, and mood) in the conversations. In addition, the written transcripts were read several times before the text was reduced to arrive at the main content and key themes. Visual displays and mind maps were worked out from the text to support the integration between different interviews and themes. The conclusions were gradually developed and were tested against new interviews and documents.

This study employs a hermeneutic circle approach [23]; each iterative circle of statements and interpretations from the participants represents the parts, while the holistic understanding of the execution of SOA projects in practice emerged gradually when details were put together into a whole.

4. The SOA program and SOA projects at ITSD

This section provides a brief background for the SOA program in NORDIC PI and then describes the projects under investigation.

NORDIC PI has continuously developed IT solutions since 1969. The organization is well known for being an early adopter of new technologies and SD tools. Therefore, a complex IT portfolio with several application silos has been developed in-house. Today, NORDIC PI has 120 to 130 applications that are more or less randomly integrated (spaghetti integrations). These integrations are mainly batch integrations and are relatively difficult to maintain. This has created an infrastructure with low reusability and a complicated structure. ITSD wanted to develop a common integration platform to more easily integrate and execute applications. This work started in 2008 and lasted 18 months, and a framework and guidelines for using the platform were developed. Obtaining efficient integrations, however, involves much more than developing an SOA platform. It is important to establish uniform guidelines for how to handle integration; accordingly, all system developers are trained in the use of the platform, and several one-to-one integrations are now replaced by web services. Recently, ITSD has started the development of common building blocks to offer shared services internally, but also integrations and shared services between NORDIC PI and their stakeholders (e.g., court of justice, chief municipal treasurers). The SOA platform is important to ensure the high security required in NORDIC PI.

In parallel with development of the SOA platform, a new system development platform was also established. This platform is based on JAVA, service-oriented architecture and web-services. The goal of this platform is to establish a standardized SD environment, more effective knowledge sharing, SD and administration of systems.

The document project (DP) began as an initiative from various functional departments with their own functional information systems (legacy systems), which were complex and difficult to maintain. In particular, they wanted to modernize the letter solutions in their respective silos. It turned out that the different functional communities faced the same challenges. Therefore, the different silos started SD projects with the aim of developing a shared document service by utilizing the established SOA platform. They wanted to create a shared component service that offered a common graphical user interface (GUI) for the different functional systems in the value chain. To obtain this, individual components had to be developed for each of the legacy systems. In addition, the aim was to create a GUI for the department responsible for printing (the print environment).

These SD projects were first organized as separate projects with no formal connection. They were initiated from three different silos for developing the building blocks necessary to get the shared document service working: (1) LegDoc1, the tax collectors' accounting and administrative system, (2) LegDoc2, the tax collectors' VAT system taking care of different kinds of fees, (3) LegDoc3, the centralized print environment responsible for large print jobs for public institutions, including NORDIC PI, and (4) SharedDoc4, a project for developing the shared component itself.

Later, DP was established to ensure communication and integration between the different sub-projects.

5. Discussion and Implications

DP is an important signal project for NORDIC PI. It was the first project that implemented a shared service after introduction of the SOA program and the infrastructural building blocks (e.g., integration platform and JAVA system development platform). The project was carried out in a complex IT infrastructure and SD environment, and several challenges had to be tackled during this process. The limited empirical research on SOA governance and methods of controlling SOA projects in practice demonstrates the need for established SOA governance principles and management of those that became an important part of the learning process for the different SD silos involved [24]. The study identified several issues and challenges which increased the complexity of the projects and complicated the accomplishment of the different SD processes. These issues were categorized into (1) internal and external competence, (2) coordination, (3) communication, (4) parallel projects, and (5) project management and SD method. These categories have implications for how to manage SD processes in SOA projects, and are discussed below.

Internal and external competence. In interviews, key participants in DP stated that lack of expertise in the project groups was challenging in the different SD processes. Historically, system developers at ITSD had carried out their work in silos and developed and maintained different functional business systems in different SD environments by using different SD languages. In DP, the SD process should be performed in a service-oriented infrastructure. In addition, the use of a common integration platform (SOA bus) and a system development platform based on JAVA was mandatory. JAVA was an unknown system language for most of the system developers in ITSD. The integration platform and SD platform were recently introduced, and the organizational building blocks (guidelines, best practices) were at the time of DP not institutionalized, and the system developers did not have experience with these guidelines.

The combination of learning a new system language and developing and implementing components in a complex and distributed SD environment became an overwhelming challenge for the system developers, who were used to working in silos. Some of the challenges arose because the SD projects were slightly ahead of the development of guidelines and lessons learned from the platforms. DP was a prestige project, and in response to pressure from the line manager and top management, the team rolled out a beta version of the shared document service that had several errors and minimum functionality. DP was implemented at an immature stage with respect to competence. At the same time, ITSD started to build competence in JAVA and architecture and began certification for the system developers. This is in accordance with Erl's [16] concept of SOA governance, in which competence is one of the crucial pillars.

The external consultants who were hired in key roles—especially in DP—lacked the necessary business understanding of NORDIC PI to establish coordinated SD processes for creating shared services in SOA architecture. Business competencies are especially highlighted in different frameworks for implementing SOA and in the architecture literature [2, 11, 25]. In modular architecture, Ross [11] emphasizes the importance of having internal system architects on the projects to ensure a unified development of the IT infrastructure and to maintain expertise in their own organization. In addition, internal experts know the organization and business areas and the history of existing IT portfolios, making it easier to develop, integrate, and roll out new systems. The NORDIC PI has a strategy for having internal staff in all important roles; however, they need to hire external consultants when the internal capacity is too low. The DP hired several consultants. The lack of business understanding among the external consultants led to delays and key constraints in some cases because the consultants did not have enough knowledge about the business logic, processes, and the SD context at ITSD. This again may have influenced the solutions that were selected. These delays resulted in the need for postponing the final delivery of the DP as a whole at several times during the project period.

Coordination. The DP started originally as four individual projects. The SharedDoc4 was perceived as the hub among the three other projects (LegDoc1, LegDoc2, and LegDoc3); however, a joint steering committee with a necessary mandate to make decisions was missing. Participants in the study point out that they put too much trust in the requirement specifications and the contracts that were established between the services. After the requirement specifications were developed, there was limited coordination between the various projects. As a result, information on the progress of projects and interdependencies between them did not reach the system developers working on the different subprojects. Previous research on SOA guidelines and agile SD has highlighted the importance of coordination and collaboration between project teams [e.g., 26]. In SOA projects, SD of shared services across projects needs to follow the same direction and tempo, in addition to the requirement specification and contracts that were agreed upon. The lack of coordination led to several challenges, such as delays, the need of rework because of misunderstanding or deviation from the requirement specification, poor social climate, and lack of trust across the various projects.

The projects followed an agile SD approach, the Scrum approach, and it became important to coordinate across different Scrum teams to increase the Scrum scalability because of the multisite SD process at ITSD. According to

Cockburn [27] and Vlaanderen et al. [20] best practices would be to arrange daily internal Scrum meetings in addition to "Scrum of Scrums," where one participant from each team meets and coordinates across the different Scrum teams to capture information that needs to flow between projects and synchronize Sprints. This was not practiced at the early stage of the DP. After establishment of the DP as a top project of the subprojects, and after the introduction of a separate steering committee, the system developers improved their collaboration climate and became more satisfied with the coordination, and thus fewer challenges were encountered.

Communication. Good communication habits and an effective infrastructure for communication are important for all larger SD projects [28]; however, SOA projects, in particular, are vulnerable [16]. The DP project was carried out at three different locations, and except for SharedDoc3 (which developed a shared delivery process), the other projects had separate deliveries at each location. Altogether, the organization resembles a multisite organization [27, 28].

Firstly, participants stated that communication had been inadequate, especially among the different projects. Several participants stated that "to be at different locations, complicates our work and communication," when describing the climate of cooperation between the projects. It seems that it was mainly the project managers who communicated across projects. At the SD level, only sporadic contact occurred, and these contacts were mainly based on private initiatives. The projects did not arrange kick-off meetings, and it was difficult to contact colleagues at other locations. In many ways, the SD continued to develop software according to their old routines in their respective silos.

Cockburn [27] describes the "not-invented-here syndrome" in which system developers want to develop their own solutions instead of implementing complete solutions that others have made. If the SD environment was more customized in its design to facilitate the required collaboration between system developers in different silos so they could work together to develop common building blocks, ITSD would probably have saved resources. When the teams changed practice from using written requirement specifications only and started to apply user stories to a larger extent, the SharedDoc4 got a better overview of the different use cases related to the shared service. Thus, to apply an appropriate and customized SD approach was essential here.

Secondly, SharedDoc4 did not include participants from the other subprojects in central meetings in the organization. Moreover, after these meetings, the decisions made were not sufficiently communicated. This again led to information delays for the subprojects; they learned about important frame factors and changes toward the latter stage of the process. Teamwork is one of three pillars of Erl's [16] model for SOA governance. Erl [16] claims that it is urgent to have a balanced cooperation both within and between project teams. In this kind of context, it is essential to achieve confidence between the participants in the project, so everyone is pulling in the same direction. Lack of trust can ultimately lead to colleagues working against each other, and consequently the overall project fails. In SharedDoc4, it seemed that trust and cooperation between the system developers at the two different locations were lacking.

Parallel projects. NORDIC PI has completed several projects in the past years to renew their IT function. These projects have modernized the IT infrastructure and the organization. However, at the same time, complexity has increased, causing challenges for the SOA projects.

Ross [11] and Hirschheim et al. [5] have studied maturity levels in different IT architectures. To increase the maturity, an IT organization needs to have experience and spend time at one maturity level before going to the next. ITSD may have proceeded too fast; for example, they started to develop shared services before they had gained experience regarding how to use the architecture. Lee et al. [29] emphasize the importance of taking into account the available capacity before starting to plan incremental introduction of shared services. NORDIC PI had created a number of technical building blocks in terms of an integration platform (SOA bus) and a system development platform (JAVA platform), but the organizational building blocks encompassing SOA governance guidelines and JAVA competencies were not yet implemented fully before proceeding to the next level. The framework (SOA roadmap) from BEA [2] recommends harvesting services through an incremental SD process and reuse functionality where possible. The reorganization of ITSD created some organizational challenges that were not envisaged. Many employees got new roles and responsibilities. The DP therefore had problems with identifying persons who were responsible for tasks influencing the project. The DP consisted of four interdependent subprojects with individual SD processes going on in parallel. This overall process needed to be synchronized to function effectively. This was not easy in itself, since the shared service at the other end (SharedDoc4) was constantly changing. Moreover, poor communication across the projects and requirement specifications that were not always followed caused several problems to escalate. Rework was needed, which again caused stress and delays.

Project management and SD method. The participants in the study stated that they use Scrum as the SD approach, and they did not take any special precautions in terms of project methodology in the DP. There is little research on leadership and management of specific SOA projects, but the abovementioned frameworks on SOA implementation [2, 25] and SOA governance in particular [16], emphasize the importance of having established SOA governance principles before starting SOA projects and developing shared services. It seems that the complexity of the SOA projects was underestimated by the project managers, and they had far too little focus on control, coordination, and communication. Clearly, there is a difference between undertaking an SD project within a silo compared to the development of shared services to provide support across the organization. Tran et al. [30] state that the complexity of SOA is so extensive that you have to take precautions to capture the entire context of an SD project. In the DP, many challenges could possibly have been avoided if they had taken into account the complexity of the management of the project.

Experiences and lessons learned from the DP should be transferred to new SOA projects in the NORDIC PI. Developing shared services would lead to the involvement of many different stakeholders who are interested in deploying the shared services into their own silo. The number of considerations that must be taken into account in the project will increase, since they need to deploy components in several different systems simultaneously and at the same time coordinate and integrate between them and the SOA bus. This will increase the complexity of the overall project context, and project managers and system developers need to have more control and communication across the organization when decisions are made. The SD approach and project methodology must be chosen and customized with care.

6. Conclusions

This study has followed several SOA projects in practice, and reported on issues and challenges in different parallel SD processes. SOA projects are accompanied by a complex socio-technical system development (SD) environment. Particular challenges were associated with competence requirements, distributed coordination principles and control, lack of communication, use of external consultants in key roles, tuning of parallel projects, and selection of appropriate project management approaches and SD methods.

NORDIC PI has received an increased level of complexity in the architecture as a result of service orientation and new challenges this brings along. One reason for increased complexity level is to implement SOA while silo applications still are in use. The lack of institutionalized guidelines for SOA governance made the organization an early adopter with respect to maturity. The organization is well known, however, for being innovative and trying out new IT solutions at an early stage. Therefore these SOA projects have been valuable experiences for the organization, despite hard learning processes. NORDIC PI has 130 applications in their system portfolio, and introduction of shared services across these diverse applications and architectures are a challenging task.

The DP project involved multiple systems that needed to talk together and share information flow and status information via the integration platform (SOA bus). This caused a complex technical situation and unexpected errors occurred. SOA projects require a holistic approach in terms of organizing the SD processes and the associated project teams involved. Altogether, this socio-technical context is far more complex compared to undertaking an SD project within a silo. The DP project was the first project focusing on developing shared services that cut across silo applications and architectures, and it represents a signal project in the organization. However, many of the participants in the study stated that the project had not been an "ideal" project – it had many challenges, time delays and the budgets were exceeded. The project has been followed with great interest across different departments, and other silo structures are waiting to connect to the shared document service. One of the project managers stated: "The real value of implementing SOA, will come when all systems can utilize common components, and over time the positive effects and gains will increase". According to the framework of BEA (2005), the investments of SOA are high; however, the effectiveness and gains will emerge over time [2].

This study has implications for future research. There are few empirical studies of SOA projects in practice, and more knowledge is needed to understand the complexity of a service-oriented SD context which requires a holistic approach and mindset regarding project management and agile SD methods. The findings indicate that SOA adopters need to develop and practice governance principles that avoid ad-hoc and silo-based SD practices when implementing shared services. In addition, time tune management, coordination of sub-projects, and "Scrum of Scrums" are important for project management and the system developers.

This study has implications for SD practice. Lessons learned from this study can be utilized of other organizations adopting SOA and developing shared services. Our research is exploratory, so it has limitations. However, our results can serve as input for subsequent qualitative studies on SOA projects in practice, and service-oriented SD processes in organizations.

References

[1] R. L. Baskerville, M. Cavallari, K. Hjort-Madsen, J. Pries-Heje, M. Sorrentino, and F. Virili, "The strategic value of SOA: a comparative case study in the banking sector," *International Journal of Information Technology & Management,* vol. 9, pp. 30-53, 2010.

[2] S. BEA. (2005, 30. Jan). *BEA SOA domain model*. Available: http://www.ebizq.net/white_papers/6196.html

[3] C. Legner and R. Heutschi, "SOA Adoption in Practice - Findings from Early SOA," in *Proceedings of the Fifteenth European Conference on Information Systems (ECIS 2007), St. Gallen, Switzerland*, 2007, pp. 1643-1654.

[4] W. Lam, "Investigating success factors in enterprise application integration: a case-driven analysis," *Eur J Inf Syst,* vol. 14, pp. 175-187, 2005.

[5] R. Hirschheim, R. Welke, and A. Schwarz, "Service - Oriented Architecture : Myths , Realities , and a Maturity Model," *MIS Quarterly Executive,* vol. 9, pp. 37-48, 2010.

[6] G. Viering, C. Legner, and F. Ahlemann, "The (Lacking) Business Perspective on SOA–Critical Themes in SOA Research," in *Wirtschaftinformatik Proceedings*, 2009, p. Paper 5.

[7] T. G. J. Schepers, M. E. Iacob, and P. A. T. V. Eck, "A lifecycle approach to SOA governance," in *Proceedings of the 2008 ACM symposium on Applied computing*, Fortaleza, Ceara, Brazil, 2008, pp. 1055-1061.

[8] M. Niemann, J. Eckert, N. Repp, and R. Steinmetz, "Towards a generic governance model for service-oriented architectures," in *Proceedings of the Fourteenth Americas Conference on Information Systems (AMCIS 2008), Toronto, ON, Canada*, 2008.

[9] A. Ordanini and P. Pasini, "Service co-production and value co-creation: The case for a service-oriented architecture (SOA)," *European Management Journal,* vol. 26, pp. 289-297, 2008.

[10] J. Antikainen and S. Pekkola, "Factors influencing the alignment of SOA development with business objectives," in *Proceedings of the 17th European Conference on Information Systems (ECIS 2009)*, Verona, Italy, 2009, pp. 2579-2590.

[11] J. W. Ross, "Creating a strategic architecture competency: Learning in stages," *MIS Quarterly Executive,* vol. 2, pp. 31-43, 2003.

[12] R. Welke, R. Hirschheim, and A. Schwarz, "Service-Oriented Architecture Maturity," *Computer,* vol. 44, pp. 61-67, 2011.

[13] P. Weill, M. Subramani, and M. Broadbent, "Building IT Infrastructure for Strategic Agility," *MIT Sloan Management Review,* vol. 44, pp. 57-65, 2002.

[14] R. Varadan, K. Channabasavaiah, S. Simpson, K. Holley, and A. Allam, "Increasing business flexibility and SOA adoption through effective SOA governance," *IBM Systems Journal,* vol. 47, pp. 473-488, 2008.

[15] E. A. Marks, *Service-oriented architecture (SOA) governance for the services driven enterprise*: Wiley, 2008.

[16] T. Erl, S. G. Bennett, B. Carlyle, C. Gee, R. Laird, A. T. Manes, *et al.*, *SOA Governance*: Prentice Hall, 2011.

[17] M. N. Haines and M. A. Rothenberger, "How a service-oriented architecture may change the software development process," *Communications of the ACM,* vol. 53, pp. 135-140, 2010.

[18] M. Kajko-Mattsson, G. A. Lewis, and D. B. Smith, "A framework for roles for development, evolution and maintenance of SOA-based systems," in *International workshop on Systems Development in SOA Environments (SDSOA'07)*, 2007, p. 7 pages.

[19] T. Owens, *An Introduction to Scrum: Understanding and Applying the Software*: Speedy Publishing LLC, 2014.

[20] K. Vlaanderen, P. van Stijn, S. Brinkkemper, and I. van de Weerd, "Growing into agility: process implementation paths for scrum," in *Product-Focused Software Process Improvement*, ed: Springer, 2012, pp. 116-130.

[21] A. Wale-Kolade, P. A. Nielsen, and T. Päivärinta, "Usability Work in Agile Systems Development Practice: A Systematic Review," in *Building Sustainable Information Systems*, ed: Springer, 2013, pp. 569-582.

[22] M. B. Miles and A. M. Huberman, *Qualitative data analysis: an expanded sourcebook*. Thousand Oaks, California: Sage, 1994.

[23] H. K. Klein and M. D. Myers, "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems," *MIS Quarterly,* vol. 23, pp. 67-93, 1999.

[24] D. Beimborn, N. Joachim, and B. Muenstermann, "Impact of service-oriented architectures (SOA) on business process standardization - Proposing a research model," in *Proceedings of the 17th European Conference on Information Systems (ECIS 2009)*, Verona, Italy, pp. 2675-2686.

[25] Oracle. (2010, Jan 25). *SOA Maturity Model - Guiding and Accelerating SOA Success.* Available: http://www.oracle.com/technetwork/topics/entarch/oracle-wp-soa-maturity-model-176717.pdf

[26] M. Cohn, *Succeeding with agile: software development using Scrum*: Addison-Wesley Professional, 2009.

[27] A. Cockburn, "Agile software development: software through people," *Wesley, Ed,* 2002.

[28] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Information and Software Technology,* vol. 50, pp. 833-859, 2008.

[29] J. H. Lee, H.-J. Shim, and K. K. Kim, "Critical Success Factors in SOA Implementation: An Exploratory Study," *Information Systems Management,* vol. 27, pp. 123-145, 2010/04/09 2010.

[30] H. Tran, U. Zdun, and S. Dustdar, "View-based and model-driven approach for reducing the development complexity in process-

driven SOA," in *Intl. Working Conf. on Business Process and Services Computing (BPSC'07)*, 2007, pp. 105-124.