The strategic value of SOA: a comparative case study in the banking sector

Richard Baskerville
Department of Computer Information Systems, Robinson College of Business, Georgia State University, PO Box 4015 Atlanta, Georgia 30302-4015
E-mail: baskerville@acm.org

Marco Cavallari
Teamlab s.a., Via Curti, 2, 6901 Lugano, Switzerland
Email: marco.cavallari@teamlab.ch

Kristian Hjort-Madsen
The IT University of Copenhagen, Rued Langgaards Vej 7, DK-2300 Copenhagen S, Denmark
Email: khm@itu.dk

Jan Pries-Heje
The IT University of Copenhagen, Rued Langgaards Vej 7, DK-2300 Copenhagen S, Denmark
Email: jph@itu.dk

Maddalena Sorrentino
Dipartimento di Studi Sociali e Politici, Università degli Studi di Milano, Via Passione 13, 20122 Milano, Italy
Email: maddalena.sorrentino@unimi.it

Francesco Virili
Dipartimento Impresa Ambiente e Management, Università degli Studi di Cassino, Via S. Angelo, Loc. Folcara, 03043 Cassino (FR), Italy
Email: francesco.virili@eco.unicas.it

Biographical notes:

Richard L. Baskerville is professor and chairman of the CIS Department in Georgia State University. His research and authored works regard security of information systems, methods of information systems design and development, and the interaction of information systems and organizations. Baskerville is the author of Designing Information Systems Security (J. Wiley) and more than 100 articles in scholarly journals, practitioner magazines, and edited books. He is an editor of The European Journal of Information Systems, and associated with the editorial boards of The Information Systems Journal and The Journal of Database Management. He is a Chartered Engineer, holds a B.S. summa cum laude, from The University of Maryland, and the M.Sc. and Ph.D. degrees from The London School of Economics.

Marco Cavallari is CEO of TeamLAB s.a., a software company located in Lugano (Switzerland). TeamLAB is focused on the financial sector, and highly specialized


in the development of software components for retail and multi-channel banking applications.

Jan Pries-Heje has a PhD from Copenhagen Business School. He now works at The IT University of Copenhagen and is also Professor (part-time) at the IT-university in Gothenburg. His main research interests are information systems development, software engineering, and software process improvement. In particular he has carried out action research with industry on specific topics such as high-speed software development, IT project management, requirements specification, and successful organizational change with IT. He has published in these areas in journals like Journal of Accounting Management and Information Technology, IEEE Computer, The Data Base for Advances in Information Systems, European Journal of Information Systems, and Annals of Software Engineering.

Kristian Hjort-Madsen is a strategic enterprise architect with Denmark’s Ministry of Science, Technology and Innovation. He is currently writing an industrial PhD in combination with his work in a public-private partnership between the Ministry, the IT-University of Copenhagen, IBM Denmark and KMD. His works and research focus on the implementation and management of enterprise architectures in public agencies and he is widely used as a guest lecturer and strategic advisor on EA related matters. A more detailed bio and significant publications can be found on Mr. Hjort-Madsen’s blog: www.EAGov.com.

Maddalena Sorrentino researches in Information Systems and Organizational Change. She teaches Information Systems and Public Sector at the University of Milan (Department of Social and Political Studies). She has published in academic proceedings, such as ECIS, DEXA-eGOV, Bled eConference, IFIP (OASIS) and is the author of seven books. Maddalena is a member of the editorial board of Information Systems and e-Business Management. She has extensive experience of IS evaluation in the public sector.

Francesco Virili, PhD in Information Systems from University of Siegen (Germany), is assistant professor in Organization Science and Information Systems by the “Università degli Studi di Cassino” (Italy). In addition to publishing a book on standards in organizations and several book chapters (in Italian), Virili presented several papers at the European Conference on Information Systems (ECIS), DEXA-eGOV, the Bled eConference, IFIP (OASIS), and at a MISQ special issue workshop on IT standardization processes. He is a member of the editorial board of Information Systems and e-Business Management.

Abstract: Information and communication technology (ICT) has helped to drive increasingly intense global competition. In turn, this intensity increases the need for flexibility and rapid changeability in ICT to support strategies that depend on organizational agility. We report a comparative, cross-cultural case study of the implementation of Service Oriented Architectures (SOA) at a Scandinavian bank and a Swiss bank. The strategic rewards in the adoption of SOA appear to go beyond marketplace issues of ICT capability acquisition, and unexpectedly arise in the creation of an extensible organizational ICT architecture. The extensibility of the ICT architecture that results from the adoption of SOA provides potential for greater organizational agility (and thereby competitiveness).

Keywords: Architecture, Service, Service-Oriented, SOA, Case Study, Strategic value, Bank, Extensibility, Reusability, Service evolution.
1 INTRODUCTION

Information and communication technology (ICT) has helped to drive increasingly intense global competition. The upswing created by electronic commerce in the late 1990s has continued to expand into the first part of the new century as global digital bandwidth has increased. Companies increase globalisation through sourcing strategies that make geography increasingly transparent for digitally based communications and operations.

ICT innovation is both driving and being driven by this global competition. For example, ICT innovation is driving competition because it helps organizations to become network enabled, lowering entry barriers and opening access to new entrants. On the other hand, global competition drives ICT innovation because intense competition generates a stronger need for more advanced ICT. Service Oriented Computing (SOC) (Papazoglou and Georgakopoulos, 2003) and Service Oriented Architecture (SOA) are among the major technical advances that have attracted significant and growing adoption since 2000.

SOA has been the subject of some hype. The simple addition of Web services and XML to the application portfolio is not magically transforming organizational technical environments (Erl, 2004). We need a better understanding of the practical implications of principles behind SOA technologies. In this paper, we explore the strategic value of applying SOA for banking institutions, and examine the impact of the promising technology behind SOA.

Given the relatively recent availability of workable SOA, there are few empirical studies of the organizational impact of SOA adoption projects, especially for industries, such as banks, that are often encumbered with intractable legacy systems. In this paper we comparatively explore two organizations from the same industry but from different countries. We study SOA adoption in a Scandinavian Bank and in a Swiss bank. For purposes of anonymity we have named them “Northern Europe Bank” and “Central Europe Bank”. The purpose of this article is to report the insights into the organizational impact of SOA adoption in these two cases.

The remainder of this paper is organized as follows: The next two sections discuss the architectural challenges today in banking and the strategic concepts behind SOA and Web services in greater detail. We then outline our research method and present the two case studies. We then discuss the two banks’ different approaches to SOA and conclude with some of the key attributes of a SOA.

2 ARCHITECTURAL CHALLENGES AND PERFORMANCE IN BANKING

Banking provides a particularly interesting setting in which to study SOA adoption. Banks are struggling with heterogeneous legacy systems that are difficult to change and integrate. Deregulation as well as shrinking profit margins and rapid growth in online banking are impacting the global financial market and requiring rising organizational agility (Research and Markets, 2005). The traditional restrictive banking laws have resulted in very fragmented industry IT infrastructures, which must now be adapted to
meet the new demands for customer-driven, specialized and highly automated virtual banking. Banks besotted with antiquated ICT architectures now confront rising demands for flexible and cost-effective information systems (IS) architectures. The ability to acquire a new ICT architecture is becoming a key competitive factor to support the new financial business drivers in banking.

However, older architectures complicate integration of enterprise applications because the underlying elements have created “closed” architectures. Closed architectures restrict access to vital software and hardware configuration and force organizations to rely on single-vendor solutions for parts of its ICT (Mazursky, 1989). Banks are struggling with a range of applications running on different COBOL, PL/1, .Net or Java platforms. The more-or-less closed architectures impede the banks’ ability to offer new and integrated financial products, merge with competitors or consolidate redundant operations.

For example, in our Scandinavian case study, it is no trivial task to offer a range of different financial services as a consolidated customer package. The Northern Europe Bank had developed a product for private customers that collected a wide range of financial products into one package. The challenge with this kind of integrated banking product is its requirement for integrated business and IT processes. Without this integration it would take two hours for a sales representative to set up the new package for one customer. Using their integrated enterprise applications, it takes ten minutes.

2.1 Strategic value of SOA: four challenges

The contemporary banking scenario briefly sketched here reveals a few basic strategic issues that are potential sources of strategic value from SOA. The strategic value of Information Technology has been extensively investigated in the IS literature, over several streams of research. Actually this topic is grounded in the tradition if IS research: For example, Mason, McKenney and Copeland (1997) report 24 exemplary studies in the period 1983-1994, commenting that "these studies establish the fact that information technology, when it is employed effectively, contributes to a firm’s strategic performance" (p. 271). Mocker and Teubner (2005) provide a critical overview of recent IS strategic management thought, and Peppard and Breu (2003) review the literature regarding strategic alignment.

Since the primary seminal works on the topic, numerous strategic management theories and frameworks have been proposed to explain IT strategic value. We used a basic, simple notion of strategic value: An information technology (and specifically the SOA technology) has strategic value if its adoption is required (among a multiplicity of other factors) in order to build and maintain a competitive advantage. In other words, the strategic value is generated by the enabling effect of SOA on the bank's sustainable competitive advantage.

As with any Information Technology, the enabling effect of SOA on sustainable competitive advantage implies that SOA adoption is a necessary but typically not sufficient condition: Further factors are required in order to form a combination of strategic capabilities that are unique and difficult to imitate. Among the numerous studies and perspectives that are converging on the factors of IT-enabled competitive advantage, Mata, Fuerst and Barney (1995) advance a convincing and authoritative resource based model. This model, as illustrated in figure 1, explains how sustained competitive
advantage stems from three conditions (from left to right): 1) **existence:** the existence of a valuable resource or capability; 2) **heterogeneous distribution:** the non-availability of the same resource/capability for competitors; 3) **imperfect mobility:** the impossibility for competitors to acquire, develop or use it.

In order to produce sustained competitive advantage, SOA technology alone cannot represent a uniquely "valuable resource or capability", because it is potentially available to all competitors. Instead, SOA is a potential enabler that permits a bank to generate an original, unique blend of resources and capabilities that could not be produced without SOA. Ideally, it will be difficult for other competitors to replicate this blend. (Section 3 discussed the technical aspects of this SOA enabling effect.)

The strategic aspects of the enabling effect of SOA relate to four challenges that confront the contemporary banking industry in Europe. First, for historical and technical reasons, among modern European banks there is typically a strong need for **application integration**. This issue is introduced in general terms by Lynne Markus (2000), and discussed more specifically for the banking sector by Pan and Viña (2004), who focus on architectures for integration. Modern banks increasingly struggle with a plethora of complex products and services managed with a wide variety of applications and systems that often operate on different platforms. A typical banking situation will include several layers of historically separate IT artefacts that were accumulated over time. It is difficult to get such different systems to interact together effectively and efficiently (Pan and Viña, 2004). As a result, IS development (ISD) in modern banking involves an increasing application integration workload in order to bridge, connect, and incorporate new functionalities into the existing IT environments characterized by intractable legacy systems. Not surprisingly, application integration is growing as a strategic aspect for competitive banking.

A second contemporary banking challenge arises in the frequent need for banks to enable **value reconfiguration** processes (Seifert and Wimmer, 2001; Homann et al., 2004). “To decrease costs and simultaneously enhance customer utility, banks are increasingly focusing on their individual core capabilities while exploring different sourcing options for non-core capabilities. Consequently, they are disaggregating their value chain into independently operable functional units. As communication capabilities reach higher levels of performance and reliability, these functional units are combined across corporate borders” (Homann et al., 2004, p. 34). In this way, value **chains** are disaggregated and recombined into value **networks**. These value reconfiguration processes are connected to environmental complexity and instability. Environmental complexity demands value networks because these networks better serve complex market demands than value chains. Environmental instability demands value networks because value chains are cumbersome against unexpected market demands or other emergent opportunities.

A third challenge is embodied in the need for ensuring **value preservation** after mergers and acquisitions (M&A). M&A reconfigure value, sometimes with significant value destruction (Bruner, 1999; Smith and O’Neal 2003). The dominant organization imposes its own management, culture, and practices; thereby it seriously compromises the target organization’s culture, knowledge, and distinctive capabilities. When the target
organization’s old systems are brutally replaced, knowledge and practices embedded in these systems are lost.

The fourth challenge is the need for more agile forms of IS development. Agility is a special form of flexibility that makes change both quick and easy. From the strategic point of view, agility is the "ability to detect and seize market opportunities with speed and surprise" (Sambamurthy et al., 2003, p. 238). For organizational change and adaptability, agile organizations are nimble and light in motion. Agile organisations “respond quickly, they are resourceful, and they are able to adapt to their environment” (Mathiassen and Pries-Heje, 2006). Quickness is about the speed with which the organization can respond to customer requests, market dynamics, and emerging technology options. This speed includes the time to sense relevant events, the time to interpret what is happening and assess the consequences for the organization, the time to explore options and decide upon actions to take, and the time to implement appropriate responses (Haeckel, 1999). Resourcefulness embodies the capabilities that are available within the organization, including people, technology, processes, and knowledge. Resources can be both tangible and intangible and they provide the basis for doing business and instantiating change (Haeckel, 1999). Adaptability relates to the organization’s ability to respond to changing demands, threats, or opportunities. This response requires an ability to learn along with flexible processes and products that can be reconfigured without extensive additional costs (Dove, 2001; Haeckel 1999).

Banks are not historically regarded as very agile organisations. Banking is weighted with regulation and security, and consequently these are very traditional organizations. But their contemporary environment increasingly rewards organizational agility. Customer needs are evolving. Consequently, modern banks are driven to continuously widen their range of different activities and services to meet continuously changing customer demands. This wider range of activities increasingly involves collaboration with partners and even with competitors. These new products, services, activities, and relationships often require embracing new information systems or undertaking systems integration, redevelopment, and adaptation. Continuously redeveloping these banking systems means that even the most conservative banking institutions are driven to transition from traditional development towards more agile ISD approaches (Truex et al., 1999; Baskerville et al., 2001; Baskerville and Pries-Heje, 2004).

Given these four challenges faced by contemporary banking institutions, the strategic value of SOA will lie in its potential to enable sustainable competitive advantage in organizations burdened with heavy regulation and intractable legacy systems; such organizations must nevertheless compete in a marketplace that requires them to be agile in using IS for value reconfiguration and value preservation. SOA as a response to the first and the fourth challenges (application integration and agility in ISD) surfaces with particular prominence in our two case studies, as discussed in section 7.

2.2 SOA and firm performance

Quantifying the sustainable competitive advantage enabled by SOA, and the positive impact of this advantage on firm performance, is problematic. The relationship between IT spending and firm performance is the object of numerous, challenging, and often surprising investigations at different levels of analysis. Work regarding the "productivity paradox of IT” is particularly important, motivated in terms coined by Nobel Laureate
Robert Solow: "You can see the computer age everywhere but in the productivity statistics" (Solow 1987, p. 36). Dozens of empirical studies in several industries, as reported over the years, offer conflicting and often negative evidence of a positive relationship between IT spending and different measures of productivity and performance (Dedrick, Gurbaxani and Kraemer 2003).

In the banking industry, which is traditionally characterized by high levels of IT spending, technological progress is widely recognized as a major source of beneficial change (ECB 1999). The few studies specifically focused on the impact of this IT spending on bank performance are congruous with the studies in other industries, and have failed to find clear financial benefits from IT spending (Markus and Soh 1993; McKinsey Global Institute 2001).

Evaluation of IT impact on performance depends on different methods of classifying and measuring both IT spending and firm performance. There have been recent advances in these methods. For example, Beccalli (2006) used distinct data sets for hardware, software and IT services spending in European banks. Instead of traditional accounting indicators (like ROA and ROI), Beccalli measured performance using a more advanced X-efficiency score that distinguishes between cost efficiency and revenue efficiency. Results confirm the conventional belief among bankers that IT spending is typically more valuable for cost reduction than for revenue expansion.

Because our study involves two SOA projects analysed in their initial stages, it was not possible to draw conclusions about the impact on firm performance. However, our studies below show how effective use of SOA leads to systems characterized by service integration, service reuse and service evolution. These characteristics indicate new ways to reduce and distribute IT spending over several different business units, and to improve cost efficiency with promising, positive short term impacts on bank performance. This indication is consistent with the published evidence that IT spending in banking may be justified by cost reduction.

In the long term, strategic management theory suggests that the achievement of a sustainable competitive advantage should have a sizeable and durable positive effect on firm performance. Given the alignment of SOA with the banking industry’s strategic challenges of application integration, value reconfiguration, value preservation, and agile development, current developments in the use of SOA in banking may open opportunities for future researchers to better measure long term positive impacts of IT on firm performance. We are studying IT developments that are preliminary to such studies.

3 SERVICE ORIENTED ARCHITECTURE

The Service Oriented Architecture (SOA) is an architectural style used for building loosely coupled distributed systems that deliver application functionality in the form of services for end-user applications (Ho, 2003). From a strategic business perspective, SOA is built around the notion that services map to business functions (Datz, 2004).

SOA promotes well-defined, published, and discoverable interfaces that deliver reusable application functionality for distributed systems, i.e., where services are invoked over a network. In this way, SOA is more than a traditional Information Systems Development method because it embraces business-process modeling and enterprise
architecture, as well as object-oriented design and distributed systems (Ho, 2003; Erl, 2004; Erl, 2005).

SOA has the following basic features (Erl, 2005):

- **Service contracts**: Each service incorporates and presupposes a contract, i.e. an agreement that specifies the functionalities offered while also guaranteeing replicability and reuse (see also Meredith and Bjorg, 2003).
- **Service reusability**: Logic is divided into services with the intention of promoting reuse.
- **Service independence and loose coupling**: Services maintain a relationship that minimizes dependencies and only requires that they maintain an awareness of each other.
- **Service autonomy**: Services have control over the logic they encapsulate.
- **Service abstraction**: Services hide internal logic from the outside world.
- **Service composability**: A collection of services can be coordinated and assembled to form composite services.
- **Service statelessness**: Services minimize retaining information to that specific to an activity.
- **Service discoverability**: Services are designed to be outwardly descriptive so that they can be found and assessed via available discovery mechanisms.

According to (Erl, 2005), besides the principles listed above, additional principles can be envisaged. In particular, the following aspects could deserve attention.

- **Service granularity**: Service granularity refers to the scope of functionality that a service exposes; while the fine-grained interface offers more flexibility, it also means that patterns of interaction may vary between different service requesters (Colan, 2004; Zhang and Tanniru, 2005).
- **Platform independence**: SOA is available on multiple platforms (e.g. Linux, Windows, Unix of different flavours, AS400, MVS).
- **Standardization**: SOA is based on common standards.
- **Service integration**: Service integration is one of the two basic economic drivers of SOA. Compared with other approaches, SOA proposes a paradigm shift from the point-to-point integration of stand-alone systems to a broad-based, seamless integration of heterogeneous and specialized systems that delivers highly customized business functionality.

As a driving economic force for SOA adoption, service integration is joined by service reusability. Reuse enables a faster return on the related implementation investments, while operating and maintenance costs of a service are divided across all service users. The actual need for reuse is not necessarily constant across the whole portfolio of enterprise business services. Reusability is guaranteed by the contract stipulated between the service supplier and the service consumer. A contract specifies the functionalities of the underlying service, which must be guaranteed and maintained over time and under all conditions.

### 3.1 SOA extensibility

In this section, we illustrate the trade-off between service evolution and service reusability in the banking industry. Then, we introduce what we call “SOA extensibility” as an answer to this trade-off.
Service reusability and service evolution in a typical banking application portfolio

A banking application portfolio typically includes several software applications that remain stable over time. The portfolio will also include other applications that need to evolve continuously in order to meet emerging business needs.

For one part of the portfolio, application stability facilitates the “freezing” of service contracts and hence reuse of existing services. In another part, the need to promptly meet uncertain and emerging requirements would conversely require changes in the service contracts over time. For example, changes may be required when a service is reused in a new way that was not originally planned when the software contract was frozen.

The Service Reusability Factor expresses the probability that a particular business service will be reused by another business service. The Service Evolution Factor expresses the probability that a business service will require an evolution of its published interface. This evolution might be due to emerging business requirements or to the need to extend its functionality after its recombination or reuse. The two factors are often antithetical, expressing a trade-off: high reuse often demands low evolution; on the other hand, high evolution may compromise reuse. This trade-off is evidenced by the analysis of banking application portfolios.

As shown in Figure 2, an application portfolio can be classified along the lines of the two dimensions mentioned above (i.e. evolution and reuse). Banking systems typically encompass a high number of basic and infrastructural services, where the potential for reuse is high. One example is that of the general ledger application, or the debit/credit transaction service. Such functionalities are often well-known, highly efficient, and widely available. They also have quite stable interfaces. Their high degree of reuse can be characterized in terms of both frequency of use and number of service consumers. Their projected evolution is usually low, except for dramatic changes due to major events, such as the introduction of the Euro, M&A operations, partial outsourcing, etc. (Figure 2, Quadrant 4: high reusability, low evolution).

Other low-evolution services in the application portfolio may have a lower degree of reuse. These services are typically quite stable and focused on specific purposes, such as mortgage calculations (Figure 2, Quadrant 3: Low reusability and evolution).

In other cases, certain banking services might demand a fast response to changing business needs, but have a low potential for reuse (Figure 2, Quadrant 2: low reusability, high evolution). These cases arise when service functionality is specifically designed as not reusable (“one-shot”) and is tailored to a particular context.

Finally, in some cases, fast evolution and intense reuse would be needed. These cases especially include those services at the forefront of business operations. For example, many banks are implementing a “multi-channel infrastructure.” These include CRM and wealth management systems for which reusability should go hand in hand with evolution (Figure 2, Quadrant 1: high reusability, high evolution). The introduction of standard Web services technology provides a new technical solution for ensuring both reuse and evolution: the extensible SOAs.

SOA extensibility

When managed appropriately, standard SOAs (usually based on Web services standards) can often provide the basis for a technical solution to needs for fast evolution and intense reuse. This solution involves a service contract extension.
A service contract can be extended in two ways: (1) by enriching the existing functionalities; and (2) by adding new functionalities. In both cases, developers of extensions must maintain compatibility with the preceding versions of the same contract. Technically speaking, Web services – with service contracts based on WSDL and protocol based on SOAP – allow a good degree of freedom because both WSDL and SOAP are XML-based and thus extensible by nature.

SOA extensibility offers a new technical solution to the trade-off between service evolution and service reuse. It has the potential to enable the development of high-reuse and high-evolution applications. These applications represent a new and strategically relevant category of applications in the banking portfolio. The strategic challenges discussed in Section 2 provide a framework for analysing the strategic relevance of these banking applications (those that fall in quadrant 4 of figure 2 and are potentially enabled by SOA extensibility).

3.2 SOA and strategic challenges

Different degrees of service evolution and service reuse represent technical enablers to cope with the strategic challenges described in section 2. These enablers permeate the banking application portfolio. In particular, the use of SOA to increase reusability and/or evolution will help the organization to cope with the strategic challenge of application integration. Similarly, SOA has similar effects when the rate of change in the requirements tends to be high. In this case, the strategic challenge is that of development agility, representing applications shown in figure 2, quadrants 1 and 2. Where the strategic challenge is that of value preservation and value reconfiguration, the need to integrate and recombine the applicable functionalities makes reusability and SOA a factor of strategic relevance. In this case, it indicates applications found in figure 2, quadrant 4. These examples show how application integration concerns the entire application portfolio.

The adoption of Service Oriented Architectures may play an enabling role in resolving four important strategic challenges in the banking sector, viz., system integration, value reconfiguration, value preservation and agility. In particular, extensible SOAs are specific technical enablers for ensuring low cost agility, i.e. coupling agility (service evolution) and efficiency (service reuse).

In the following sections we will examine two different approaches to SOA adoption through a comparative case-study analysis.

4 RESEARCH METHOD

Given the exploratory nature of the research question, we selected an in-depth case study approach relying on data triangulation (Yin, 1994). Case studies facilitate multi-perspective analyses that consider not just the voice and perspective of the actors, but also those of the relevant groups of actors and the interaction between them (Tellis, 1997). This analytical depth leads to a holistic understanding of cultural systems of action (Feagin et al., 1990; Tellis, 1997), providing the insight that satisfies exploratory questions.
We collected data by using both semi-structured interviews and document reviews, two important forms of evidence collection that satisfy Yin’s (1994) principles of data collection: (1) multiple sources of evidence, (2) a case study database, and (3) a chain of evidence. Our mode of analysis used Yin’s “pattern matching,” a comparative analysis involving the establishment of non-equivalent dependent variables as a pattern. Each case has a variety of outcomes that we analyse to develop common explanatory patterns across the comparative case data. This qualitative research design supports the study of complex, dynamic social phenomena that are ‘both context and time dependent’ (Orlikowski and Baroudi, 1991). In our results, we engage in a class of generalizability that Lee and Baskerville (2003) call type “EE”, the generalization of data to a measurement, observation, or other description.

To study the strategic value of SOA we developed access to two banks, each in a different area of Europe. Our selection criteria limited us to banks with active SOA development projects. We sought different areas in order to distinguish comparable aspects of banking as an industry from contrasting aspects a regional or national culture. We studied projects in a Swiss bank and in a Danish bank. Two of the authors participated in architecture related projects in the Northern Europe Bank and three of the authors had the same opportunity in the Central Europe Bank. Participant observation is common in case studies, but raises limitations because of a trade-off between its “unusual opportunities for data collection” (Yin, 1994) and the “potential biases” produced by close involvement. We minimized this bias through a full-day workshop in which all of the authors presented findings and non-participants scrutinized patterns proposed by participants.

In the following two sections, the case studies are briefly analysed. This analysis surfaces evidence the way that different contexts, IT strategies, challenges and results can generate strategic value. Two tables summarize the contribution of each project to strategic value generation across the four strategic dimensions discussed in section 2.

5 CASE: CENTRAL EUROPE BANK

Business Model: Central Europe Bank, located in Switzerland, is part of a large financial group that operates both at the national and international level. The core business of Central Europe Bank is private banking (i.e. banking services, including lending and investment management, for wealthy individuals). It is one of the market leaders in this sector. The typical customer of Central Europe Bank has substantial wealth. The bank’s distinctive values include discretion, privacy, professional service, personal trust, and individual relationships. Specialized professionals provide the core business services.

IT Strategy: In one interview, a manager described their approach to IT: “In IT adoption we are a fast-follower.” Their attitude towards IT is quite conservative (e.g. the company’s information system is managed internally by the IT Division). As a market leader, they must still pursue new opportunities offered by innovation, but only on the condition that it involves a low level of risk. Central Europe Bank displayed a general preference for well-defined and orderly rules (something often characterising the Swiss national culture). They set up a small Enterprise Architecture Integration (EAI) organizational unit, and started investing in new IT resources like middleware and infrastructure software. Their general aim was to gradually, prudently build a corporate-
wide EAI architecture. A strategy of radical innovation would have been somewhat incongruous with other bank values: those of prudent, risk-averse Swiss bankers.

**Project Description:** A new CIO was attracted by the potential benefits of EAI and SOA in this environment. He was also aware of the related high level of uncertainty arising from the still immature nature of this technology, and the competence gap due to lack of experience.

The need for a new cashier management system (CMS) provided a low risk, “test bed” project opportunity for developing experience with SOA. Initially the main motivation for launching the project was to quickly and cheaply build a state-of-the-art experimental CMS based on the promising (but unfamiliar at the time) SOA and Web services technology. The CMS project, being limited to the bank’s teller area, was low risk because teller operations are typically minimal in the CE bank. The teller is not involved in the portfolio management transactions that represent the bank’s core business. Moreover, customers rarely approach tellers for transactions like bill payment, cheque encashment, or cash withdrawal. Mostly tellers deliver a limited number of complex transactions of high average amount, along with other services like safe deposits management and non-cash items collection. Cashier management does impact the bank’s image, but it is not a mission-critical activity. Development of a new CMS was less risky than other operational process areas that were closer to the bank’s core business. However, as it will become evident below, the CMS project itself had relevant strategic value independent of the underlying CMS. The strategic value of the project arose because it contributed to the enhancement of organizational “fitness for future” (Thompson 1967, p.84). The organizational learning from the CMS project helped the bank achieve a new kind of IS development process. This process had some (but not all) typical patterns of agile development and some (but not all) patterns of traditional development. The bank obtained a higher degree of application integration through Web services and Service Oriented Architecture.

Two years earlier, an internal project team conducted a thorough system analysis for the CMS using a traditional, waterfall-style approach. The low priority of CMS had delayed its development until the new CIO elevated it as a low risk learning opportunity for the bank to acquire sought-after SOA competences.

CMS development was partly outsourced to a vendor that was highly specialized in software development for financial and retail banking systems. Actually, the relationship was closer to a partnership than to traditional outsourcing (Ye and Agarwal, 2003). According to several interviewees both in the bank and the vendor, trust had a central role in determining project success. The project team included specialists from both the bank and the vendor. After a short revision of the existing analysis, extensive prototyping was started. System prototypes were used as “boundary objects” (Carlile, 2002) in order to incrementally review and collaboratively test parts of the system with users. User interactions were also opportunities to discuss and negotiate new requirements.

In September 2002 the first prototype was delivered to six tellers and was used to clarify and refine the requirements that were later incorporated in subsequent releases. The full testing phase engaged the chief cashier (head teller) in one branch for three months. In June 2003 the system began operation in a second branch. The old system was discontinued in December 2003.

**Project Objectives:** The CMS project had three main objectives: First, to introduce a new philosophy of service leading to innovative organizational practices for cash management. Second, to leverage the recently introduced EAI through a pilot Web-
services based SOA application that had an acceptable level of risk. Third, to build new competences and knowledge in the SOA field that will enable the bank to continually redevelop its systems as markets demand.

Project Challenges: The CMS project expected three main challenges: First, the new application philosophy would be a hurdle for developers experienced in older software development paradigms. The introduction of the new development paradigm would be a major technical challenge because it had to be tested within the Central Europe Bank. Few comparable solutions had been developed elsewhere in the banking market, making previous experience rare and technical competencies rather unavailable. Second, the innovative software development tools such as Web services on a Microsoft .Net platform had just entered the market at that time. There was little expertise or experience with these tools. Third, the integration between the CMS and the legacy systems through EAI and SOA was complex. The bank’s pre-existing CMS was developed in the ‘80s in a hybrid environment with multi-terminal and LAN, OS/2, CICS and COBOL.

[INSERT HERE TABLE 1]

Project results: The integration of the new CMS with the bank’s legacy system was achieved via EAI and SOA. The EAI unit worked in strict coordination with almost every functional unit belonging to the IT Division. The new CMS replaced the pre-existing client/server application with a multi-tier, web based system. Using the new CMS, the end-user (i.e. the bank teller) interacts with the application via a web-browser. The user interface was implemented in Microsoft ASP .Net technology and the business logic was implemented as Web services.

The new CMS enables a new way of working and interacting with the customer. The bank decided to replace the traditional “transaction based” approach in favour of a more sophisticated “operational approach”. Within the CMS, a “service operation” is a set of single transactions. The desired service operation is assembled at the counter on the basis of customer’s requirements. The whole operation can be traced on a step-by-step basis. Alternative solutions (e.g. potential revenues across exchange rates and service charging) can be simulated and the effects of certain actions can be interactively discussed with the customer before actually launching the transactions. When ready, transactions that exactly meet customer’s objectives are executed through a single command.

SOA Business benefits: Some of the key business benefits of SOA in the Central Europe Bank arise from the project’s enhancement of application integration. The integration approach based on SOA, compared with a more traditional EAI approach, has significant added value. It enables decomposition and recombination of new and pre-existing functionalities into an overall service oriented reference model. The reference model, once implemented, hides much of the complexity of its technical implementation and it masks the peculiarities of legacy software modules. For instance, in the CMS system, whenever a cash transaction is involved, the number of bills exchanged is explicitly defined. The legacy system had only able been able to deal with a total amount for each currency. The new service is based on a reference model that is closer to the teller view. Whenever required, the CMS translates the number of banknotes into a total amount, masking the underlying legacy transaction and hiding the complexity of functionality extension. In business terms, the new system is able to build, simulate and quickly execute complex and composite transactions in real time. This service thereby discloses a new way of interacting with the customer and providing new high-value services. It
also offers the possibility for the cashier to keep the bank’s profit margins under control when making a proposal. In this way, the new system generates new capabilities, by original (and unique) recombination of pre-existing information resources, thanks to SOA and Web services. The new capabilities, being valuable, unique, and difficult to replicate, are a source of strategic value, in the sense discussed above in section 2.

Systems development practices: fast development cycles were observed, even under Central Europe Bank’s stringent quality standards. The new software development process shares aspects of traditional development (like deep, extensive initial analysis; project planning with milestones; emphasis on quality and reliability). It also shares aspects of agile development (like incremental development, requirements revision, and new requirements emerging through use).

The Web-services based SOA was able to deliver good levels of flexibility and organizational agility without the need to freeze service specifications at any stage of the development. It enabled a degree of adaptability and rethinking that could be accepted even at late stages during the implementation phase without compromising previous developments. The reusability of business components (i.e. of Web services) has subsequently been demonstrated by more recent projects in the Central European Bank. In these recent projects, new functionality was added to the original business components, e.g. by adding new methods or by adapting the existing ones⁴. Very few Web services could be reused in the exact form as originally implemented; most of them required minor changes. Still, changes in the original Web services did not affect their consumers. Software modules, developed on the basis of the previous version of a business component, were usually not affected by extended functionality.

Experience gained in the CMS project showed that, in order to be reused by different consumers, a service component often needs to be extended. Web services extensibility proves to be a key enabler of such reuse, unleashing the significant business benefits of reusability.

Moreover, Central Europe Bank formed a new software development team that is rapidly acquiring skills and capabilities on .Net and Web services technologies. More than half of new internal IS development efforts are now based on SOA and Web services. The partnership with the external vendor was especially valuable for organizational learning. The CMS project had a key role in launching and nourishing an organisational learning process with high potential strategic impact on IS development practices.

In summary, the Cashier Management System, based on Service Oriented Architecture and Web services, generated new strategic value for the Central Europe Bank, achieving the business benefits of organizational agility by innovative recombination of existing IT capabilities. The system was built in a peculiar way, merging aspects from traditional development (quality, reliability) and aspects from agile development (speed, flexibility). In this case, neither purely traditional nor purely agile systems development were found to be the best solution. The best solution arose in the fortunate encounter of the CE bank’s traditional ISD culture with the agile culture of the software company. This encounter provided opportunities for organizational learning by the bank.
6 CASE: NORTHERN EUROPE BANK

Business Model: With around 18,000 employees and more than three million private customers in Denmark, Norway and Sweden, Northern Europe Bank is today one of the largest banks in Northern Europe. The bank offers a wide range of financial services ranging from insurance to traditional banking to real estate sales. It has nearly one million home banking customers using a range of advanced financial services online. There are relatively few large providers of financial services in the region and customer loyalty has traditionally been very high. But the growing international competition that has emerged as a consequence of the open market in European Union, together with the technological innovations that facilitate time-space dissociation, are challenging Northern Europe Bank’s dominant position.

IT Strategy: Northern Europe Bank’s IT strategy is today focused at aligning the business and technology environments to match the request for changeability and reusable business functionality. The recent history of many banks has been one of a long series of mergers and acquisitions that hobble consolidated banks with an incompatible hodgepodge of legacy systems. Northern Europe Bank is no exception. But in terms of its value configuration, the bank distinguishes itself from competitors by its consolidation of its entire infrastructure on the same IT platform. Northern Europe Bank calls this strategy, “one bank, one system”. All divisions within the company work on the same IT infrastructure. They developed a strategy called “first prover”. The strategy reflects their intention to be among the first companies to reap the benefits from new ideas and technologies in the IT world in order to gain competitive flexibility and to cut IT spending.

Northern Europe Bank’s IT architecture was developed as a network centric architecture with a range of separate self-contained “silo” application systems that were oblivious to other systems within the bank. The architecture was not only a roadblock to their one-bank-one-system application integration strategy; it was a roadblock to the continuous redevelopment implied by their first prover strategy. To overcome these enterprise integration challenges, they championed a company-wide SOA strategy beginning in 2002.

Project Description: Northern Europe Bank started running pilot-projects using component based development methods in the middle of 2001 and early 2002. At the end of 2002 the SOA concepts were introduced. These were based on the limited theory available at that time and the experience gathered by the bank’s chief architect from participating in different international forums with leading software vendors. The management board adopted the SOA strategy and a new company-wide development model (based on the new SOA concepts) replaced Northern Europe Bank’s existing development model in the middle of 2002.

Project Objectives: The project sought to make the bank a first prover of SOA as a means of integrating silo banking applications into a cohesive, integrated, enterprise information architecture.

Project Challenges: The silo applications were the main challenge. The customer “packages” mentioned in Section 2 are a good illustration of the complexity and
redundant functionality that existed in the old IT architecture. These packages were a combination of different financial products and they achieved great commercial success with over 300,000 customer packages sold. From the sales rep’s perspective it was easy to combine existing offerings into new packages. The problem was that the IT systems did not support this combination and recombination of different offerings. Each time a customer bought a customer package, a sales representative had to access each individual bank product’s interface included in the package, cut data out from a Word document and write them into the customer’s individual package. Afterwards the customer signed the agreement and the sales representative finally created the package as a combination of all the systems. To a certain extent, the one-bank-one-system philosophy saved the bank because it was fairly easy to automate the business processes in the back office. This example illustrates the integration challenges that made the IT-department at Northern Europe Bank reconsider its current IT architecture.

**Results:** For Northern Europe Bank the move towards SOA represents a paradigm shift in the way they develop application systems. Their approach was incremental. The implementation of any new system is based on SOA development principles. A service integration layer now facilitates the integration of the many legacy systems in the bank. This layer is a broker-like service integration built in-house. It is based on XML and MQSeries technologies and facilitates the publication of services for service providers and the lookup of service interfaces for service consumers. Such functionality is often referred to as an “enterprise service bus” (cf. Colan, 2004). The service integration layer has a many-to-many infrastructure, making it possible for applications on the same platform (e.g. Cobol) to communicate directly with each other via a local service registry. Registries are replicated across the service layer. This replication means services across different platforms (e.g. Java and COBOL) can find and invoke each other’s services across the company just by using their own service registry. Because the services have well defined interfaces as their fundamental attribute, the integration with other services is conducted without regard to the physical implementation of the services. The bank emphasises that one of the key advantages of the broker-like service integration layer is call mechanism optimization. The layer intelligently (based on built-in knowledge) optimises which call mechanism is the best for calling and receiving a service request.

Although SOA addressed many of the traditional problems of integrating disparate business processes and applications, deploying service-based applications for reuse introduced new management challenges. These challenges include defining what services to build for reuse, developing and testing applications composed of operational services, deploying and provisioning distributed service-based applications across organizational boundaries in a secure, reliable, and repeatable manner. These activities are difficult and require a new way of thinking about ISD. Because the Northern European Bank views services as key assets in the design of systems, technologies and techniques for management and governance of assets had to become an integrated part of the organizational development model. The strategy involves the reuse of as many services as possible in new projects (cf. the reusability factor). Services are stored in a “service library” where projects can identify and publish common services in accordance with a strict set of guidelines. When potential services for reuse are identified, an architectural board decides whether to fund the development of common services. Subsequently, projects are required to reuse the services.

The bank’s external integration with customers and suppliers in the financial value chain has also been improved as a result of the SOA strategy. The bank’s initial SOA
strategy had an internal focus. But as more than a hundred Web services developed in the bank initially, these can be used both in internal systems and by external partners. Over time the bank expects to sell and distribute hundreds of thousands of service packages. The Web services are exposing their service interfaces and operations in the service integration layer which offers great extensibility advantages. In other words, the wall-to-wall SOA strategy has built a flexible and cost-effective IS architecture for Northern Europe Bank that makes it less important whether services are internal or external.

7 DISCUSSION AND CONCLUSION

Our analysis of these cases indicates several common organizational needs that appear to be growing for the two banks we studied. These needs are connected to rising demands for application integration and continual redevelopment in the banking sector. The answers analysed here were based on SOA and in one case on Web services; indeed, the Northern Europe Bank case illustrates that it is not necessary to use Web services internally to achieve a Service Oriented Architecture.

Figure 3 visualizes how the SOA approaches in the two banks could respond to such demands, facing the four strategic challenges discussed in section 2. The figure contrasts the SOA potential and the results that were achieved in the two projects.

There were Business Drawbacks from the use of SOA. The Northern Europe Bank case showed that the service concept was difficult to define in practice. They defined services in a way that was partly different from the definition of SOA generally used. In actuality, they left out business processes. However, they defined services in a way that is similar to the way that Web-services experts use the term. It appears that companies adopting a SOA strategy may struggle when defining the difference between classes, components and services. Nevertheless good application integration and value preservation was achieved.

Central Europe Bank faced problems because its overall enterprise architecture strategy was still at an early stage. The bank had to train a new person to manage the CMS application under the complex new service oriented architecture.

Both bank cases illustrate how it is difficult to leverage the business benefits from a SOA strategy because aligning business and technology is still difficult with SOA. The alignment difficulty remains because the business community is not automatically included in the new development process. Indeed, the complexity makes it more difficult to include business managers in the development of SOA landscapes because it adds new layers to the incomprehensibility of IT to the untrained. While SOA was applied differently in the two banks, architectural extensibility was a strategic feature that was both sought and achieved in the organisations’ ICT.

SOA adoption approaches are characterized by the two areas of “service evolution” and “service reusability”. We can evaluate the different strategic approaches to SOA adoption on this basis. The black circle in Figure 3 identifies the use of SOA for application integration across the whole application portfolio. In the Northern Europe Bank the SOA infrastructure is extended to the whole of the application portfolio, while in the Central Europe Bank application integration in quadrants 2, 3 and 4 is obtained via Enterprise Architecture Integration.
The key strategic goals guiding the Northern Europe Bank in its introduction of SOA are oriented to value preservation and value reconfiguration; they concern the bank’s core business and, therefore, the applications shown in quadrant 4. On the other hand, the Central Europe Bank SOA adoption is limited to the application area, where the need for both service evolution and service reuse was of greater importance (quadrant 1). This is in line with the Central Europe Bank’s strategic goal of more agile software development. In contrast to Northern Europe Bank, SOA extensibility plays a key strategic role in Central Europe Bank.

8 IMPLICATIONS

The successful development of an extensible SOA by the two banks, along with the rather dissimilar nature of their approaches suggests a number of deeper insights that we can draw from the cases.

Agile systems development is not essential to agile organizational development. The stringent demands for security and reliability in banking systems are not conducive to the use of many of the agile development approaches. These approaches focus on development speed rather than product quality (Baskerville et al., 2001; Baskerville and Pries-Heje, 2004), and as such are typically unacceptable for banking systems. Each of the banks developed extensible, service oriented architectures, and did so for the most part with their usual development strategies while adopting new frameworks and tools. A still unclear evolution of traditional development practices may be under way with the adoption of SOA and Web services, enacting more agile organizational development while attaining high quality, security, and reliability standards.

Service Oriented Architecture is a means rather than an end. The context of both case settings was driven by organizational strategy. The move to the development of SOA at Northern Europe Bank was part of a strategy of aligning business and technology for agility. While something of an experiment, the development of SOA at Central Europe Bank responded to a strategy of innovation at calculated risk with a goal of retaining market leadership. Both organizations approached SOA as a means to achieve their strategies. In this sense, SOA is only one ingredient in a recipe for a much larger meal.

SOA builds affinities for recognized principles of software development. The architecture inclines those developing software within this architecture toward recognized principles of sound and professional software development. SOA is a naturally evolutionary architectural style arising from implementation independence and an extensible, open framework. In the Northern Europe Bank case, they achieved logical independence from the physical implementation of the code, and they realized an unexpected potential to offer services externally. In a different way, Central Europe Bank opened a closed architecture by wrapping Web services around a legacy system, converting physical dependence to a logical independence. These natural outcomes satisfy at least two central principles in software development: (1) the aim of building with and for reuse, and (2) controlling complexity with multiple levels of abstraction (Bourque et al., 2002).
SOA invokes an unfamiliar concept that raises barriers to adoption. The service concept is unfamiliar to many experienced system developers. Like the transition to object-oriented frameworks a decade ago, both Northern Europe Bank and Central Europe Bank encountered steep learning curves. Developers grappled with the essential differences between the fundamental concepts of SOC that lie at the heart of SOA. For example, in Central Europe Bank, a new IS development team was formed as a result of an organizational learning process enacted by first experiences with SOA and Web services. Furthermore, new skills were also required in tuning and managing applications based on SOA.

ACKNOWLEDGEMENT

A preliminary version of the manuscript was presented at the European Conference of Information Systems (ECIS) 2005. The authors would like to acknowledge the Editor of the special issue and two anonymous reviewers for their thoughtful and constructive comments.

REFERENCES


WEBSITES

Figure 1 A Resource-based model of competitive advantage (Mata, Fuerst and Barney 1995, fig.1)
Figure 2  Service Reusability vs. Service Evolution in a typical banking application portfolio
## Central Europe Bank

(SOA based on EAI integration platform+Web services)

<table>
<thead>
<tr>
<th>SOA Strategic Challenge</th>
<th>SOA Potential result (Null, Limited, Medium, High)</th>
<th>SOA Achieved result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application integration</td>
<td><strong>High</strong>: Service Oriented architectures are a step towards modular design of business functionality, providing a method to reduce the complexity of connecting and reconfiguring business functions.</td>
<td><strong>Limited</strong> (compared to the whole application portfolio): SOA application integration model is adopted, at the moment, by the Cash Management System (CMS) and by another recent project (Portfolio Management System). Still, IS development strategy is now heavily based on SOA (see below).</td>
</tr>
<tr>
<td>Value reconfiguration</td>
<td><strong>Medium</strong>: Web services standard architecture provides a more standardized approach to value reconfiguration compared with XML. Drawbacks due to immature technology and overall complexity (Web-services security, performance, management…).</td>
<td><strong>Null</strong>: No actual business based on value reconfiguration (e.g. organizational networks with external banks) at the moment. Indeed the CMS project provided an opportunity to take into account this potential issue for the future. This is in line with the CE Bank conservative, risk adverse attitude and culture.</td>
</tr>
<tr>
<td>Value preservation in M&amp;A</td>
<td><strong>Limited</strong>: Web-services based SOA could facilitate value preservation in Merge &amp; Acquisitions.</td>
<td><strong>Null</strong>: Central Europe Bank has not been active in M&amp;A recently. In the last few years no M&amp;A operations.</td>
</tr>
<tr>
<td>Agile IS development</td>
<td><strong>High</strong>: Enabled by Web-services based SOA, due to extensibility and reusability.</td>
<td><strong>Medium</strong>: IS development practices observed within the CMS project are partly traditional and partly agile. More than half of new internal IS development efforts are now based on SOA and Web services.</td>
</tr>
</tbody>
</table>

*Table 1* Central Europe Bank: SOA potential and achieved results contributing to strategic value generation across the four challenges discussed in Section 2.
<table>
<thead>
<tr>
<th>SOA Strategic Challenge</th>
<th>SOA Potential result</th>
<th>SOA Achieved result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application integration</td>
<td><strong>High</strong>: Potentially no limits to extending Web services based AI to other applications, thanks to SOA</td>
<td><strong>Medium</strong>: More than 1000 operations in production. E.g. the complete customer portal works with a SOA interface to the systems behind.</td>
</tr>
<tr>
<td>Value reconfiguration</td>
<td><strong>Limited</strong>: XML has no standard architecture providing a standardized approach to value reconfiguration.</td>
<td><strong>Null</strong>: No actual business based on value reconfiguration at the moment.</td>
</tr>
<tr>
<td>Value preservation in M&amp;A</td>
<td><strong>High</strong>: Northern Europe Bank active in M&amp;A.</td>
<td><strong>Medium</strong>: The way SOA is implemented allows for better value preservation in M&amp;A. This is actually being demonstrated at the time of writing – after the Northern Europe Bank bought two other banks</td>
</tr>
<tr>
<td>Agile IS development</td>
<td><strong>High</strong>: Developing new systems with existing services has the potential of being very agile. However, XML approaches may have a lower potential for standardized component based reuse than WS approaches.</td>
<td><strong>Medium</strong>: The gradual reconfiguration of the system portfolio towards SOA is providing progressively higher levels of flexibility. The current development model embraces and utilises the potential for agility that SOA provides.</td>
</tr>
</tbody>
</table>

Table 2  *Northern Europe Bank: SOA potential and achieved results contributing to strategic value generation across the four challenges discussed in Section 2.*
SOA adoption strategies

![Diagram showing two factors: Service Evolution and Service Reusability, divided into High and Low categories. The diagram illustrates different strategic approaches to SOA adoption.]

Figure 3: Different strategic approaches to SOA adoption

ENDNOTES

1 This observation is in line with well recognized theoretical views, including structurational approaches, and studies of social actors. For example: "ICTs are intrinsic to the exchanges of organizational entities: they both shape and are shaped by the interaction practices that may proliferate within and among organizations" (Lamb and Kling 2003, p. 216).

2 Jay B. Barney is one of the recognised "fathers" of the resource based view, significantly extending the original contribution by Birger Wernerfelt (Barney, 1991; Wernerfelt, 1984).

3 A promising direction for future studies might be indicated by the three logics of strategy affecting firm performance recently proposed in the IS literature: positioning,
leverage and opportunity (Sambamurthy et al., 2003). The strategic challenges in banking discussed here are in fact connected with these logics. The logic of strategy of "positioning" is affected by the "extent of integration among the activity systems" (challenge of application integration). The logic of strategy of "leverage" is affected by the "ability to create capabilities through integration and reconfiguration of internal and external resources and embedding in firms’ social, structural, and cultural contexts" (challenges of application integration, value reconfiguration and value preservation). The logic of strategy of "opportunity" is affected by the "ability to continuously innovate; ability to develop superior market intelligence; ability to coevolve assets, capabilities, and knowledge" (challenges of application integration, value reconfiguration and agile development) (Sambamurthy et al., 2003; citations are from Table 1, p. 240).

At the time of observation, and given the limited scope of the CMS project in the CE bank, an explicit strategy for systematic discovery of existing services had not yet been adopted. Even the basic UDDI technology of Web services architecture had not been exploited to this aim.