Stephan Aier, Joachim Schelp

How to Preserve Agility in Service Oriented Architectures

An Explorative Analysis

Although companies introduced enterprise application integration (EAI) a couple of years ago, the complexity of corporate application landscapes is ever increasing. The current wave of technology being introduced into these application landscapes are service oriented architectures (SOA). Not unlike EAI before the introduction of these technologies is associated with re-use of software components and reaping cost cutting potentials. But when looking at the still increasing complexity of application landscapes following the introduction of SOA, the re-use and cost cutting arguments lead to disappointment. However, SOA offers a great potential to increase corporate agility. In order to gain and preserve corporate agility it is necessary to explicitly manage enterprise architecture. This paper discusses the problems of re-use and cost cutting expectations in SOA and contrasts them with the potentials related to make sustainable contributions to corporate agility. Structures, processes, and instruments to realise these potentials are discussed with reference to a literature review as well as to selected case studies.

1 Introduction

When complexity of evolutionary grown application landscapes made them unmaintainable, practitioners introduced enterprise application integration (EAI) technologies and (mostly) succeeded in reducing the interconnection complexity by replacing 1:1 by m:n interfaces via a common middleware layer. However, the same cost cutting and re-use potentials are endeavoured to persuade practitioners to introduce service oriented architectures (SOA) nowadays (e.g., Linthicum 2000; Starke and Tilkov 2007). With EAI's first introduction goals like cost cutting could be met, e.g., by standardising and reducing the number of interfaces. Other goals like increasing flexibility – e.g., mentioned in Aier (2004), Kaib (2002), Keller (2002) or by practitioner contributions in Aier and Schönherr (2005) – could not be met. The confrontation with exaggerated expectations has put EAI into the background. However, similar expectations are now connected to SOA as the participants in the underlying longterm study state. Experiences with EAI are bringing re-use and cost cutting potentials into the foreground. Driven by vendors, analysts, and consultants the expectation is rising (again) that the introduction of this technology leads to a better adaptation of corporate IS to changing business processes and to a better business process support in general.

Instead of discussing flexibility and better business/IT alignment in general this contribution offers an agility oriented goal system, in which other goals are positioned. Because agility is to be realised not only once, but continuously, this contribution asks for the structures, processes, and instruments necessary for a sustainable achievement of these goals. Goal system, sustainability and resulting requirements are discussed in Sect. 2. Because only few companies have established the required structures and processes, first an overview of related work in literature is given in Sect. 3, before current practices are outlined by the discussion of selected cases studies in Sect. 4. Section 5 will analyse these case studies regarding sustainability. A summary

and an outlook on further research conclude this contribution in Sect. 6. The discussion of the goal system and its application to SOA are based on a literature analysis. An explorative case analysis according to Yin (2002) is made to discuss the required organisational structures and processes.

2 Agility and sustainability as complements in a corporate goal system

For most industries we can assume that companies face an intense, if not increasing competition. Furthermore, we presume that increasing corporate agility is an important goal in the companies' goal system. Additionally we assume that business is passing these goals down to corporate IS, requiring them to contribute to corporate agility as well. If this is the case, the design of the application landscape has to reflect it either. These assumptions are based on the studies on key issues for IT executives by Luftman et al. (Luftman 2005; Luftman and McLean 2004; Luftman et al. 2006) and the contributions of Weill et al. on strategic alignment and agility (Weill 2004; Weill et al. 2002).

However, an analysis of mature engineering disciplines has shown that the only way to transform complex systems is by division of labour (Shaw 1990). Therefore it is neither desirable nor feasible to constantly challenge an entire organisation. The ability to transform an organisation in an agile way has to be limited to 'reasonable' changes. Therefore we propose the concept of sustainability as a complementary goal to agility.

2.1 Agility

Although agility and flexibility are often used as synonyms, we follow the discussion in production management and understand flexibility as a part of agility. E.g. Yusuf et al. (1999) understand agility as the capability to adapt to unexpected changes, whereas flexibility is focused on expected changes only (Becker 2001; Goldman et al. 1995; Sharifi and Zhang 1999; Vokurka and Fliedner 1998; Yusuf et al. 1999; Zhang and Sharifi 2000). In production management flexibility is 'built-in' by considering configurability in early design stages, both for production structures (e.g., by using highly configurable CIM or CAD/CAM systems) and products (e.g., by component based design). Configurability fosters flexibility, but it cannot (or only very limited) contribute to unexpected changes, because only expected changes could be considered at the design stage. Consequently agility is required as a corporate goal in production management (Becker 2001; Duguay et al. 1997). Yusuf et al. (1999) define agility as 'the successful exploration of competitive bases (speed, flexibility, innovation proactivity, quality and profitability) through the integration of reconfigurable resources and best practices in a knowledgerich environment to provide customer-driven products and services in a fast changing market environment'. Furthermore in production management the importance of being pro-active is stressed instead of being only reactive to the changing business environment (Goldman et al. 1995). The five sub-goals of agility as defined by Yusuf et al. (1999) serve as the evaluation framework for the subsequent discussion of SOA in Sect. 2.3. However, agility is a goal to be reached not only once, but permanently. Therefore the following subsection investigates the concept of sustainability.

2.2 Sustainability

Sustainability may easily be identified as a topic deeply rooted in environmental economics. The most cited definition may be found in the so called Brundtland report: 'Sustainable development seeks to meet the needs and aspirations of the present without compromising the ability to meet those of the future' (World Commission on Environment and Development 1987, p. 43). Other prominent (but analogous) definitions exist as well (Conrad 2000, p. 2; Teichert 2002, p. 16). Based on a broad literature review Huber (1995, p. 39) decomposes sustainability into the strategies of efficiency, sufficiency and

consistency, and Gronau (2003, p. 222) added the strategy of participation.

The *strategy of sufficiency* asks: How much is enough? An answer can hardly be found. However, it seems to be advisable to be modest. Sufficiency raises the demand to realise a certain degree of modesty by renunciation. The criticism of this strategy reads: It is unrealistic and full of false effects – the former, because it contradicts the general standard of individual utility maximisation. The latter, because restrictions can lead to economic and social stagnation or at least to undesirable development within these areas (Huber 1995, p. 40). The *strategy of efficiency* focuses on increasing productivity, to make (economic) services profitable, i.e., to provide services with minimal use of resources, addressing a sub-goal of agility at the same time. Central concepts are reuse and longevity. From an economic point of view the strategy of efficiency is the most compatible strategy; which therefore is often equated with sustainability. The *strategy of consistency* defines either the complete isolation of a system from its environment or the enforcement of its consistency between a system and its environment as the design goal (Huber 1995, p. 41). The *strategy of participation* demands the involvement of concerned stakeholders. This is required both for an optimal design of a system and to increase acceptance of a system within its user base (Gronau 2003, p. 222).

If sustainability is a relevant topic for a company, in the majority of cases reducing negative external effects to its physical and/or social environment are addressed (Leitschuh-Fecht and Steger 2003, p. 259). This will be referred to as *externally oriented sustainability* in the following. Furthermore the underlying understanding of sustainability will be detached from ecological aspects. Hahn and Hungenberg (2001, p. 13) define the main goal of each company as preservation (i.e., sustainability) and successful development, in order to meet the individual expectations of all stakeholder groups. This *internally oriented sustainability* defines a corporate view on sustainability – addressing a long-term, efficient corporate leadership and management as the primary corporate goal, which contributes to agility as well.

Aspects of the *strategy of consistency* can be found within the dimension of organisational architecture. It is obvious that organisational structures and business processes cannot be designed independently from each other, because they are different views of the same object. A lacking consistency between these views leads to inefficiency (Bleicher 1991, p. 42). Analogous consistency between IS systems is required as well as consistency between organisational and IS architectures supporting them. Within the IS department the *strategy of efficiency* demands for a minimal time and effort to implement changes in IS structures. Changes (or extensions) have to be implemented fast and cost efficient. At the same time complexity is not to be increased more than necessary (Hagen 2003, p. 67). Is the overall corporate system to be changed – i.e., by introducing new technologies like web services or new design paradigms like service orientation – the resulting changes have to be reflected in the overall system, i.e., by new organisational structures, rules, or management instruments. To be compliant with the *strategy of sufficiency* at the same time requires all necessary changes to be minimal, not to be larger than required. To master all changes within the organisation, acceptance by all persons being affected is required, leading to their involvement in the transformation process (*strategy of participation*).

When comparing agility and sustainability a conflict between both goals becomes obvious: if all necessary changes are required to be minimal, no additional flexibility gets introduced; there will be no slack for unanticipated change requirements. Therefore agility relies the more on the ability to implement changes fast. Although minimalism of changes fosters their acceptance, the strategy of participation calls for a broader transformation process, which requires

more time. Finally, the strategies of sustainability provide a valuable mindset in order to limit non-reflected decisions allegedly contributing to agility. Service oriented architectures can contribute to a sustainable increase in corporate agility only if they provide positive contributions to time to market, flexibility, pro-active innovation, quality, and profitability as shown in Sect. 2.1 and at the same time being compliant with the strategies of efficiency (profitability), consistency (consistency of architectures), participation (involvement of business and IS departments). The later demands for all measures within the change and transformation process to meet the goals and addressing the issues stemming from the inherent goal conflicts, but not to reverse the goals by introducing redtapism.

2.3 A first assessment of SOA

How can a SOA contribute to corporate agility? According to Yusuf et al. (1999) the sub-goals of agility as mentioned in Sect. 2.1 have to be met: (1) speed and better time to market respectively, (2) flexibility, (3) pro-active innovation, (4) quality, and (5) profitability. The aspect of pro-active innovation is disregarded in this discussion, because it is primarily influenced by organisational issues, less by architecture paradigms. As SOA can be implemented with various technologies – see, e.g., the technologies referenced in Hagen (2003) and Herr et al. (2004) – a detailed view on technologies would be out of scope of this article as well. Furthermore this article will not enter the discussion of 'IS as an enabler'.

Regarding *quality* modern technologies (i.e., tools and frameworks) are enforcing generation of code and documentation more than they did in the past. Additionally newer infrastructure technologies (i.e., middleware) foster the use of repositories, so that manual search for components, documents etc. is reduced, if not avoided. Therefore these repositories and (enhanced) search functions for components and documentation enhance the development process and its quality. They positively contribute to product quality. But due to the use of tools of different vendors ('best of breed'), a general conclusion regarding quality is difficult to make: A potential for a positive contribution is given, but depends on the effective combination of employed tools and their integration to determine the extent, at which this potential can be realised.

Another contribution to agility is offered by SOA by fostering the use of standardised interfaces and harmonising infrastructure interfaces, therefore facilitating the combination of (existing) services and increasing the *flexibility* of the overall system. To use this advantage to its full extent, standardisation and harmonisation of interfaces is strictly required. But even if standardised systems are not available and/or legacy systems are dominating the application landscape, SOA offers advantages in terms of flexibility: Standardisation is facilitated by improving the infrastructure (repositories etc.), offering a better basis for further development with an increased share of standardised interfaces. In our discussion standardisation is improving flexibility by increasing exchangeability in the context of expected changes. For unexpected changes standards might set obstacles higher, because the standard itself has to be changed, which cannot be done in the short run. But this might not necessarily result in an increased reaction time for the individual project, if architectural exceptions (for deviations from the standard) are allowed.

Similarly advantages in terms of speed are unlocked by reducing development time of (enterprise) services and components not only for new components, but for changing existing systems as well: We assume that a business department's request for IS support has led to the development of an (enterprise) service consisting of partly existing, partly newly developed components (basic services). Reuse of existing components has increased the development time of this service first. Then we assume that another business department's request leads to a change of this service. This change request can be implemented in different scenarios: In the simplest scenario

it is a 'black box' change. The service can be changed internally without enforcing changes in dependent services or components. We assume that the test efforts will be limited, because the tests required relate only those services and components directly connected. In this scenario the change can be implemented fast compared to the full change cycles in previous legacy systems.

In a second scenario the change request requires a change of at least one interface of one of the dependent services. This leads to a higher change effort than in the first scenario. However, SOA offers all advantages of object and component orientation: The change can be implemented by simply offering an additional interface via a service variant, which has a different signature. In this case all services depending on the older variant do not have to be changed, as 'their' interface is still working. Other services developed later on and requiring the changed service, simply make use of the new service variant. The advantages regarding development time compared to development in legacy systems is obvious and there is only a small difference compared to the first scenario. Under good circumstances the development simply requires an additional rule within the integration infrastructure (integration bus). If additional logic beyond transformation rules is not allowed on the integration bus by policy, an explicit variant has to be built. This is a realistic scenario: in some cases up to six variants of a service are reported to be used in parallel.

Although the gains in development time and therefore in time to market are obvious, there are some problems related to the increasing number of services, which is fostered by the introduction of variants. In reality reuse of existing services or components is high only for a small number of services; however, the average reuse rate is very low. An impressive example, e.g., Schwinn and Hagen (2006) are giving: They have shown that at the time of the study 34 % of 650 services have been reused, but with an average reuse rate of 1.7 only. Especially the low number of services with a high reuse rate (14 services have been reused 10 times or more; for the yet positive evaluation of SOA see Hagen 2003) is interesting. Every additional variant reduces reuse. Furthermore every additional variant increases maintenance efforts. Therefore the evaluation of SOA regarding *profitability* has to be very cautious, especially if the ROI of a SOA project can be determined. Another disadvantage is the increasing complexity of the overall system, which may have – especially in the long run – a negative impact on quality, time to market, flexibility, pro-active innovation, and profitability. And finally different variants of a service facing different reuse by newly developed services, may lead to a tightly intermeshed net of dependencies, whose complexity is bigger than the legacy system replaced by SOA.

We can conclude that SOA can contribute to higher corporate agility. But it is questionable, if this is sustainable. Especially complexity issues and alignment to business changes seem to require solutions beyond the (technical) design paradigm. In addition the issue of re-use stresses the goal conflict between the strategy of efficiency (as part of sustainability) and the focus on fostering speed of change as a key to agility. To further elaborate on success factors of sustainably changing an organisation in an agile way the next section analyses a subset of current literature on agility and sustainability in the SOA context. Because these success factors may be affected by situational factors and may supersede the direct contribution to agility, further analysis is done via an explorative case study to determine further relevant success factors in Sect. 4.

3 Literature Analysis

In the following section we discuss selected approaches to service orientation and enterprise architecture (EA) management in literature that discuss the goals of agility and sustainability as well as their success factors. Due to limited space we only include approaches which do comprehensively contribute to these aspects

and which are published in larger conferences and selected journals since the year 2000[1]. It has turned out that these topics are usually developed over a period of time and several publications by groups of researchers and therefore constitute language communities (Schelp and Winter 2008b). As a representative for practitioner's approaches we have included the Open Group's TOGAF approach (Dico 2008; The Open Group 2003). Although TOGAF has a limited concept of EA compared to the other approaches analysed, it addresses essential areas of EA and represents the scope of experience of a large number of organisations.

Based on the discussion of integration methodologies in the context of EAI a research group of TU Berlin developed an integration and architecture approach which even though focuses on challenges of integration has an holistic understanding of architectural layers (Aier 2007; Aier and Schönherr 2004, 2006a,b). At EPFL Lausanne the SEAM approach to architecture design has been developed although without an explicit reference to service orientation challenges of alignment and modelling are well addressed (Balabko and Wegmann 2006; Wegmann 2002, 2003; Wegmann et al. 2005). At KTH Stockholm a series of contributions to EA design and development has been developed which also address questions of service orientation (Johnson and Ekstedt 2007; Johnson et al. 2004, 2006; Lindström et al. 2005; Lindström et al. 2006). The same is true for the contributions from Telematica Institute Enschede (Jonkers et al. 2003, 2004, 2006; Lankhorst et al. 2004; Lankhorst 2005). They have not only developed method fragments for design and management of EA but also address SOA. In the context of business engineering the University of St. Gallen developed method fragments for EA management equally addressing management and governance of SOA (Baumöl 2006; Klesse et al. 2005; Österle and Winter 2003; Schelp and Stutz 2007; Schelp and Winter 2007, 2008a,b; Winter and Fischer 2006; Winter and Schelp 2006, 2008).

Service oriented design fosters EA agility. However, the challenge is to sustainably preserve agility. The analysed EA approaches suggest that the alignment of business structures and IT structures is an essential sub goal. To continually achieve this business/IT alignment most approaches focus on transparency as a precondition of consistency. Consistency within EA (e.g., as business/IT alignment) is a fundamental strategy of sustainability (Huber 1995). Most of the analysed approaches therefore developed own notations and tools to generate transparency (Aier et al. 2009; Lankhorst et al. 2004; Wegmann 2002). Some of these tools provide corresponding analyses for the identification of inconsistencies (Aier et al. 2009). The lowest common denominator of all approaches is the promotion of standards to improve consistency. Beside the success factors of transparency and consistency strict governance and appropriate communication within IT and business departments as well as between both of them are identified. Both are vital to achieve significant service reuse and thus sufficiency. Well established communication processes for architectural principles, strategies and design contribute to the strategy of participation (Gronau 2003). Especially the importance of top management support is stressed. Some approaches propose to employ EA as an additional management (information) system (Baumöl 2006; Johnson et al. 2004; Österle and Winter 2003).

Because the identified success factors may be affected by situational factors further analysis is done via an explorative case study to determine further relevant success factors in the next section.

4 Case Studies

In the following subsections the case studies of eight companies are outlined. These companies have introduced SOA several years ago and made experiences with the evolution of these

[1] We have explicitly not included contributions that exclusively focused on technical aspects of SOA.

architectures. In each case the motivation to introduce SOA will be presented, as well as the architectural levels which are distinguished and the structures which have been chosen to design and run these architectures. Architectural levels are outlined because of the diversity of the term service. Comparability is provided by translating the individual companies' terms into the architectural levels defined in (Winter and Fischer 2006): business architecture, organisational architecture (covering business processes and organisational structures), integration architecture, software architecture, and infrastructure architecture. A SOA is to be defined on the integration architecture level, with the term service interpreted as enterprise services (defined from a business perspective), in contrast to a technical understanding of software components (alias software services). The decoupling of services, i.e., designing them self-contained and not dependent on the state or context of other services (see, e.g., Aier et al. 2008; Aier and Winter 2009), is a general requirement to constitute a SOA, which is met by all companies. The discussion of architecture management processes focuses especially on any alignment between these levels and the involvement of the business side.

Data for the case studies have been collected with three of these companies since 2002/2003 and with the remaining five since 2005. We have interviewed key stakeholders in IT management, architecture management (i.e., IS and business architects), and business/IT relationship management have been interviewed. In addition to the interviews regular review meetings have been set up to observe state, development, and architectural issues in the companies involved. Two of the companies described participated in long term collaborative research projects in IS integration and EA involving ten companies in the period of 2002–2008. The companies chosen for this study have a long term experience with SOA and have mature architecture management structures in place (i.e., defined organisational structures and processes, metrics, and governance structures). With two companies individual research projects on integration and architecture have been done. These projects offered the opportunity to contact a broad range of employees and roles to gain additional insight into internal changes in structure, state, and strategy of both IS and business perspectives. With all companies involved agility has been identified as a goal – although the time when it was introduced to IT departments' goal set varied and not all companies identified agility as the dominating business goal. Data presented in the case studies below aggregate the research results gained with these companies until spring 2008. Due to company request the case studies have been made anonymous.

4.1 Company A

Company A is one of the major banks in Switzerland. In its history mergers had a major impact on the complexity of the application landscape. Increasing demand for application integration led to the introduction of SOA. In 2002 the core banking system consisted of more than 450 host based and client server systems. To meet the resulting integration complexity a first SOA vision was developed in 2001. First steps to its implementation have been made by encapsulating existing functionality as business services resp. by implementing new functionality as business services. The encapsulation of existing functionality increased flexibility and implementation time in general could be reduced, i.e., time to market could be improved. Regarding architecture the levels business (i.e., partly business and organisational architecture), application and integration (i.e., both integration architecture, where business services are located), software, component (i.e., both software architecture, where technical services can be identified), and technical architecture (i.e., infrastructure architecture) are distinguished. All technical architectures are more detailed than the business architecture: only a few

business models are modelled explicitly; modelling of business processes is diverse regarding details, scope, completeness, and timeliness depending on the business departments. Architecture management processes are strong regarding the IS department: a team of more than 90 architects guarantees architecture communication and enforcement to IS development. Architecture processes are clearly defined in this context and strong architecture governance is established. Architecture impact regarding the business departments is less strong; although impact and enforcement of architecture is strong regarding individual IS projects. However, architecture models and processes (and resulting transparency) within the business departments are not equivalent. Scope, models, model granularity and quality of processes are more process driven and coincidental.

4.2 Company B

Company B is one of the biggest power supply companies in Germany. Whereas EAI has been understood and operated as a technical project, SOA has been taken differently. SOA governance issues have been a major concern which has early been addressed. Driven by the group IS department governance models have been developed alongside detailed technical models. Business owners have been involved early, resulting in shorter (SOA based) implementation times for new or changed business processes today, thus contributing positively to agility. However, this IS driven SOA initiative has been limited to selected business departments. A reason for this may be the lack of any holistic enterprise architecture: the architecture team in the IS department is focusing on IS architecture management only and business process management is done for selected business departments only. Architectural levels covered are (selected) business processes (i.e., organisational architecture), business service architecture (i.e., integration architecture), basic service architecture (i.e., software architecture). Analogous to company A the focus has been more on technical architectures; whereas business oriented architectures have been rudimentarily covered at best.

4.3 Company C

Company C is a major financial service provider in Switzerland primarily focusing on standardised retail banking and transaction processing. Increasingly complex dependencies between applications within the evolutionary grown application landscape combined with an increasing demand for application integration led to a major EAI project. This resulted in a SOA vision to repeat on business side the advantages gained by faster project execution, realised reuse and resulting cost efficiency within the IS department. Regarding architectural levels all levels mentioned in Winter and Fischer (2006) can be found with broad, defined architecture management processes on IS side to develop and carry on the service oriented architecture. An initiative was started to manage business and organisational architecture artefacts by the IS department architecture team as well. However, this has been dropped in favour of managing all business related artefacts by an explicit business architecture management team itself: All business related architecture artefacts are managed by an organisational unit attached directly to the CEO. The alignment of business and IS architectures is explicit and facilitated by a personal interweavement by having former IS architects included in the business architecture unit. The service oriented application landscape has led to increased reuse and reduction of costs, thus contributing to agility. The effect of better alignment processes could not be evaluated sufficiently by now, because they have not been ‘in production’ long enough. However, first observations indicate a positive contribution.

4.4 Company D

Company D is a major telecommunication service provider in Germany. In the context relevant for this paper the telecommunication industry

features two special characteristics: On the one hand the industry is technology affine and technology driven. On the other hand implementation speed is high regarding both design and implementation of new products (e.g., new pricing models, technological product innovation) to be distinguishable from competitors. Both characteristics have led to an early start of an enterprise architecture project to define a framework for technical change projects. This framework has been considered helpful for both identifying the impacts of scheduled change projects and to govern conformance of planned changes regarding defined architecture rules. Additionally special processes have been established, to break architecture rules temporarily, if both a project plan and a budget are defined to re-establish architecture conformance. These rules have been helpful to gain additional speed advantages, thus contributing positively to agility. In a technology driven industry like telecommunication the cultural distance between business and IS departments is not as large as it is in other – less technology affine or technology driven – industries. To retain this closeness in the long run, all change projects have to show a clear business case. Due to the stiff competition in the market and the resulting change frequencies, even a business case for infrastructure projects like SOA can be shown. The enterprise architecture management in this case has clearly defined structures and processes, as well as the governance has. The company has architectural layers compatible with Winter and Fischer (2006), with special importance given to a domain model (organisational and integration architecture layers), being relevant for all change projects.

4.5 Company E

Company E is another telecommunication service provider operating in Germany, as a subsidiary of a foreign telecommunication group outside the German-speaking countries. Analogous to company D the company and its business departments are technology affine and the business is technology driven. Furthermore the stiff competition defines speed to a (business) goal, dominating design, adoption and execution of new business models and their supporting applications. Just as in company D the application landscape has grown fast over the last couple of years and has reached a high level of complexity (in terms of number of applications/services, interconnections between them, and resulting dependency structures). In previous years applications have been designed and introduced individually in each country the group is present. Later on a centralisation initiative was started to align the formerly distributed and non-uniform architecture activities in the different countries. Due to a change of corporate strategy the cross-country architecture alignment was revoked. Service orientation is a design goal for application development now to tame the ever increasing complexity of the (distributed) heterogeneous application landscape. The understanding of services is compatible to company D with enterprise services defined on the integration architecture layer and basic services defined as (software) components on the software architecture layer. However, business processes are captured inconsistently, hampering the alignment of business and IS architectures with this regard. Due to the weak cross-country alignment of architectures within the group, the resulting architecture can be described as a federally structured application landscape. Within the individual companies the country-specific architecture management structures and processes are mixed: In the German subsidiary structures and processes are mature, architecture competencies are high and the underlying principles of service orientation are well understood. Processes, structures, and rules defining the governance are clearly defined, too. But due to quickly changing corporate strategies and their organisational impact it is difficult to estimate, how these structures and processes can be effective with sustainable results.

4.6 Company F

Company F is an IT service provider for a large banking network in Germany. In its current form the network is the result of several mergers of formerly independent and regionally organised IT service providers. Every formerly independent company had its own evolutionary grown banking solution. However, none of these solutions had a predominant position within the network. Therefore the network decided to implement a new and common system as their core banking solution. The development started in 2002 and was finished in 2005 for the time being. The new system is designed following a service oriented paradigm in its integration and software architecture in order to adapt and to consistently provide the implemented functionality to every partner.

On a business level the enterprise architecture of company F is designed following the network's process reference model. The process reference model serves as a structural blueprint for the design of actual business processes consisting of several steps, e.g., choosing a certain product for a credit application. Single steps of a process are designed as enterprise services. These enterprise services are eventually implemented as software services on a system level in the enterprise architecture.

Each enterprise service, e.g., management of a credit application, may be used in the entire network for a broad range of products. Throughout the network, reuse of enterprise services is explicitly intended. An enterprise service is comprised of a self-contained set of business functionalities and belongs to a specific domain but may also be reused in other domains. Each enterprise service (on integration architecture level) is linked with exactly one software service (i.e., software architecture level) as a technical implementation. However, software services may be called in different contexts which may result in a different behaviour.

4.7 Campany G

Company G is one of the largest insurance companies in Switzerland. They have started their first projects utilising a service oriented software design at the end of the 1990ies with the introduction of web applications which integrated basic functionalities of the host systems. These early projects aimed at providing functionality over internet technology. Effects like reuse occurred rather accidentally. However, the potential of service oriented design has soon been recognised and resulted in standardisation initiatives as well as an embedment in the company's EA framework in order to systematically foster reuse and maintainability of services.

Company G differentiates three layers of service orientation in their architecture: A user-access layer, a process layer and a service layer. The service layer contains business activity services (which can be assigned to the integration architecture level) which call business object services (residing on the software architecture level). Business object services directly access software systems and may run updates on database records for example. The process layer contains business processes, sub-processes as well as detailed workflow definitions. Workflows employ the functional specifications of business activity services but may also access business object services directly. On the top level access to application's graphical user interfaces is designed by employing access services implemented in, e.g., portals.

The variety of possibilities to use the service framework of company G provides a high flexibility and enables the adaptation to a variety of situations. However, it also demands for strong governance in order to preserve the maintainability of such a framework.

4.8 Company H

Company H is a globally operating telecommunications service provider with a large, complex and heterogeneous application landscape. At the

end of the last century, corporate management decided to structure the corporate group into four rather independent divisions representing the four strategic business areas. The new structure reduced the overall complexity by reducing the interdependencies between the four divisions on a business layer as well as on a technology layer by definition. At the same time, however, heterogeneity as well redundancy between the divisions grew as a result of their new independence. This independence resulted in, e.g., inconsistent information about customers where different divisions served the same customer segments with different, division-specific products. Consequently, divisions have been continually integrated again in order to leverage synergies among them.

Company H primarily focuses the definition of enterprise services as a solid and standardised definition of business functionalities. The major goal of this standardisation is to provide a reusable repository of enterprise services in order to enable the flexible and fast definition of new or changed products. Consequently the initial identification of enterprise services will be derived from product definitions, e.g., an availability check for an internet connection. For actual execution enterprise services employ software services which implement necessary functionality, e.g., the measurement of signal quality on the physical wire. The encapsulation of required functionality in enterprise services provides the necessary decoupling of the product layer and the technical software layer.

As a control instance company H has implemented an EA function on group level which addresses the entire stack from business models to questions of low-level technologies (e.g., network infrastructure). The primary mean of alignment therefore is the enterprise service model (also called capability model) which is structured in group-wide domains.

5 Results

As proposed in Sect. 2.3 SOA has the potential to contribute positively to corporate agility. In fact in all cases positive effects could be observed after the introduction of SOA or their underlying design paradigm: Improvements in speed could be observed in every company. The improvements in the technical infrastructure have led to a better (at least technical) quality. Reuse, flexibility, and profitability potentials could be realised only in some, but not in all cases (see Tab. 1). Furthermore in all cases the complexity of the overall system has grown. When considering the long term effects of increasing complexity the positive effects of introducing SOA may show to be temporary. For being sustainable this complexity has to be tamed (Schelp and Winter 2007; Winter and Schelp 2008). Experiences from the case studies outlined give hints on critical success factors.

Company / Contributions to	A	B	C	D	E	F	G	H
Agility in the short run								
- Time to market	+	+	+	+	+	+	+	+
- Reuse	+		+			+	+	+
- Quality	+	+	+	+	+	+	+	+
- Flexibility	+		+	(+)	(+)	+	+	(+)
- Profitability	+		+			+		+
Increase of complexity	y	y	y	y	y	y	y	y
Strategies for Sustainability								
- Consistency	o	o	+	+	+	+	o	+
- Efficiency	o	+	+	+	+	+	+	+
- Sufficiency	+/o	o	(+)	+	+	+	+	(+)
- Participation	o	o	+	o	o	o	o	+
SOA contribution: + positive, o undecided, - negative; y: yes; empty field: not available/applicable; ( ) preliminary observation								

Table 1: Case Analyses: Contributions to Agility

Within all companies measures are established to probe the effects mentioned: E.g., time to market has been measured in terms of (reduced) project time, which is an indicator for flexibility (in

change projects) as well. Quality improvements could be measured in terms of better documentation related data from repositories. Due to varying development process standards and a diverse set of software tools employed measures are heterogeneous among the companies – especially regarding documentation (e.g., documentation rate or change rates). Because of the unsatisfying situation with diverse measures making the effects only partially comparable, we organised joint workshops within the research projects to have the self-assessments peer reviewed by experts from other companies (including those taking part in the individual research projects or in the interview series only). These discussions between experts in a controlled setting revealed inconsistencies, contradictions and additional insights, thus providing a more solid foundation for the analysis.

In the following the cases are summarised regarding the strategies of consistency, participation, sufficiency, and efficiency as introduced in Sect. 2.2.

Regarding *strategy of consistency* company A has to be assessed very cautiously. The organisational architecture in terms of business processes is not managed at the same extent as it is done for integration, software, and infrastructure architectures. Business departments' views are collected via IS projects only. Although the imbalance is identified from both business and IS departments and first measures are discussed, this issue is not solved in the short run. A similar conclusion can be drawn for company B. Although governance and interfaces to business process management have been established for SOA projects, these measures are limited to single issues – the stabilising (and balancing) frame of an (enterprise) architecture management does not exist. By contrast, in companies C, G, H the importance of a consistent modelling (and management of models) across the different architectural layers is not only identified, but addressed by explicit architectural units. The situation for company D is similar: The cultural differences between business and IS departments are limited due to the technology affine industry, and the (enterprise) architecture management is complete and pragmatic, so that a sound base for the deployment of SOA is given; analogous company E (with focus on Germany), although the architectural scopes are limited in comparison with D.

A similar picture can be drawn regarding the *strategy of participation*. In company A business department representatives are involved in a limited number and scope only. They initiate IS projects, but are not involved in the (further) development of architectures or in the alignment process of business and IS architectural views. In company B participation is well developed regarding the IS department driven SOA governance model, but is limited regarding selected business departments. In contrast to other cases in companies C, H the participation of business representatives is maximal, due to the positioning of the enterprise architecture team on the business side close to the CEO. Here the potential for contribution to a sustainable alignment is obvious. With limitations this is visible in companies D and E as well, as the cultural gap between business and IS departments is smaller and the cooperation of business, IS, and architectural units is given. This is fostered by technological issues being relevant to corporate product development and marketing on the one hand and the requirement to show business cases for every aspect in the IS departments on the other hand.

Regarding *strategy of sufficiency* company A has implemented a wide range of measures for architecture management, especially architecture communication and enforcement. To ensure consistency among all IS related architectural views this is very helpful. But these measures may lead to a size of the architectural unit and a level of detail in architecture management processes and structures that might lead to over-bureaucracy and limited consistency and alignment in the long run. With company B the state of architecture management is not sufficient enough, to

anchor service orientation in piloting projects in the overall group. In companies C, H sufficiency cannot be evaluated yet, because the implemented architecture management structures are not productive long enough. With companies D and E (with focus on Germany) the sufficiency is best, because even SOA projects have to show a clear and verifiable benefit for the business. Pure SOA projects for architectural purposes do not occur there.

According to Sect. 2.2 a positive contribution of SOA to the *strategy of efficiency* can be assumed, if re-use could be established to heal the inherent goal conflicts – at least partly. This can be confirmed by the experiences in companies B, C, D, E, F, G and H. Especially in company C efficiency potentials have been realised by reuse. Decoupling the systems has led to increased complexity, but the systems are more flexible and integration is easier now. With companies D and E SOA projects have to lead to more efficient solutions, otherwise the constraint to provide business cases would prevent their implementation.

While SOA is a design paradigm, it does not contribute to sustainable agility. Those companies where indicators for contributions to sustainable agility could be found, showed structures, processes, and instruments (i.e., policies, measures, and metrics), that are characteristic features of an explicit enterprise architecture management. A complementary study (Winter and Schelp 2008) indicates, that especially in enterprise architecture management involving business departments in both structures and processes is a crucial success factor to sustainability. The situation at company B illustrates that focusing on the IS departments' perspective is not sufficient.

6 Summary and Outlook

Our study shows that SOA offers the potential to contribute to an increased corporate agility. However, not in every case this design paradigm does offer the reuse and cost cutting potentials as claimed by vendors and consultants. But it enables companies to react faster with their IS projects to (changing) change requests by business departments and thus contribute to overall corporate agility. This is complemented by the potential of higher flexibility and reuse of existing components, although these potentials only are to be realised in the long run to pay off – mostly. As positive the contribution to corporate agility is, as difficult is the communication task of the IS department: cost cutting by reuse is easy to communicate and required infrastructure investments are (comparatively) easy to achieve, but their benefits can only be reaped in the long run. Positive contributions to agility and required infrastructure investments are difficult to communicate; they have to meet an explicit demand of business departments for a better time to market, quality, innovation capability, or flexibility, which is explicit and strong enough to be endorsed with the necessary will to invest.

From a research perspective SOA adoption in companies shows further research potential beyond the discussions of technical artefacts. The cases shown document an increasing inclination of companies to invest in (enterprise) architecture management to meet the increasing complexity coupled with the introduction of SOA. Most of these activities are focused on IS architectures by now, but the requirement to address artefacts within the broad range of EA is clearly visible. A complementary study in Winter and Schelp (2008) shows the emergence of suitable structures and processes to govern architecture development and enforcement. It supports the impression gained from these cases: Especially SOA has to be governed strictly. But setting up a governance structure from a technical perspective only would be limited to a short term perspective. The evolution of the service landscape has to be governed from IT departments and business perspectives as well. Relevant IT government perspectives include production (build and run) metrics. From a corporate governance

perspective the overall architectural evolution has to reflect business goals. The inclusion of business departments in EA activities is a phenomenon showing a broad range of challenges and is very interesting from a research point of view, e.g., which artefacts are suitable for both business and IT perspectives, which architectural capabilities foster the acceptance of EA on the business side, government aspects mentioned above, or how can EA contribute to a better business/IT alignment? Beyond technical questions especially service orientation requires the construction of appropriate methods to foster a sustainable alignment between business and IS departments.

As mentioned in the introduction of Sect. 4 the company data, especially the measures employed, the repository reports used, and project documentation standards illustrated that the measurement and evaluation of SOA is still in its infancy. Therefore we started a research project dedicated to the establishment of a proper SOA and EA government framework, which requires reformulating the measurement of structures and processes. The measures will be derived from the corporate goal system with special emphasis on agility as laid out in Sect. 2. The resulting metrics will allow statistical testing within the company. Within the joint research project the other companies involved will (partially) apply the resulting measures and metrics, allowing both intra-organisational analysis and the derivation of causal models based on a detailed statistical foundation.

References

Aier S. (2004) Sustainability of Enterprise Architecture and EAI. In: Soliman K. S. (ed.) Information Technology and Organizations in the 21st Century: Challenges & Solutions. Proceedings of The 2004 International Business Information Management Conference. IBIMA, pp. 182–189

Aier S. (2007) Integrationstechnologien als Basis einer nachhaltigen Unternehmensarchitektur – Abhängigkeiten zwischen Organisation und Informationstechnologie. Enterprise Architecture Vol. 6. Gito, Berlin

Aier S., Schönherr M. (2004) Flexibilisierung von Organisations- und IT-Architekturen durch EAI. In: Aier S., Schönherr M. (eds.) Enterprise Application Integration – Flexibilisierung komplexer Unternehmensarchitekturen. Gito, Berlin, pp. 1–59

Aier S., Schönherr M. (2005) EAI als integrierendes Element einer nachhaltigen Unternehmensarchitektur. In: Aier S., Schönherr M. (eds.) Unternehmensarchitekturen und Systemintegration. Gito, Berlin, pp. 4–56

Aier S., Schönherr M. (2006a) Evaluating Integration Architectures – A Scenario-Based Evaluation of Integration Technologies. In: Draheim D., Weber G. (eds.) Trends in Enterprise Application Architecture: VLDB Workshop, TEAA 2005, August 28. Lecture Notes in Computer Science. Springer, Trondheim, pp. 2–14

Aier S., Schönherr M. (2006b) Status Quo geschäftsprozessorientierter Architekturintegration. In: WIRTSCHAFTSINFORMATIK 48(3), pp. 188–197

Aier S., Winter R. (2009) Virtual Decoupling for IT/Business Alignment – Conceptual Foundations, Architecture Design and Implementation Example. In: Business & Information Systems Engineering 51(2), pp. 150–163

Aier S., Discher S., Saat J., Scholl U., Winter R. (2008) Economic Justification of Service-Oriented Architecture (SOA) – Research Study: Experiences and Guidelines on Building SOA Business Cases

Aier S., Kurpjuweit S., Saat J., Winter R. (2009) Business Engineering Navigator – A 'Business to IT' Approach to Enterprise Architecture Management. In: Bernard S., Doucet G., Gøtze J., Saha P. (eds.) Coherency Management – Architecting the Enterprise for Alignment, Agility, and Assurance. Author House, Bloomington, pp. 77–98

Balabko P., Wegmann A. (2006) Systemic classification of concern-based design methods in the context of enterprise architecture. In: Information Systems Frontiers 8(2), pp. 115–131

Baumöl U. (2006) Methodenkonstruktion für das Business-IT-Alignment. In: WIRTSCHAFTSINFORMATIK 48(5), pp. 314–322

Becker F. (2001) Organisational agility and the knowledge infrastructure. In: Journal of Corporate Real Estate 3(1), pp. 28–37

Bleicher K. (1991) Organisation: Strategien, Strukturen, Kulturen, 2nd ed. Gabler, Wiesbaden

Conrad J. (2000) Nachhaltige Entwicklung – einige begriffliche Präzisierungen oder der heroische Versuch einen Pudding an die Wand zu nageln, FFU-report 00-07. http://www.fu-berlin.de/ffu/download/rep_00-07.PDF. Last Access: 2007-08-09

Dico A. (2008) Delivering SOA with TOGAF. The Open Group. San Francisco. https://www.opengroup.org/projects/soa-togaf/uploads/40/15844/dico.pdf

Duguay C. R., Landry S., Pasin F. (1997) From mass production to flexible/agile production. In: International Journal of Operations & Production Management 17(12), pp. 1183–1195

Goldman S. L., Nagel R. N., Preiss K. (1995) Agile competitors and virtual organizations: strategies for enriching the customer. Van Nostrand Reinhold, New York

Gronau N. (2003) Wandlungsfähige Informationssystemarchitekturen: Nachhaltigkeit bei organisatorischem Wandel. Gito, Berlin

Hagen C. (2003) Integrationsarchitektur der Credit Suisse. In: Aier S., Schönherr M. (eds.) Enterprise Application Integration – Flexibilisierung komplexer Unternehmensarchitekturen. GITO, Berlin, pp. 61–81

Hahn D., Hungenberg H. (2001) PuK – Wertorientierte Controllingkonzepte, 6th ed. Gabler, Wiesbaden

Herr M., Bath U., Koschel A. (2004) Implementation of a Service Oriented Architecture at Deutsche Post MAIL. In: Zhang L.-J., Jeckle M. (eds.) European Conference on Web Services (ECOWS 2004), September 27–30. Springer, Erfurt, pp. 227–238

Huber J. (1995) Nachhaltige Entwicklung durch Suffizienz, Effizienz und Konsistenz. In: Fritz P., Huber J., Levi H. W. (eds.) Nachhaltigkeit in naturwissenschaftlicher und sozialwissenschaftlicher Perspektive. Hirzel, Wissenschaftliche Verlagsgesellschaft, Stuttgart, pp. 31–46

Johnson P., Ekstedt M. (2007) Enterprise Architecture – Models and Analyses for Information Systems Decision Making. Studentlitteratur, Pozkal

Johnson P., Ekstedt M., Silva E., Plazaola L. (2004) Using Enterprise Architecture for CIO Decision-Making: On the Importance of Theory. In: Proceedings of the Second Annual Conference on Systems Engineering Research, pp. 1–10

Johnson P., Lagerström R., Närman P., Simonsson M. (2006) Extended Influence Diagrams for Enterprise Architectural Analysis. In: Proceedings of the 10th IEEE International Annual Enterprise Distributed Object Computing Conference, pp. 1–10

Jonkers H., van Buuren R., Arbab F., de Boer F., Bonsangue M., Bosma H., ter Doest H., Groenewegen L., Scholten J. G., Hoppenbrouwers S., Iacob M.-E., Janssen W., Lankhorst M., van Leeuwen D., Proper E., Stam A., van der Torre L., van Zanten G. V. (2003) Towards a Language for Coherent Enterprise Architecture Descriptions. In: EDOC '03: Proceedings of the 7th International Conference on Enterprise Distributed Object Computing. IEEE Computer Society, Washington, p. 28

Jonkers H., Lankhorst M., van Buuren R., Hoppenbrouwers S., Bonsangue M., van der Torre L. (2004) Concepts for Modelling Enterprise Architectures. In: International Journal of Cooperative Information Systems 13(3), pp. 257–287

Jonkers H., Lankhorst M. M., ter Doest H. W. L., Arbab F., Bosma H., Wieringa R. J. (2006) En-

terprise architecture: Management tool and blueprint for the organisation. In: Information Systems Frontiers 8(2), pp. 63–66

Kaib M. (2002) Enterprise Application Integration – Grundlagen, Integrationsprodukte, Anwendungsbeispiele. DUV, Wiesbaden

Keller W. (2002) Enterprise Application Integration. dpunkt, Heidelberg

Klesse M., Wortmann F., Schelp J. (2005) Erfolgsfaktoren der Applikationsintegration. In: WIRTSCHAFTSINFORMATIK 47(4), pp. 259–267

Lankhorst M. M., van Buuren R., van Leeuwen D., Jonkers H., ter Doest H. (2004) Enterprise Architecture Modelling – The Issue of Integration. In: Advanced Engineering Informatics 18(4), pp. 205–216

Lankhorst M. (2005) Enterprise Architecture at Work: Modelling, Communication and Analysis. Springer, Berlin

Leitschuh-Fecht H., Steger U. (2003) Wie wird Nachhaltigkeit für Unternehmen attraktiv? – Business Case für nachhaltige Unternehmensentwicklung. In: Linne G., Schwarz M. (eds.) Handbuch Nachhaltige Entwicklung. Leske + Budrich, Opladen, pp. 257–266

Lindström Å., Gammelgård M., Simonsson M., Jonsson N. (2005) A Method to Assess the Enterprise-Wide IT Resources for Performance and Investment Justifications. In: Proceedings from the CSER Conference on System Engineering Research

Lindström L., Johnson P., Johansson E., Ekstedt M., Simonsson M. (2006) A survey on CIO concerns – do enterprise architecture frameworks support them? In: Information Systems Frontiers 8(2), pp. 81–90

Linthicum D. S. (2000) Enterprise Application Integration. AWL Direct Sales, Reading

Luftman J. N. (2005) Key Issues for IT Executives 2004. In: MIS Quaterly Executive 4(2), pp. 269–285

Luftman J. N., McLean E. R. (2004) Key Issues for IT Executives. In: MIS Quaterly Executive 3(2), pp. 89–104

Luftman J. N., Kempaiah R., Nash E. (2006) Key Issues for IT Executives 2005. In: MIS Quaterly Executive 5(2), pp. 81–99

Österle H., Winter R. (2003) Business Engineering, 2nd ed. Springer, Berlin

Schelp J., Stutz M. (2007) A Balanced Scorecard Approach to Measure the Value of Enterprise Architecture. In: Journal of Enterprise Architecture 3(4), pp. 8–14

Schelp J., Winter R. (2007) Towards a Methodology for Service Construction. In: Sprague R. H. (ed.) Proceedings of the 40th Annual Hawaii International Conference on Systems Sciences (HICSS'07), Jan 3–6. IEEE Computer Society Press, Los Alamitos, 64a (1–7)

Schelp J., Winter R. (2008a) Entwurf von Anwendungssystemen und Entwurf von Enterprise Services – Ähnlichkeiten und Unterschiede. In: WIRTSCHAFTSINFORMATIK 50(1), pp. 6–15

Schelp J., Winter R. (2008b) On the Interplay of Design Research and Behavioral Research – A Language Community Perspective. In: Vaishanvi V., Baskerville R. (eds.) Proceedings of the Third International Conference on Design Science Research in Information Systems and Technology (DESRIST2008), May 7–9. Georgia State University, Atlanta, Georgia, pp. 79–92

Schwinn A., Hagen C. (2006) Measured Integration – Metriken für die Integrationsarchitektur. In: Schelp J., Winter R. (eds.) Integrationsmanagement. Springer, Berlin, pp. 267–292

Sharifi H., Zhang Z. (1999) A methodology for achieving agility in manufacturing organisations: An introduction. In: International Journal of Production Economics 62(1–2), pp. 7–22

Shaw M. (1990) Prospects for an Engineering Discipline of Software. In: IEEE Software 7(6), pp. 15–24

Starke G., Tilkov S. (2007) SOA-Expertenwissen – Methoden, Konzepte und Praxis serviceorientierter Architekturen. dpunkt, Heidelberg

Teichert V. (2002) Indikatoren zur Lokalen

Agenda 21: ein Modellprojekt in sechzehn Kommunen. Indikatoren und Nachhaltigkeit Vol. 1. Leske + Budrich, Opladen

The Open Group (2003) TOGAF (The Open Group Architecture Framework) Version 8.1 'Enterprise Edition'. http://www.opengroup.org/architecture/togaf8-doc/arch/

Vokurka R. J., Fliedner G. (1998) The journey toward agility. In: Industrial Management & Data Systems 98(4), pp. 165–171

Wegmann A. (2002) The Systemic Enterprise Architecture Methodology (SEAM) – Business and IT Alignment for Competitiveness. Technical Reports in Computer and Communication Sciences 200265. École Polytechnique Fédérale de Lausanne, Faculté I&C, School of Computer and Communication Sciences

Wegmann A. (2003) On the systemic enterprise architecture methodology (SEAM). In: Proceedings of the 5th International Conference on Enterprise Information Systems 2003 (ICEIS 2003), Apr 23–26. Angers

Wegmann A., Regev G., Loison B. (2005) Business and IT Alignment with SEAM. In: Cox K., Dubois E., Pigneur Y., Bleistein S. J., Verner J., Davis A. M., Wieringa R. (eds.) Proceedings of the 1st International Workshop on Requirements Engineering for Business Need, and IT Alignment (REBNITA2005), Augt 29–30. University of New South Wales Press, Paris, pp. 74–84

Weill P. (2004) Don't Just Lead, Govern: How Top-Performing Firms Govern IT. In: MIS Quaterly Executive 3(1), pp. 1–17

Weill P., Subramani M., Broadbent M. (2002) IT Infrastructure for Strategic Agility. 4235-02. Massachusetts Institute of Technology (MIT), Sloan School of Management

Winter R., Fischer R. (2006) Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. In: Hung P. C. K. (ed.) EDOCW'06 Workshop on Trends in Enterprise Architecture Research (TEAR 2006). IEEE Computer Society Press, Los Alamitos, pp. 30–37

Winter R., Schelp J. (2006) Integrationsmanagement – Notwendigkeit und Gestaltung aus Controlling-Sicht. In: Zeitschrift für Controlling & Management 50(2), pp. 22–32

Winter R., Schelp J. (2008) Enterprise Architecture Governance: The Need for a Business-to-IT Approach. In: Liebrock L. M. (ed.) Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC2008), Mar 16–20. ACM Press, Fortaleza, Ceará, Brazil, pp. 548–552

World Commission on Environment and Development (1987) Our common future. Oxford University Press, Oxford

Yin R. K. (2002) Case Study Research. Design and Methods, 3rd ed. Applied Social Research Methods Series Vol. 5. Sage Publications, London

Yusuf Y. Y., Sarhadi M., Gunasekaran A. (1999) Agile Manufacturing: The Drivers, Concepts and Attributes. In: International Journal of Production Economics 62(1–2), pp. 33–43

Zhang Z., Sharifi H. (2000) A methodology for achieving agility in manufacturing organisations. In: International Journal of Operations & Production Management 20(4), pp. 496–512

Stephan Aier, Joachim Schelp

Institute of Information Management
University of St. Gallen
Müller-Friedberg-Strasse 8
CH-9000 St. Gallen
Switzerland
{stephan.aier | joachim.schelp}@unisg.ch