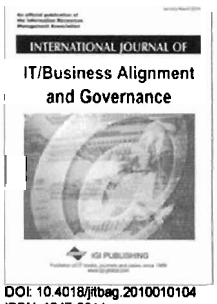
Structuring SOA Governance



DOI: 10.4018/jitbag.2010010104
ISSN: 1947-9611
EISSN: 1947-962X

Author(s)/Editor(s): Michael Niemann (Technische Universität Darmstadt, Germany); André Miede (Technische Universität Darmstadt, Germany); Wolfgang Johannsen (Technische Universität Darmstadt, Germany); Nicolas Repp (Technische Universität Darmstadt, Germany); Ralf Steinmetz (Technische Universität Darmstadt, Germany)

Pages: 58-75 pp.
Source Title: International Journal of IT/Business Alignment and Governance (IJITBAG)
Editor(s)-in-Chief: Wim Van Grembergen (Ed.) (University of Antwerp Management School, University of Antwerp, Belgium); Steven De Haes (Ed.) (University of Antwerp Management School, University of Antwerp, Belgium)
Volume: 1
Issue: 1

Abstract

Companies' IT Systems are confronted with constantly changing market conditions, new competitive threats and a growing number of legal regulations. The service-oriented architecture (SOA) paradigm provides a promising way to address these challenges at the level of a company's IT infrastructure. These challenges, as well as the management of the newly introduced complexity and heterogeneity, are targeted by SOA Governance approaches. In recent years, a number of concrete frameworks for SOA Governance addressing these issues have been proposed. There is no holistic approach considering all proposed elements, consolidating them in order to form a universally applicable model. In this contribution, we motivate SOA Governance, investigate and compare different approaches, identify common concepts, and derive a generic model for governance of Service-oriented Architectures.

[NMJ+10]
Michael Niemann, Andre Miede, Wolfgang Johannsen, Nicolas Repp, Ralf Steinmetz: *Structuring SOA Governance*. In: *International Journal of IT/Business Alignment and Governance (IJITBAG)*, vol. 1, no. 1, p. 58-75, January 2010. ISSN 1947-9611. http://www.igi-global.com/Bookstore/Article.aspx?TitleId=40958.

Structuring SOA Governance

Michael Niemann, Technische Universität Darmstadt, Germany

André Miede, Technische Universität Darmstadt, Germany

Wolfgang Johannsen, Technische Universität Darmstadt, Germany

Nicolas Repp, Technische Universität Darmstadt, Germany

Ralf Steinmetz, Technische Universität Darmstadt, Germany

ABSTRACT

Companies' IT Systems are confronted with constantly changing market conditions, new competitive threats and a growing number of legal regulations. The service-oriented architecture (SOA) paradigm provides a promising way to address these challenges at the level of a company's IT infrastructure. These challenges, as well as the management of the newly introduced complexity and heterogeneity, are targeted by SOA Governance approaches. In recent years, a number of concrete frameworks for SOA Governance addressing these issues have been proposed. There is no holistic approach considering all proposed elements, consolidating them in order to form a universally applicable model. In this contribution, we motivate SOA Governance, investigate and compare different approaches, identify common concepts, and derive a generic model for governance of Service-oriented Architectures.

Keywords: *IT Governance, Service-oriented Architectures, SOA Governance, Strategic IT Management*

1. INTRODUCTION

In recent years, governance approaches for Service-oriented Architectures (SOA) have been intensively discussed. Due to the complexity and heterogeneity of SOA systems, governance is considered crucial to successful long-time operation and control of a SOA. However, a consensus concerning a uniform approach has not been achieved yet. This article gives an overview of current proposals, and introduces a first approach to structuring SOA Governance.

The remainder of this contribution is structured as follows. After a short introduction on SOA and its characteristics as a background, we outline the motivation for SOA Governance and discuss new challenges of SOA systems concerning control and supervision. Next, we provide a comparing literature review of SOA Governance approaches, identify major common concepts, and introduce a generic governance model for SOA. A summary and outlook on future work concludes the article.

DOI: 10.4018/jitbag.2010120404

1.1 Service-Oriented Architectures

Today, we live in a highly competitive and globally distributed economy. As a result, modern enterprises face additional requirements which affect existing and future enterprise information technology (IT) architectures, with the following two being very important (Josuttis, 2007; Krafzig, Banke, & Slama, 2004; Newcomer & Lomow, 2004):

- Achieving a high agility of business processes and their underlying IT.
- The capability of integrating heterogeneous systems.

These are particularly important, as flexible IT systems are needed to support dynamic business processes, which are subject to rapid changes. As companies merge or increase cooperation, it becomes crucial to integrate both various heterogeneous legacy systems and different systems of business partners.

However, many enterprise software solutions in use do not address these requirements, as continuous changes seriously affect the systems' ability to adapt. In addition, enterprise IT appears as a very special field, as, unlike many other domains of IT, enterprise software is developed and maintained in very close collaboration with the end customer, where usually multiple and very different departments are involved. Here, highly political scenarios and very heterogeneous teams face a multitude of requirements, many of which are either coming into conflict with each other, are unclear, or both. However, the challenge is less of a technical nature than an organizational one (Krafzig et al., 2004). Due to changing business models, mergers, and acquisitions, many EAs could not be realized as they were planned in advance and rather grew organically into their current state over time. This usually results in a vertically organized architecture with a so-called pillar or silo structure. These are quite sophisticated and particularly suit the support of operational sequences in their domain (Melzer, 2007). Difficulties and even serious problems arise if this structure has to be modified significantly. Common side effects include data redundancy and multiple implementations of the same functionality in different places. A reason for these silos is the fact that many IT systems used to serve only a single department or business unit—something true even until 1990. This raised the well-known issue of integration, which has challenged IT departments for decades (Newcomer & Lomow, 2004). It is a further example for the need of tight coupling of an enterprise's business to the underlying IT. Although it is more of a technical problem in the end, the main reasons behind integration can be found on the business side. Key business drivers include but are not limited to the following (Krafzig et al., 2004):

- Mergers and acquisitions,
- Internal reorganization,
- System consolidation,
- New business regulations,
- Compliance with new government regulations, and
- Streamlining business processes.

Within this context, the introduction of new software—maybe even across department borders—usually causes huge problems which can outweigh the actual advantages of integrated systems. To resolve the challenges discussed above, the Service-oriented Architecture (SOA) paradigm allows enterprise IT to be aligned with business processes and to make the technical infrastructure flexible enough for quick and continuous changes (Papazoglou, 2003; Newcomer & Lomow, 2004). This is achieved by SOA's focus on describing business problems and decoupling these descriptions from specific implementation technologies (Newcomer & Lomow, 2004). As it is independent of any specific technology, it provides a high level concept for designing IT architectures (Krafzig et al., 2004).

According to Melzer (2007), the main attributes of a SOA include the following:

- Loose coupling,
- Dynamic binding,
- A service repository, and
- Using open standards.

These are necessary to achieve the ambitious goal of separating interfaces from their implementations (Newcomer & Lomow, 2004). Within standard literature, several different definitions for a SOA exist. We prefer the definition by Melzer et al. due to its completeness and conciseness (Melzer, 2007, translated from German): "A SOA is a system architecture that presents manifold, different, and possibly incompatible methods or applications as reusable and openly accessible services to enable a platform and language independent use and reuse."

Although a SOA is rather business-driven, benefits of its application can be found both on the business and the technical side, as shown in Table 1 (Newcomer & Lomow, 2004).

At the heart of SOA is the concept of a service—still being an actively discussed concept—which is generally to be understood as the technological representation of business functionality (Krafzig et al., 2004). By using services as building blocks, business processes can be composed from them, abstracting the processes from the underlying (monolithic) applications and allowing for compositions even across organizational boundaries.

An important goal for any business is to have proper alignment between the services it provides, and the underlying IT infrastructure. In the case of a SOA, a service offers concrete benefits to the business itself by providing access to a high level business concept in the form of business processes. For the service consumer it is not necessary to know how his requests are fulfilled and the service can be viewed as a black box. This aims at making it easy to modify or exchange a service while maintaining its expected or required output (Krafzig et al., 2004).

In order to define the architectural part of SOA, we make use of the following principles (Channabasavaiah, 2003):

All functions (e.g., business functions) are defined as services.
All services are independent and can be used without paying attention to the actual implementation.
- Services can be accessed by an invokable interface without any knowledge of its location.

Accordingly, a SOA is "an application architecture within which all functions are defined as independent services with well-defined invokable interfaces which can be called in defined sequences to form business processes" (Channabasavaiah, 2003).

1.2 New Challenges

According to a recent survey conducted among companies that use SOA as enterprise architecture, 79% stated that they feel a large negative risk by putting services into production which are not effectively "governed". On top of that,

Table 1. Business and technical benefits of a SOA

Business Benefits	*Technical Benefits*
Increase of business agility	Efficient development
Reduced integration costs	Simplified maintenance
Better business alignment	Easier reuse
	Graceful evolution
	Incremental adaptation

88% of the companies consider their current SOA Governance approach as not sufficient—only 12% implemented a sufficient approach according to their own estimation (WebLayers, 2007).

This draws the picture of an extreme disaccord. Although companies are aware of the high risk of a lack of governance, they have not installed sufficient mechanisms to address it. The need for appropriate governance approaches is high – *and* companies are aware of the emerging risk. Apparently, companies have not been supplied with a satisfying approach. Approaches by software vendors often suffer from a "narrow view" on the topic, i.e., from the emphasized focus on the abilities of their own software products (Allen, 2008).

Basically, the SOA paradigm describes a way to realize agile implementations of business structures being able to flexibly adjust to changing environments. Typical "promises" are increased code reuse, reduced integration expense, better security, greater business agility, and a shorter realization time (Windley, 2006, cf. Table 1).

With a SOA, the number of flexible parts of enterprise architecture increases. The price of faster and more flexible adoption of application landscapes to changing requirements on business side is an increasing complexity. This raises the violation probability of business and technical rules or guidelines. As complexity increases, keeping an acceptable level of overview and control of dependencies between services becomes difficult (Kalex, 2007). Variants of services, for example, reduce the overall service reuse rate. Efforts to design a reusable service are estimated up to three times higher than to not do so (Schelp & Stutz, 2007). Further challenges are growth control of the SOA, avoidance of inefficiencies (by design guidelines, implementation standards, controlling mechanisms, and so on), the management of the new heterogeneity, and compliance: governance is to ensure that the SOA system complies with "all applicable regulatory, competitive, operational, and other baseline requirements" (Kobielus, 2006; Schelp & Stutz, 2007). Without appropriate control structures, the SOA-inherent complexity can lead to structures, whose maintenance might be similarly extensive as the one of the legacy application landscape that is to be replaced by SOA (Schelp & Stutz, 2007). Or, in other words, SOA becomes "a mess waiting to happen" (Kobielus, 2006).

Thus, SOA introduces new challenges. The homogenization and control of this emerging complexity is the central challenge to a SOA Governance approach. In order not to be overwhelmed by this new complexity, more organizational discipline is required (Windley, 2006). Transparency and conformance of SOA System have to be ensured – a holistic management approach is needed.

In the discussion on the classification of SOA Governance in the scope of corporate governance, most authors agree that it is a subset (WebMethods, 2006; Keller, 2007), extension (Holley, Palistrant, & Graham, 2006; Woolf, 2006) or specialization (Schelp & Stutz, 2007) of IT Governance. Although SOA Governance addresses special SOA-related issues, such as service ownership or cross-company service deployment, it is still a part of the IT in an enterprise. Hence, IT Governance mechanisms apply to a SOA (Kobielus, 2006; Manes, 2005).

Concluding, due to the new system complexity and its multitude of potential threats, a new form of holistic management—SOA Governance—is required. This approach is to address conformance achievement, avoidance of new heterogeneity, and preservation of the ability to be controlled, and profits from experiences made in the context of IT Governance.

In this context we examine what techniques are used by a number of approaches to SOA Governance, and provide a comparison and structure.

2. EXAMINATION OF SOA GOVERNANCE APPROACHES

During the last years, a multitude of models and frameworks for SOA Governance were proposed. In this section, we provide a com-

parison of these approaches, extending previous work (Niemann et al., 2008). As findings of this literature review, we identify the obviously most important concepts of approaches to SOA Governance.

2.1 Literature Review

A multitude of different approaches and proposals concerning SOA Governance exists. Most of them proceed from different challenges and definitions, but address similar goals. The majority defines completely differing instruments and techniques to reach these goals. Obviously, only few accepted standards and procedures, goals and techniques for use in governance environments exist. One reason might be the fact that there is no common definition of SOA Governance that could form a foundation for the different methodologies.

In this section, we present a literature review. We investigated and compared a number of SOA Governance approaches developed by the research community and the software and consulting industry. During this analysis, we identified concepts of SOA Governance being considered most important by the respective authors. As a result, we present the identified concepts in section 2.2.

Figure 1 gives an overview of our analysis, the rows showing the approaches with authors and the columns the deployed concepts. Concerning the assessment, *proposed and integrated* (•) stands for a sufficiently explained and motivated concept as part of an approach. Some approaches, however, are characterized by a narrow view of the governance problem field. This leads to lack of an actual clear instantiation, level of details, or specification. In these cases, the according concepts are marked *partially integrated* (◦) in the table. A hyphen (-) indicates that the according concept is not integrated in the current approach.

Frequent concepts identified during the approach are the following (cf. Figure 1). *Impact on organization* represents the need to change existing organizational structures in a company. This often includes the introduction of *new roles and accountabilities*, as well as the necessity to influence the employees' *behavior* concerning the "new" system. *SOA maturity models* provide mechanisms to assess a SOA concerning its progress and maturity. *Best practices* and *metric models* are often used when specifying governance *policy catalogs*. *SOA lifecycles* and *SOA roadmaps* are defined to designate future developments of a SOA system. They differ in structure (cyclic vs. linear). The *service lifecycle* describes a service's phases from plan to realization, similar to the common software lifecycle. *Governance processes* define the actual business and IT-internal processes needed to perform and operate an governance approach. By the term *policy enforcement mechanisms* all proposals concerning automated governance compliance checks are summarized. These concepts will be discussed in detail in the sections 2.2 and 3. Below, we outline the single approaches.

As main instruments of their SOA Governance approach, Brauer and Kline (2005), HP Labs and Systinet, mention the business service registry and business service management. They define a five-stages-service lifecycle and a detailed SOA roadmap that shows elements of a maturity model. In general, their approach addresses service security, service auditing, service level compliance (SLA Monitoring), and service lifecycle management. The policy catalog defined addresses standards compliance, SLA specification, service configuration and security-related issues. The authors state that a governance model should focus on "people, processes, and technology". However, they do not clearly specify what is meant by "people" and "processes". Summarizing, the approach by Brauer and Kline (2005) addresses SOA Governance almost exclusively on the technical level, while, however, lacking a detailed description of governance methods.

Bieberstein, Bose, Fiammante, Jones, and Shah (2006) propose a SOA Governance model consisting of six governance processes and three steps for launching the governance model, combined with a SOA roadmap. Policies defined by new organizational governance entities form the basis for any decision. Their model is completed

Figure 1. Comparison results

Legend: ● — proposed and integrated ○ — partially integrated	Impact on organization	SOA Maturity Model	New roles & accountabilities	Best Practices	Metrics model	Impact on people's behaviour	SOA lifecycle	SOA roadm	Policy catal	Service life	Governance	Policy enfo ment m chanisms
Brauer and Kline (2005)	-	○	-	-	-	-	-	●	○	●	-	-
Bieberstein et al. (2005, 2006)	●	-	●	●	-	●	-	●	○	-	●	-
WebMethods (2006)	○	-	-	-	-	-	●	-	●	●	-	●
Software AG (2005)	●	●	●	●	-	-	-	●	●	●	○	-
BEA Systems, Inc. (2006)	-	-	-	-	-	-	○	-	●	●	-	-
SAP AG (2004, 2006, 2007)	●	-	○	-	○	-	-	-	●	●	○	●
Afshar (2007), Oracle	●	●	●	●	-	●	-	-	●	●	-	-
IBM (2006)	●	-	○	●	○	○	●	-	○	●	-	-
Marks and Bell (2006)	●	-	●	●	●	●	●	○	●	○	●	●
Schelp and Stutz (2007)	●	-	●	-	-	-	-	-	-	-	○	-
Allen (2008)	●	●	●	○	-	○	-	-	●	●	●	-

by a set of best practices. In a previous publication, Bieberstein, Bose, Walker, and Lynch (2005) describe an approach to guide a SOA successfully, emphasizing transformation of organizational structures and behavioral practices. They propose the Human Services Bus (HSB) as a new organizational institution, streamlining cross-department processes, thus optimally exploiting the SOA approach. Compared to others, their approach lacks a maturity model, metrics, a SOA lifecycle, a service lifecycle, and policy enforcement techniques.

The approach by WebMethods (2006) focuses on two parts: *Architecture Governance* and *Service Lifecycle Governance*. Architecture Governance comprises corporate technology standards, the definition of a SOA topology, and determination of a SOA platform strategy. Service Lifecycle Governance is divided into design-time, run-time and change-time governance, and focuses on the regulation of service design through according policies, and three different types of enforcement mechanisms. Additionally, they mention organizational changes and define a SOA lifecycle. Further techniques such as maturity models, metrics, or governance processes are not part of the approach.

The approach by Software AG (2005) includes a maturity model, a service lifecycle model, a SOA roadmap, and governance processes. However, the latter are not explicitly defined. A policy framework, based on best practices, is used in order to ensure the successful long-time operation of a SOA. They consider new roles as well as a new governance team necessary. A SOA lifecycle, a metrics model, impact on employees' behavior, and policy enforcement techniques are not explicitly included.

At BEA Systems, Inc. (2006) service lifecycles are considered the most critical requirement for a successful SOA Governance approach. They define a service lifecycle with six phases. It is the task of a central policy definition and enforcement authority to regulate the design, building, provisioning, and operation of services. Main goals are quality insurance, monitoring, and SLA management inside the SOA system. A generic SOA lifecycle concludes the approach.

The SOA Governance approach by SAP AG comprises a guidelines framework and an organizational institution, the Process Integration Content (PIC) Council. The framework consists of three parts: *modeling and implementation guidelines*, a special *review process* performed by the PIC council (guidelines enforcement), and the *continuous execution of manual and automated service tests* (SAP AG, 2007). The PIC Council guarantees quality of process integration content by reviewing interfaces for semantic correctness, ensuring standard conformity, encouraging reuse, establishing enterprise-wide consolidation and improving the integration guidelines (Wagner & Krebs, 2004). This includes the usage of assessment metrics – however, these are not explicitly specified. The design of individual services is governed by an enterprise services design guide that promotes a business-driven view based on processes and scenarios. Thereby services are not to be designed isolated from each other and are generally meant to be reused (SAP AG, 2005). The guidelines include concepts of service design for SAP internal development, business analysts, and system integrators. The approach defines basic governance processes as well as the introduction of new roles. A maturity model, best practices integration, incentives, as well as a SOA lifecycle or roadmap are not considered.

The approach at Oracle is characterized by a policy framework. Afshar (2007) considers governance policies to be the central tool of every governance approach. Eight policy domains define the decision fields and topics that have to be managed and controlled by policy enactment. These cover architecture, technology, information, financial, portfolios, people, project execution, and operational, each of them complemented by a concrete list of best practices. In particular, they define new roles and responsibilities in the domain people and demand a new organizational entity as well as the concrete definition of incentives in order to have impact on employees' behavior. As one policy category, under project execution, they define service lifecycle governance formulating the main stages of such a cycle. Additionally, Afshar (2007) describes a SOA maturity model consisting of six steps and supporting continuous improvement of the SOA. Concluding, the author presents a comprehensive governance policy framework covering a large number of aspects or problem fields of a SOA system. However, Oracle's approach lacks a SOA roadmap, SOA lifecycle, explicit governance processes, metrics, and methods for policy enforcement or compliance monitoring.

At IBM, Holley et al. (2006), Brown, Moore, and Tegan (2006), and Woolf (2006) define SOA Governance as extension of IT Governance, focusing on service lifecycle management, decision rights, policies and measures. The IBM SOA Governance model consists of a service lifecycle and a SOA governance lifecycle, both consisting of four phases: plan, define, enable, measure, and model, assemble, deploy, manage, respectively. These mutually congruent lifecycles form the core of IBM's approach and are based on best practices. Among others, organizational changes, employee training, and, implicitly, new SOA roles are included. A maturity model, a SOA roadmap, governance processes, and policy enforcement techniques are not included.

Marks and Bell (2006) define a SOA Governance framework identifying organization, SOA processes, policies, metrics, and behavior as crucial to success. They propose three steps for the actual governance process: (a) the definition of the overall SOA Governance model, organization and process, (b) the definition of SOA policies, and (c) the implementation of SOA Governance policies and their enforce-

ment. Marks and Bell (2006) define policies in six different domains, based on best practices: enterprise, business, process, compliance, technology standards, and security policies. They propose policy enforcement models and define new roles for several new tasks being introduced along with the SOA. The proposed SOA lifecycle consists of design-time, publishing and discovery, and run-time governance aspects. A SOA roadmap and a service lifecycle are mentioned, but not specified in detail. The only technique not considered by Marks and Bell (2006) is a SOA maturity model. This concept is the most comprehensive approach in our comparison.

Schelp and Stutz (2007) define a SOA Governance model consisting of a set of *management activities* combined with *organizational structures* based on governance principles. The activities, related to governance processes, comprise three groups: implementation, management, and control. The components of organizational structure are SOA strategy, organizational and operational structure, including new SOA roles. Referring to our comparison, no further techniques are mentioned.

A recent proposal for a SOA Governance Framework was made by Allen (2008). He defines a SOA Governance Framework that consists of five views: an *organizational view*, a *process view*, a *policy view*, an *infrastructure view*, and a *maturity view*. The first view defines organizational structures, roles and responsibilities that are needed by SOA Governance. The process view describes management processes at the one hand and operational processes at the other. In the policy view, several types of governance policies are described. The infrastructure view provides the technical means to support governance, e.g. by policy enforcement, or change management. In the maturity view, maturity assessment for the first four views is provided. Best practices are only partially considered, as part of the service lifecycle. Allen defines a task "communication" as part of the infrastructure view that covers impact on behavior. A SOA roadmap, a metrics model, and a SOA lifecycle are not included.

All things considered, the presented approaches introduce a number of different concepts to address governance challenges. Some are similar or related to each other. Most of them can be combined. We describe each of these concepts resulting from the analysis in detail – against the background of new model incorporating them.

2.2 Common Concepts in SOA Governance

During the analysis we performed, we identified a number of concepts that are integrated in the different approaches (cf. Figure 1). We summarize and examine them regarding their applicability in the following. In particular, we consider common best practice techniques, but do not refer to concrete implementations or instances of these.

According to the majority of the authors, a central leading *organizational entity* for the operation of a SOA-System is required, often called "SOA Governance Board". It coordinates, controls, and improves the SOA Governance processes. Along with these, special *roles and accountabilities* are defined. These new structures help fulfilling new tasks, such as regulating the impact of the introduction of a SOA on the behavior of employees (Fabini, 2007; Tilkov, 2007; Bieberstein et al., 2005).

Best practices represent the fundament for more than half of the discussed governance approaches. The best practices catalog collects experience and provides support for the creation of new policies. In general, *policies* are based on best practices. As soon as policies are enacted, they apply to the SOA System. The enactment, changing, and abolishment of governance policies is the central duty of the SOA Governance Board. Governance Policies are usually defined in a set of domains and linked with the according metrics for assessment. Policy domains cover all concerns of SOA Governance, and thus give a structure to key application domains of SOA Governance reflecting the overall governance application area. Depending on the perspective, a number of different domains were proposed

(Afshar, 2007; Allen, 2008; Marks & Bell, 2006; WebMethods, 2006). As an example, Figure 2 illustrates consolidated common policy domains.

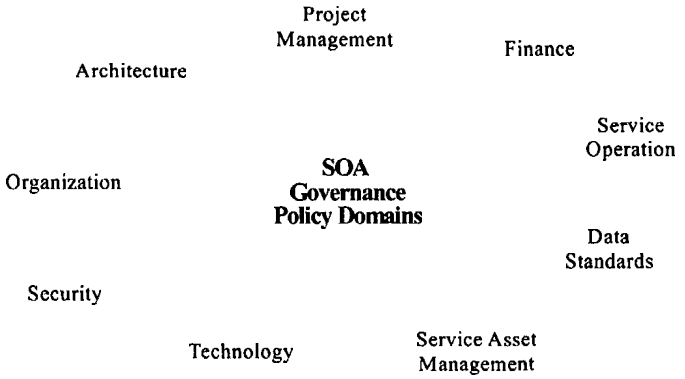
Many approaches define either a *SOA lifecycle* or *SOA roadmap*. This construct acts as a global guideline for the overall future development of the SOA system. While lifecycles assume a development cycle, roadmaps define milestones in the overall development. In some cases, the latter also consider maturity development of SOA systems, which makes it similar to SOA Maturity Models (Afshar, 2007; Bieberstein et al., 2006). The service lifecycle as part of a SOA is also discussed (BEA Systems, Inc., 2006; Software AG, 2005; Brauer & Kline, 2005; Holley et al., 2006). The common software lifecycle, defined in Software Engineering, is quite similar (Ludewig & Lichter, 2007). The challenge regarding SOA is the multitude of software artifacts that are to be treated simultaneously (cf. section 1.2).

The design of a *metrics system* is a central issue. It is common to align metrics with specific goals in order to assess the achievement of these goals, e.g. the fulfillment of a policy. Goals are usually arranged in nesting levels. Low level goals are defined by the governance policies, such as the implementation of interfaces or the adherence to a standard. These are part of higher level goals, e.g., "conformity of service design" or "general standards conformance" respectively. Marks and Bell (2006) distinguish business, process, performance, service level agreement (SLA), and SOA conformance metrics. Each of these corresponds to a specific type of policies. Metrics are dynamic, i.e. subject to change when policies change. The measured result provides feedback regarding the degree of adherence to a given policy.

The *governance processes* are the actual implementation of governance. Marks and Bell (2006) and authors at WebMethods (2006) distinguish *design-time* and *run-time* governance processes. The latter define as third group the *change-time* governance processes, while Marks and Bell (2006) add *publishing and discovery* governance processes. All of these, similar to IT Governance processes, are used to *instantiate* control of the system, i.e. regulate, in contrast to the first, the operational processes (Allen, 2008).

Policy enforcement mechanisms target the operational enactment and monitoring of adherence to policies. So far, little technical support has been proposed in this area. Authors at WebMethods (2006) propose policy enforcement points for automation: compliance checks are performed at the service registry or a proxy when invoking services. Marks and Bell (2006)

Figure 2. SOA governance policy domains



additionally propose the enterprise service bus (ESB) for this task. Authors at SAP AG (2005) define policy enforcement as manual task of the organizational entity. We discuss this topic under "Compliance Observation" in section 3.2.

SOA maturity models perform maturity checks of current IT environments of an enterprise, giving a profound indication whether an organization is ready to introduce SOA, whether its SOA implementation needs an improvement to meet minimum criteria, and what gaps it might have to bridge before it is able to do so. A SOA System can be continuously assessed and checked on a high level by a SOA maturity model. In a SOA Governance model, this assessment delivers feedback to the governance board, where decision on abolishment and enactment of policies, i.e., the next steps in control of the SOA Systems, are made. Most maturity models for SOA are based on the Capability Maturity Model Integration (CMMI) by SEI (Software Engineering Institute, 2007). Johannsen and Goeken (2007) adopt the basic structure of a CMMI maturity model with its characteristic five levels of maturity and expand the model by analyzing the maturity along three characteristics: *technology, processes*, and *organization*. For each maturity level and each characteristic a profile is available with criteria to be fulfilled on that specific level. The underlying assumption—that technology has to reach a higher level of maturity earlier than processes and organization—reflects the fact that a given technology requires certain management processes, roles and responsibilities and often causes organizational change.

The performed analysis discussed the weaknesses and strengths of several approaches to SOA Governance and identified a set of techniques that are used by these approaches. These techniques will be discussed further and checked concerning applicability in the next section.

3. STRUCTURING SOA GOVERNANCE

Despite the fact that SOA Governance is considered a crucial and mandatory element of every SOA project, so far there is no consensus about its actual structure, definitions, or comprised elements. We investigated a number of approaches to SOA Governance so far, and identified several commonly used concepts.

Basically, the term governance is borrowed from politics. Several empowered institutions interact with the purpose of regulating a large complex heterogeneous system, the state. The main target is to keep it controllable by introducing and enforcing policies or laws (structures, rights, behavioral guidelines, standards, etc.)—the system, here a state, is *governed*. In the control of IT systems, parallels emerge with this concept. The object to be regulated or governed is the enterprise architecture (the SOA). The actual laws ensuring the conformance or compliance of the system map to policies. The third element, the observation and control of adherence to laws, i.e., the "police" in SOA Governance, are *compliance observation mechanisms*.

In a governance approach, generally, two process levels are distinguished – *governance* processes and *operational* processes. The first ones regulate operational processes by control and monitoring mechanisms, while governance processes itself are mostly defined *a priori*. Hence, when building or introducing a governance approach, as first step, governance processes and structures are to be established (Marks & Bell, 2006; Afshar, 2007; WebMethods, 2006; Allen, 2008). Governance processes form the foundation for control structures and provide techniques, methods, and decision rights in order to design, form, monitor, and control operational process structures. As initial task of the actual governance, processes on the operational level are set up, changed, and

adjusted in order to be effectively monitored. The continuous consistent regulation of the SOA system is guaranteed by well-formed and reliable control structures.

Concerning concepts in SOA Governance, the *service lifecycle* is a central issue and one big requirement regarding a SOA system. Lifecycles and their management are crucial for success of a SOA Governance approach. Service lifecycles are often adjusted to the specific needs, perspectives, and notions of a company's IT. There are, for example, almost as many different lifecycles as proposals (BEA Systems, Inc., 2006; Brauer & Kline, 2005; Woolf, 2006; Afshar, 2007; Software AG, 2005). As these various different definitions of "the service lifecycle" show, obviously there is no standard or generic lifecycle that fits all perspectives. In our perspective, *SOA roadmaps* or *SOA lifecycles* hence constitute guidelines that depend on the single system and its particularities.

We designed a generic model for SOA Governance which is based on these principles allowing for the characteristics and peculiarities of all described approaches. For the development, we proceeded as follows. Based on the findings of the literature review, we identified common concepts for SOA Governance approaches in general. In the next step, we composed and consolidated these common concepts according to their actual tasks and arranged them to best fulfill their primary purposes. In the following section, it is detailed.

The Model

Main elements of our model are *SOA goals*, the *SOA as enterprise architecture*, and the *control cycle* (cf. Figure 3). Below, we discuss the main elements and characteristics that have not been outlined in section 2.2.

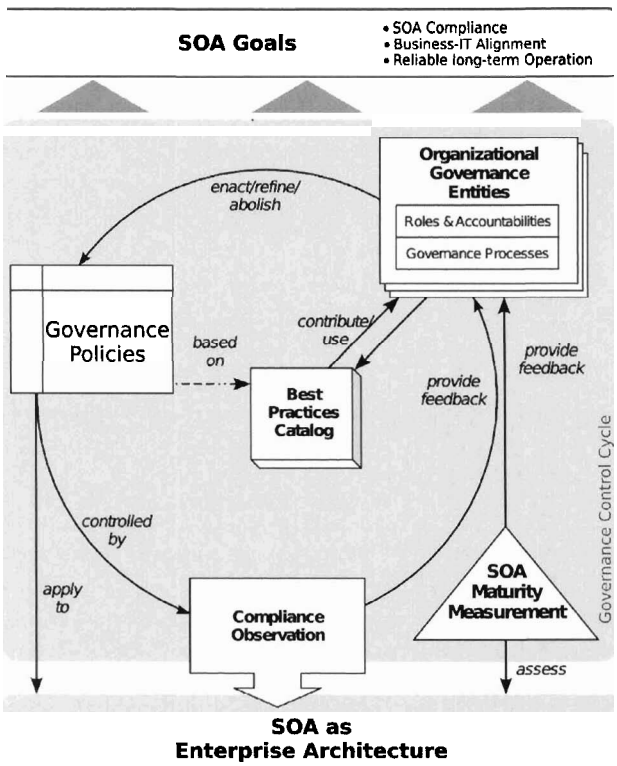
The overall purpose of the approach is to assure the achievement of goals for the SOA system. In general, these goals are derived from the overall IT goals (in IT Governance) which are specialized business goals (in Corporate Governance). Overall goals are SOA Compliance, Business-IT Alignment and reliable long term operation. They determine the necessary actions of the underlying control structures. SOA Compliance refers to the adherence of the system to legal, normative (technical) and internal regulations. Compliance with legal specifications is mandatory (e.g., Sarbanes Oxley Act), compliance with ISO norms or standard frameworks often is a benefit for a company. Internal regulations, e.g., the enforcement of company security directives are also tasks for governance. Business-IT Alignment and its improvement is part of the SOA challenges (cf. Section 1.2). The best possible integration and adaption of IT processes into the business environment is crucial to the success of a SOA (Allen, 2008). Reliable long term operation is a goal that results from due diligence management of a SOA. The overall goal of a governance approach is to provide the achievements of these goals in the long term.

The *SOA System as enterprise architecture* represents the IT system to be controlled. It consists of SOA processes, such as service production, operation, maintenance, etc., including the according business processes. The technical backbone is a central part, representing the actual architecture including registries, repositories, and the enterprise service bus (ESB). All these technical and non-technical elements are subject to the governance control.

The central part is the *SOA Governance Control Cycle* (cf. Figure 3). It represents one crucial governance process, including and involving organizational entities, governance policies, a best practices catalog, compliance observation and enforcement techniques, and a SOA maturity measurement component. Its task is the implementation and operation of an effective governance to control the SOA system, and the achievement of the SOA goals.

In the following, the organizational impact and the compliance observation component is outlined.

Figure 3. Generic model for SOA governance



Organizational Governance Entities

The Organizational Governance Entities (e.g., "SOA Board") bear responsibility for the reliable operation, regulation, and control of the SOA system. They consist of representatives from every organizational layer—optimally, they are manned with members from the upper management as well as IT architects and developers. The single organizational entities can be structured in a hierarchical or in a coordinating manner, e.g., in the case of territorially structured company branches (Fabini, 2007). Together, these entities form a new organizational structure.

The SOA Boards define and abolish governance policies. In this respect, they act like a republic parliament. During this procedure, the best practices catalog is continuously maintained, i.e., enhanced or adjusted, respectively. The enacted policies apply to the SOA, and the components *compliance observation* and *maturity measurement* give feedback to the SOA Boards. While the latter assesses the system from a general point of view, e.g., regarding its load capacity, the first component performs detailed compliance checks, e.g., concerning security issues or process compliance. Based on their feedback, the SOA Boards decide on further steps (cf. Figure 3).

Along with the introduction of new organizational governance entities come new roles and accountabilities that are implemented or realized by those entities. Bieberstein et al. (2006) define a number of new SOA-related roles for a company, introducing new accountabilities and decision rights. More than 70% of the examined approaches in section 2.1 also address

these issues (cf. Figure 1). The new SOA Boards are the actors in the overall SOA Governance processes, forming the superior level compared with operational processes. The above outlined control cycle is one of them. Further processes and tasks are policy management, enforcement mechanisms, communications, change management, architecture review processes, design-time and run-time governance, SOA maintenance and many more (Allen, 2008; Marks & Bell, 2006; WebMethods, 2006; SAP AG, 2005; Bieberstein et al., 2006; Schelp & Stutz, 2007).

Compliance Observation

Regulating a system without effective enforcement and control mechanisms is not possible. One central element of governance is the *compliance observation* component that enforces and monitors system compliance. The deployed mechanisms and techniques check adherence to the given regulations specified by policies in the different domains and enable the continuous compliance observation of the SOA system.

Regarding the processing, most techniques are of manual nature. According to WebLayers (2007), 83% of interviewed companies perform manual design reviews, 54% rely on manual and only 13% perform automated pre-registration checks. Concerning the techniques, mostly automated business activity monitoring and data consolidation techniques or check lists are implemented (WebMethods, 2006; Software AG, 2005), although the technical opportunities in this area are manifold. The following list itemizes general technical methods usable for the implementation of compliance checks:

- Automated check list processing as proposed by WebMethods (2006) and Software AG (2005), combined with checks at proxies or intermediaries, realized by, e.g., an ESB (Marks & Bell, 2006; WebMethods, 2006)
- Business activity monitoring (WebMethods, 2006) linked with automated data consolidation in order to compute warning levels
- Formal verification methods using, e.g., process algebras ("hard" verification)
- Matching by semantic description using ontologies ("soft" verification)
- Rule-based decision-support systems providing automated analysis, independently taking or proposing (re)actions

A further aspect of compliance observation is the time component. Depending on the system and its peculiarities, compliance checks can be scheduled in different ways.

- **Ex-post vs. ex-ante analysis:** Is a check to be performed upon demand, ex-post or is the system required to identify potential future violations, i.e. to work proactively?
- **Frequency:** When are compliance checks to be performed—upon request or frequently? This mainly targets SLA, business process, security-related, and similar regulations.

Compliance observation is yet an open research field. Many challenges exist, mostly neglected by industry white papers and commercial products. Existing approaches often provide basic, yet insufficient support for compliance observation.

Summary

The model at hand has been developed based on the results of the above presented literature review. It aims at integrating all aspects of the compared approaches in order to allow for their perspectives and capabilities.

We defined a *control cycle*, being a crucial part of the SOA Governance processes. It covers the basic process of policy design, enactment, enforcement, and abolishment, supported by best practices, enforcement methods, and maturity assessment. It is based on the PDCA-Cycle by Deming (1986).

The model is the first to explicitly address and discuss the issue *compliance observation*

covering *policy enforcement mechanisms*. This issue is addressed by vendors in case their products support a type of enforcement, e.g. check at service registry. Additionally, questions concerning the frequency, analysis type, and level of detail of service checks arise and are to be answered. In literature as well as in industry white papers these issues are often omitted, thus attesting the "narrow view" of the software vendors identified by Allen (2008).

Concluding, the model consolidates perspectives and techniques of existing approaches to SOA Governance, emphasizes the control cycle and enhances it by the concept of compliance observation. So far, all concepts of the compared approaches map to this model.

4. CONCLUSION

So far, *Governance for Service-oriented Architectures* is an open, unstructured field. Though there are a number of approaches, mostly frameworks and best practices, a consensus on what a SOA Governance approach is required to be capable of and on what are the main concepts has not been achieved by now. By providing a literature review on related proposals and deriving a model on top of the results, this article aims at providing a first step concerning the active structuring of this domain.

Summarizing, we conclude the following statements from our examinations:

- Based on common characteristics of SOAs and the emerging challenges, the installation and operation of governance approaches for SOAs is essential, especially regarding challenges such as managing and unifying the SOA-inherent heterogeneity and complexity, as well as the regulation of cross-organizational service deployment and further new capabilities of a SOA as EA.
- As a result of the literature review, we stated that few proposals (e.g., Marks & Bell, 2006) represent *holistic* approaches. Most approaches are characterized by a "tunnel perspective", focusing on selected issues and motivated by a specific software product. In contrast, most authors agree that a holistic governance approach is crucial for SOA Governance. This proofs a big gap between understanding and action in this domain and discloses room for improvement.
- According to the literature review, *policy frameworks* combined with *best practices* are an accepted element of SOA Governance. There seems to be no adequate alternative mechanism with comparable abilities.
- *Organizational changes* and the introduction of new SOA-specific *roles and accountabilities* are considered crucial by the majority of the examined approaches. Together with policy frameworks and best practices, they are regarded as the *four common concepts* for SOA Governance.
  While *policy enforcement mechanisms* as extension to policy frameworks are integrated by only a few approaches (e.g., WebMethods, 2006), no approach considers the partial automation of compliance checks. An according functionality here could strongly support software vendor products.
  Based on the contributions compared in the literature review, we consolidated *common SOA Governance policy domains* that structure a common policy framework and cover all concerns of SOA Governance.
- According to the literature review results, *service lifecycle* and its management is a very important concept in the context of SOA Governance. It is explicitly considered by almost all approaches and represents the most prominent concern of SOA Governance.
- A significant deficit has been identified concerning *compliance enforcement*. The least approaches make proposals concerning any technical support for governance activities. Of those proposing according techniques, almost none exceed the processing of a checklist.

- The developed model comprises all main concepts and components of the examined approaches, and extends them by a control cycle. With this model, we provide a first version of a generic reference model for SOA Governance.

Overall, this contribution aims at supporting and driving structuring intensions in the area of SOA Governance and at identifying concepts that are common to all approaches in this domain.

Future Trends

The approaches' diversity in understanding, perspective, scope, and techniques shows that SOA Governance is an upcoming research topic that has not been intensively examined yet. Hence, we see a variety of future trends.

Developments might include a formalization of approaches to SOA Governance, similar to frameworks in IT Governance. Discussions to be held will include how to cover SOA peculiarities and at the same time avoid reinventing IT Governance frameworks. Most probably, part of these trends will be a well-founded enhancement and a combination of existing IT Governance frameworks in a way that allows for the special regulation and control requirements of service-oriented enterprise architectures.

The largest area of potential scientific achievements is located in the field of automated compliance checks. As discussed above, there is a multitude of potential mechanisms and techniques to address the problem of automating the check of a system's adherence to regulations or policies. Though a small number of SOA Governance models propose ideas (for techniques to be deployed), so far they did not establish as technologies in operation.

Concluding, we see a huge research potential in the area of SOA Governance in particular.

GLOSSARY

Service-Oriented Architecture (SOA)

A SOA is an application architecture within which all functions are defined as independent services with well-defined callable interfaces which can be called in defined sequences to form business processes. (Channabasavaiah, 2003)

IT Governance

IT governance is the organizational capacity exercised by the Board, executive management and IT management to control the formulation and implementation of IT strategy and in this way ensure the fusion of business and IT. (De Haes & van Grembergen, 2004)

SOA Governance

SOA Governance is a holistic long-term management model. It guarantees sufficient adaptability and integrity of an SOA system as well as the ability to check services concerning capability, reusability, security, and strategic business alignment. Overall goals are SOA Compliance, and the guarantee of reusability and standardization throughout the system.

SOA Maturity Model

A SOA capability maturity model (CMM) is a structured collection of elements that describe certain aspects of maturity in an IT environment (technology, processes, and organization) where a SOA is implemented or is considered to be implemented. Described aspects may comprise, e.g., basic requirements, responsibilities, documentation and continuous improvement. A maturity model can be used as a benchmark for comparison and as an aid in steering and planning capabilities needed for running a SOA.

Generic Model for SOA Governance

The generic model for SOA Governance consolidates, structures, and enhances common techniques deployed in SOA Governance. It represents a general approach to combining the capabilities of existing SOA Governance approaches.

Compliance Observation

Compliance Observation in the context of SOA Governance comprises policy enforcement and monitoring of the adherence to policies. Examples for deployable techniques are automated check list processing, business activity monitoring, formal verification methods, rule-based decision-support systems, and similar techniques.

Policy Framework

In the context of SOA Governance, a policy framework is the instantiation and collection of explicit, formalized regulations. It is often structured into policy domains, such as, e.g., architecture or security-related regulations and represents a central component in the operation of SOA Governance.

REFERENCES

Afshar, M. (2007). *SOA Governance: Framework and Best Practices.* Retrieved July 2008, from http://www.oracle.com/technologies/soa/docs/oracle/soa-governance-best-practices.pdf

Allen, P. (2008). SOA Governance: Challenge or Opportunity? *CBDI Journal,* 20-31. Retrieved June 2008, from http://www.cbdiforum.com/secure/interact/2008-04/challenge_opportunity_br.php

Bieberstein, N., Bose, S., Fiammante, M., Jones, K., & Shah, R. (2006). *Service-Oriented Architecture (SOA) Compass – Business Value, Planning, and Enterprise Roadmap.* IBM Press.

Bieberstein, N., Bose, S., Walker, L., & Lynch, A. (2005). Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals. *IBM Systems Journal, 44*(4), 691–708.

Brauer, B., & Kline, S. (2005). *SOA Governance: A Key Ingredient of the Adaptive Enterprise.* Retrieved June 2008, from http://www.managementsoftware.hp.com/ products/soa/swp/soa_swp_governance.pdf

Brown, W., Moore, G., & Tegan, W. (2006). *SOA governance – IBM's approach.* Retrieved July 2008, from ftp://ftp.software.ibm.com/software/soa/ pdf/SOA_Gov_Process_Overview.pdf

Channabasavaiah, K., Holley, K., & Tuggle, E., Jr. (2003). *Migrating to a service-oriented Architecture, Part 1.* Retrieved August 2008, from http://www-128.ibm.com/developerworks/library/ws-migratesoa/

De Haes, S., & van Grembergen, W. (2004). IT Governance and its Mechanisms. *Information Systems Control Journal, 1.* Retrieved August 2008, from http://www.uams.be/uploads/ documentenbank/92e0ee8f586d709d438799d377656b1b.pdf

Deming, W. E. (1986) *Out of the crisis.* Cambridge, MA: MIT Center for Advanced Engineering Study. ISBN 0-911379-01-0.

Fabini, M. (2007). Governance für komplexe SOA-Unternehmungen: Eine Vision für das Schweizer Gesundheitswesen. In G. Starke & S. Tilkov (Eds.), *SOA-Expertenwissen* (pp. 309-323). Heidelberg, Germany: Dpunkt Verlag.

Holley, K., Palistrant, J., & Graham, S. (2006). *Effective SOA Governance.* Retrieved July 2008, from ftp://ftp.software.ibm.com/software/uk/flexible/wp/effective_soa_governance.pdf

Johannsen, W., & Goeken, M. (2007). *Referenzmodelle für IT-Governance – Strategische Effektivität und Effizienz mit COBIT, ITIL & Co.* (1st ed.). Heidelberg, Germany: Dpunkt Verlag.

Josuttis, N. M. (2007). *SOA in Practice: The Art of Distributed System Design (Theory in Practice).* Sebastopol, CA: O'Reilly.

Kalex, U. (2007). Von der Geschäftsarchitektur zur SOA-Governance. In G. Starke & S. Tilkov (Eds.), *SOA-Expertenwissen* (pp. 325-340). Heidelberg, Germany: Dpunkt Verlag.

Keller, W. (2007). SOA-Governance: SOA langfristig durchsetzen und managen. In G. Starke & S. Tilkov (Eds.), *SOA-Expertenwissen* (pp. 289-307). Heidelberg, Germany: Dpunkt Verlag.

Kobielus, J. (2006). *SOA governance: Preventing rogue services*. Retrieved July 2008, from http://www.networkworld.com/supp/2006/ndc3/062606-ndc-soa-governance.html

Krafzig, D., Banke, K., & Slama, D. (2004). *Enterprise SOA: Service-Oriented Architecture Best Practices (The Coad Series)*. Upper Saddle River, NJ: Prentice Hall.

Ludewig, J., & Lichter, H. (2007). *Software Engineering* (1st ed.). Heidelberg, Germany: Dpunkt Verlag.

Manes, A. T. (2005). *The elephant has left the building*. Retrieved July 2008 from http://www.intelligententerprise.com/showArticle.jhtml?articleID=164301126 &pgno=1

Marks, E., & Bell, M. (2006). *SOA: A Planning and Implementation Guide for Business and Technology*. New York: John Wiley & Sons.

Melzer, I. (2007). *Service-orientierte Architekturen mit Web Services. Konzepte–Standards–Praxis* (2nd ed.). Spektrum Akademischer Verlag, München.

Newcomer, E., & Lomow, G. (2004). *Understanding SOA with Web Services (Independent Technology Guides)*. Reading, MA: Addison-Wesley Professional.

Niemann, M., Eckert, J., Repp, N., & Steinmetz, R. (2008). *Towards a Generic Governance Model for Service-oriented Architectures*. Paper presented at the Fourteenth AMCIS Conference, Toronto, Ontario, Canada.

Papazoglou, M. P. (2003). Service-Oriented Computing: Concepts, Characteristics and Directions. In *Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE 2003)*, Rome, Italy (pp. 3-12).

SAP AG. (2005). *Enterprise Services Design Guide*. Retrieved July 2007, from https://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uuid/943e83e5-0601-0010-acb5-b16258f5f20a

SAP AG. (2007). *Governance for Modeling and Implementing Enterprise Services at SAP Enterprise SOA Solution Management*. Retrieved July 2007, from https://www.sdn.sap.com/irj/ sdn/go/portal/prtroot/docs/library/uuid/f0763dbc-abd3-2910-4686-ab7adfc8ed92

Schelp, J., & Stutz, M. (2007). SOA-Governance. In H.-P. Fröschle & S. Reinheimer (Eds.), *Serviceorientierte Architekturen (SOA)* (pp. 1-10). Heidelberg, Germany: Dpunkt Verlag.

Software, A. G. (2005). *SOA Governance – Rule your SOA*. Retrieved August 2007, from http://www.softwareag.com/de/Images/WP_SOA_Governance_D_tcm17-22130.pdf

Software Engineering Institute (SEI). (2007). *The Capability Maturity Model Integration (CMMI)*. Retrieved September 2007, from http://www.sei.cmu.edu/publications/documents/ 07.reports/07tr017.html

Systems, B. E. A. Inc. (2006). *Service lifecycle governance*. Retrieved April 2007, from http://www.itworldcanada.com/Admin/Pages/Assets/DisplayAsset.aspx?id=e0a24263-a10a-4887-8d45-582261587176

Tilkov, S. (2007). *Roles in SOA Governance*. Retrieved June 2007, from http://www.infoq.com/articles/tilkov-soa-roles

Wagner, J., & Krebs, F. (2004). *PIC Council Charter*. Retrieved July 2007, from https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/d987b590-0201-0010-1684-c4754d40ddb9

WebLayers. S. F. (2007). *2007 Survey on SOA Governance*. Retrieved July 2008, from http://www.bitpipe.com/detail/RES/1209665671_580.html

WebMethods. (2006). *SOA Governance – Enabling Sustainable Success with SOA*. Retrieved May 2007, from http://www1.webmethods.com/PDF/whitepapers/SOA_Governance .pdf

Windley, P. J. (2006). SOA Governance: Rules of the game. *INFOWORLD.COM*, 29-35. Retrieved October 2007, from http://akamai.infoworld.com/pdf/special_report/ 2006/04SRsoagov.pdf

Woolf, B. (2006). *Introduction to SOA Governance*. Retrieved July 2007 from http://www.ibm.com/developerworks/library/ar-servgov/

Michael Niemann is a researcher within the multimedia communications lab at the Technische Universitaet Darmstadt, Germany. He is a work package lead for "Service Governance" in the research project THESEUS/TEXO initiated by the Federal Ministry of Economy and Technology. His research focuses on the management of distributed workflows in service-oriented architectures, IT governance for large heterogeneous systems, and usage of semantic web technologies in operational IT governance.

André Miede is a researcher with the E-Finance lab and multimedia communications lab partnership at Technische Universitaet Darmstadt. His research focuses on security for service-oriented architectures and the Internet of Services, especially on attack and countermeasure aspects. In addition to his research activities, he is a consultant in the financial services sector for BearingPoint in Germany.

Wolfgang Johannsen is managing partner of its-okay Ltd. & Co. KG, a Germany-based company offering educational services and consultancy in the broader context of IT Governance and IT Compliance. He also is a scientist at Technische Universitaet Darmstadt, Germany, and lecturer at Frankfurt School of Finance & Management. He co-authored a well-recognized book on IT-Governance (in German). Before running his own enterprise, Wolfgang Johannsen was associate partner at Accenture GmbH, Germany. In his work as a consultant he ran strategic IT projects for financial services clients. Johannsen started his career with IBM as research staff member for computer science. He then joined Deutsche Bank where his last management position was chief of staff (IT/operations).

Nicolas Repp has been a researcher at multimedia communications lab at the Technische Universitaet Darmstadt. His research focuses on service management, the monitoring of service quality in cross-organizational business processes, as well as software architecture management and IT Governance.

Ralf Steinmetz worked for over nine years in industrial research. Since 1996, he is head of the Multimedia Communications lab at the Technische Universitaet Darmstadt, Germany. From 1997-2001 he directed a Fraunhofer Institute. In 1999, he founded the Hessian Telemedia Technology Competence Center. His thematic focus is multimedia communications. With over 400 refereed publications he became ICCC Governor (1999); was awarded as the ranking of Fellow of both, the IEEE (1999) and the ACM (2002). In 2005, as member of the Hessian technology advisory board he is appointed as the advisor for information and communications technology by the Hessian government.