André Miede, Michael Niemann, Stefan Schulte, Julian Eckert, Aneta Kabzeva, Nicolas Repp, Ralf Steinmetz:
Selected Topics in Service Engineering and Management for Enterprise Systems. In: Darshana Sedera, Mary Sumner, Bjoern Johansson: Proceedings of the Pre-ICIS 2008 Workshop on Enterprise Systems Research in MIS, December 2008.

Enterprise Systems Workshop Submission

Title:
Selected Topics in Service Engineering and Management for Enterprise Systems

Authors:
André Miede, Michael Niemann, Stefan Schulte, Julian Eckert,
Aneta Kabzeva, Nicolas Repp, Ralf Steinmetz

Contact Information:
Fachgebiet Multimedia Kommunikation (KOM)
Technische Universität Darmstadt
Merckstr. 25
D-64283 Darmstadt
Telephone: +49 (0) 6151 16 6156
Fax: +49 (0) 6151 16 6152
Corresponding Author: Andre.Miede@KOM.tu-darmstadt.de

Type:
Issues and opinion

Total Word Count:
3.923

Abstract:
The service-oriented architectures paradigm offers enterprises the chance to address their integration and agility issues both strategically and technologically. This article presents selected research issues and results in the field of distributed, service-based enterprise systems, ranging from semantic service description and resource planning to management issues such as service monitoring, governance, and security.

Key Words:
Service-oriented Architectures, Service Engineering, Service Management, Governance, Security

Information about the Authors:

André Miede is a researcher with the E-Finance Lab and Multimedia Communications Lab partnership at Technische Universität Darmstadt. His research focuses on service-oriented architectures, especially on security, audit, and compliance aspects. In addition to his research activities, he is a technology consultant in the financial services sector for BearingPoint in Germany. André received a Master's degree from the University of Magdeburg in 2006.

Michael Niemann studied computer science and business administration at the Technische Universität Kaiserslautern and at the Lomonosov Moskow State University in Moscow (Russia). He wrote his master's thesis within the scope of a cooperation ("Virtual Office of the Future") between Ricoh R&D Department in Tokyo (Japan) and the German Research Institute of Artificial Intelligence (DFKI) in Japan enhancing a semantic workflow management tool by project management functionality. Since June 2007, he is a research assistant at the Multimedia Communications Lab (KOM) at Technische Universitaet Darmstadt. His research topics are business process evaluation and verification in the scope of governance for service-oriented architectures (SOA Governance).

Stefan Schulte is a PhD Candidate at the Multimedia Communications Lab (KOM) at Technische Universität Darmstadt. He received a diploma degree in economics and a Bachelor in computer science from the University of Oldenburg and a Master of Information Technology from the University of Newcastle, New South Wales, in 2005 and 2006. His research interests include SOA in the banking industry, semantic concepts in SOA and especially semantic descriptions of Web services. Findings from his research have been published in more than twenty refereed scholarly publications.

Julian Eckert studied joint electrical engineering and business administration at Technische Universität Darmstadt and at the University of Massachusetts (USA). After receiving a scholarship from the German Academic Exchange Service (DAAD) he received his diploma degree in 2006. In 2006 he started his research at KOM in performance analysis of service-oriented workflows. He is the co-head of the SOA Competence Center of httc e.V. and participates in the research cooperation "E-Finance Lab".

Aneta Kabzeva is a PhD Candidate at the Integrated Communication Systems Group at the University of Kaiserslautern in research cooperation with the IT architectures group at the Multimedia Communications Lab at Technische Universität Darmstadt. Her research focuses on service engineering, in particular on architecture description languages, evaluation methods, and management models. She received her diploma degree in Computer Science from the University of Kaiserslautern in 2007.

Nicolas Repp is head of the IT architectures research group at the Multimedia Communications Lab at Technische Universität Darmstadt. His research focuses on the management of distributed workflows, the monitoring of service quality as well as IT governance in large heterogeneous systems. He primarily works for the E-Finance Lab Frankfurt am Main e. V. research partnership. Nicolas received a diploma in Business Informatics from Technische Universität Darmstadt.

Dr.-Ing. **Ralf Steinmetz** is professor of Multimedia Communications at Technische Universität Darmstadt in Germany. Together with more than 30 researchers, he has been working towards the vision of real "seamless multimedia communications". He has contributed to over 400 refereed publications, become an ICCC Governor; and is a Fellow of both the IEEE and ACM. Professor Dr. Ralf Steinmetz is a member of the Scientific Council and president of the Board of Trustees of the international research institute IMDEA Networks.

1 Introduction and Scenario

A global, fast-paced, and highly competitive economy confronts modern enterprises with many challenging requirements. To overcome these challenges, both the business side and the technology side have to work hand-in-hand seamlessly, while maintaining a mutual understanding of both the challenges and their possible solutions on both sides. Among various requirements which affect existing and future enterprise information technology (IT) architectures, the two following have a very strong impact on both research and applications [Jos07, KBS04, NL04]:

- The integration of heterogeneous systems.
- Achieving a high flexibility of business processes and their underlying IT.

The service-oriented architectures (SOA) paradigm can be used to facilitate an enterprise infrastructure which supports the above requirements.

Concerning the relevance and impact of SOA, the yearly survey “SOA Check” has shed some light on these issues for the German, Swiss, and Austrian market in 2008 [SC08]. Key results include:

- More than 50% of the study’s participating enterprises consider the relevance of SOA “high” or “very high”.
- About 50% of them plan to introduce an SOA.
- However, more than 50% admitted to have dealt with the issue only within the last two years.

These figures and further results of the survey indicate that enterprises now have a more realistic understanding of the benefits SOA offers than before, i.e., during the SOA hype. This is a good opportunity and foundation for both a business-driven SOA approach and research in this field.

At the heart of SOA is the concept of a “service” – still being an actively discussed concept – but which is generally to be understood as the technological representation of business functionality [Jos07]. By using services as building blocks, business processes can be composed from them, abstracting the processes from the underlying (usually monolithic) applications and allowing for compositions even across organizational boundaries. Such common and relevant scenarios remain an important focus of research in this field: cross-organizational, service-based workflows. Figure 1 gives a schematic impression of this scenario’s setup. A common variation is also to have an intermediary additional layer between service consumers and service providers, which can serve several organizational purposes such as load balancing (cf. section 2.2).

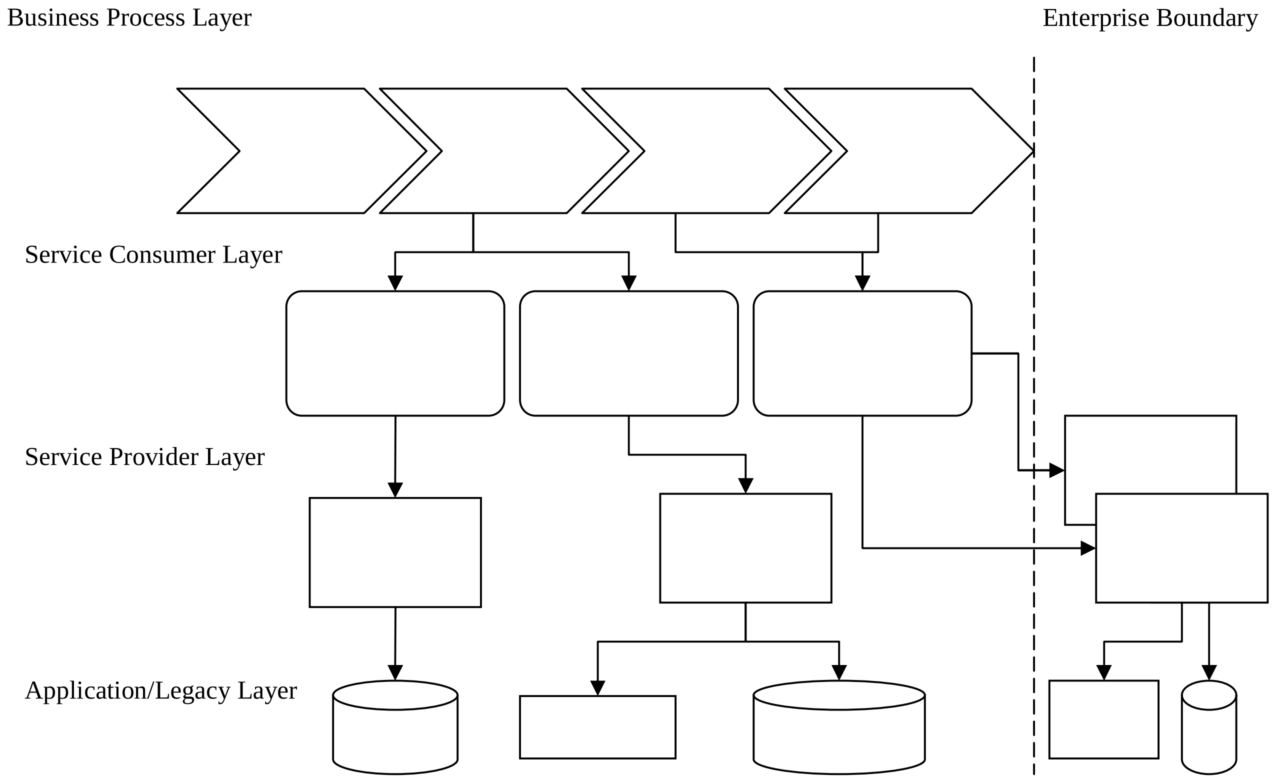

Figure 1: Cross-Organizational, Service-Based Workflows.

This paper presents several selected topics in the field of service engineering and management for enterprise systems. The focus is on challenges that have an impact on both research and its applications in industrial contexts. For each of these topics, relevant related work that addresses the challenges is presented to give the reader a brief, but thorough introduction.

The rest of the paper is structured as follows:

- Section 2 gives an overview of selected major research challenges the SOA community faces. These challenges cover both the business- and technology-relevant topics and range from semantic service descriptions over resource planning issues, service monitoring, and security to top-level issues such as architecture design and governance.

- Section 3 concludes this article, summarizing the main results and future challenges.

2 Selected Topics in Service Engineering and Management

Designing, building, and operating service-based systems is a process that has to consider many different layers of abstraction in order to support the requirements mentioned above. While the following selection of research challenges is by no means complete, it represents an important subset of key points both on different abstraction levels between business and technology and in different steps of a system's life cycle.

The research topics are structured and ordered according to Papazoglou's SOA pyramid [Pap03] as depicted in Figure 2. While each topic presented in this section is treated in a self-contained manner to make it easily accessible even apart from the paper, all topics are interlinked via the shared base scenario described in section 1 and their respective contribution to the pyramid. However, the selected topics constitute together an overview of key elements towards an enterprise service-oriented architecture.

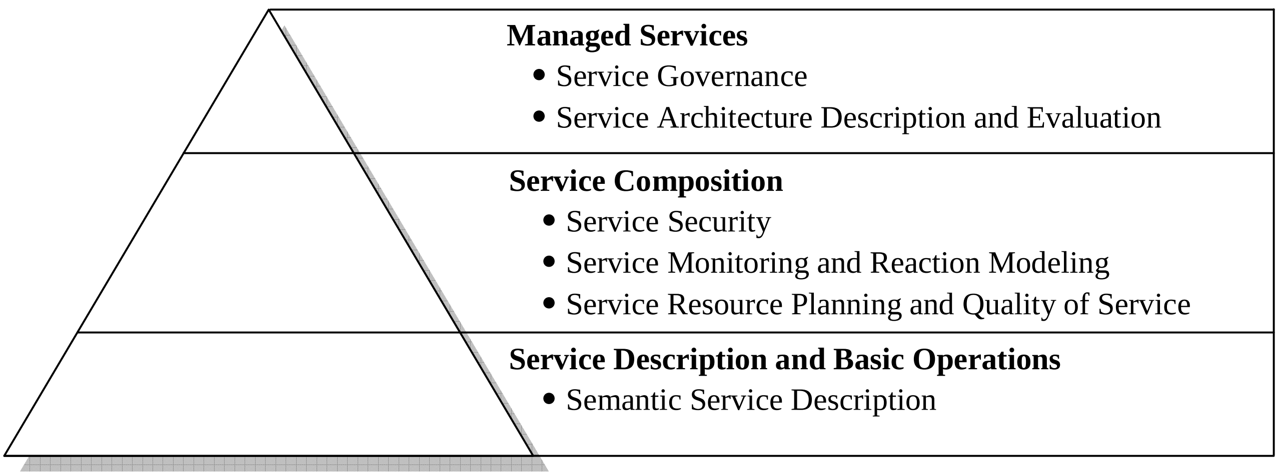

Figure 2: The Topics of this Paper Structured According to Papazoglou's SOA Pyramid.

2.1 Semantic Service Description

Service-orientation is a widespread topic which does not only affect the way enterprise and IT architectures are planned, set up, and implemented, but it will also enable the outsourcing of minor tasks instead of whole business processes and/or complete departments. As these minor tasks can be accomplished by single services, it is quite likely that service markets, i.e., marketplaces, where services are provided and requested, will be established in the near future [Pap03]. One example how to implement such marketplaces is the topic of the research project Theseus TEXO[1].

In such markets, the service providers, service requestors and service brokers involved rely on a common way to describe the functionalities and characteristics of services in order to advertise and find the demanded services. Thus, it is necessary to provide a powerful but easy way to describe and retrieve services.

Only if the demanded services can be identified, it is possible to address further problems, e.g., Quality of Service (QoS) [BGR+05], and finally invoke a service. Unfortunately, a pure syntactic description of a service's functionalities and characteristics can only be sufficient if all parties involved use exactly the same vocabulary. This is quite unlikely even within the same corporate environment. Hence, it is necessary to provide information about the meaning of, e.g., service descriptions which can be accomplished by providing semantic annotations for these descriptions.

Since the first presentation of semantic markup for Web Services in 2001 [MSZ01], the (semi-) automatic annotation, retrieval, and composition of Semantic Web Services has been a research topic of great interest and several different approaches to achieve these goals have been

[1] http://theseus-programm.de/scenarios/en/texo

proposed. However, especially the automatic annotation of services with semantic information and the discovery of services with incomplete semantic annotations still have not been answered satisfyingly.

Furthermore, approaches are often standard-specific und can only be applied to a particular Web Service standard, e.g., OWL-S [MBM+07] or SAWSDL [FL07]. Hence, it is assumed that these two major problems have to be tackled in order to achieve the automation of information use and dynamic interoperability, which are the main motivation for the usage of semantic Web Services [SER+08].

2.2 Service Resource Planning and Quality of Service

Due to increased collaboration between enterprises, cross-organizational workflows become very prominent. The SOA paradigm supports those cross-organizational workflows by enabling the on-demand invocation of several services from internal as well as from external partners and composing them to a workflow. In a scenario with a lot of workflow requestors and service providers, an intermediary has to monitor all incoming workflow execution requests and has to build up workload forecasts. Further, the workflow execution requests have to be prioritized according to their execution deadline, determined by the workflow requestors.

The challenge in this context is to ensure several QoS demands during workflow composition. Thus, performance evaluation of service-oriented workflows plays an important role in order to avoid performance degradation. As a means to ensure that the workflow execution remains feasible and that Service Level Agreement (SLA) violations due to overload are avoided, resource planning of services and workflows is necessary. Accordingly, by allowing just-in-time integration and interoperability of multiple and alternative services (with the same functional parameters), service-oriented workflows comprehend yet unsolved complexity and dynamics in resource allocation. Formerly long-term and therefore more strategic decisions of resource selection/ allocation develop to short-term decisions that have to be made instantaneously. Enterprises have to react very fast to a changing environment and changing business needs.

The addressed service composition problem is widely studied in literature. Those composition problems can be solved with the help of genetic algorithms [CPE+05]. Naseri and Towhidi analyze the QoS-aware composition of services with the help of ontologies and artificial intelligence planner [NT07]. An approach for the QoS-optimization with tasks deadlines is shown by Orleans and Furtado [OF07]. These approaches only focus on service composition problems for one workflow request and do not consider limited execution capacities of services as well as several cost models.

Due to a limited execution capacity of services within a specific time slot, resource planning addresses the problem of online resource allocation on the side of an orchestrator of service-oriented workflows under real-time conditions. Besides this, also other non-functional QoS parameters, e.g., security, availability, and delay are similarly important concerning Web Service selection scenarios. A detailed resource planning decides how many services have to be invoked (in parallel) and at which process step in order to ensure that all workflow execution requests can be served in a specific time period. In the literature, the selection of Web Services as an optimization problem – regarding user's preferences and constraints on the non-functional attributes of the resulting service – are discussed extensively [BSR+06]. This selection problem has proven to be NP-hard [LY05] and therefore difficult to solve. Due to the fact that exact solutions comprise a high computational overhead, heuristics have to be developed in order to solve this problem in an adequate computational time. For resource planning purposes, also worst-case as well as average-case considerations become important [PYY+04, ESN+08, ESR+08]. Currently, there is a lot of research effort going on in the area of resource planning and resource allocation.

2.3 Service Monitoring and Reaction Modeling

Establishing cross-organizational workflows is based on a dependable and trustworthy service exchange between enterprises. SLAs are used to define both the responsibilities and requirements of the participants. However, one cannot only rely on the SLAs themselves, the actually fulfilment or non-fulfillment of SLAs have to be monitored, ideally live at runtime. To achieve this automatically, monitoring requirements should be specified or derived from higher level business requirements. This is usually part of the general automated transformation of business processes into IT applications. Thus, important questions which arise in the context of monitoring services include the following:

- Which elements or parameters of a service have to be monitored?
- What should and can be the reaction to any detected deviation from the SLAs?
- Where should monitoring be performed, i.e., where should monitoring units be distributed in a distributed infrastructure?

Basically, the manifold approaches for the monitoring of service-based workflows can be divided into two classes: the monitoring of functional and non-functional requirements. In the following, an overview of the existing approaches is given. Both Robinson [Rob05] and Spanoudakis and Mahbub [SM06] use logical languages to describe functional monitoring requirements. The latter discuss the transformation of BPEL4WS code into a language based on event calculus and its resulting monitoring. Both approaches do not consider any deviation handling as a result of monitoring and its reporting. Using pre- and post-conditions for monitoring included as extensions to BPEL code is an idea by [BG05]. It uses a BPEL pre-processor to extract the monitoring requirements from the code. Deviation handling is also not integrated in this solution.

For the monitoring of both functional and non-functional requirements Lazovik et al. [LAP06] use a proprietary language to describe business rules, [LDK04] uses WS-Agreement to negotiate requirements specified in different languages. Repp [Rep08] presents AMAS.KOM, an integrated monitoring approach which supports both the detection of SLA violations and first steps to remedy the problem, using a requirements and reactions policy language based on WS-Policy for modelling.

Given the demands and approaches as described above, it becomes evident that monitoring has still various challenges to offer such as the distribution problem of monitoring units or how to integrate monitoring capabilities into existing architectures. Important aspects deserving more attention and work are issues such as deriving both monitoring and deviation handling requirements from SOA governance information (cf. section 2.6) or use monitoring infrastructure for security purposes, i.e., auditing of critical transactions.

2.4 SOA Security

As shown in the previous sections, an SOA imposes various and difficult challenges. This might be the reason why security aspects are often excluded from concepts and postponed to be integrated “afterwards”. This section briefly outlines what SOA-specific challenges security has to offer and why security has to be addressed from the very start of any SOA project and not just ad hoc during operation.

SOA security is different from the security of monolithic legacy systems regarding the following aspects [SSK08, KBS04, Jos07]:

- Communication and interaction occurs across enterprise boundaries. This exposes parts of an enterprise's IT to the outside world, which is not as safe and controlled as the inside. Thus, it becomes crucial for protection and, for example, for accounting, which services of an enterprises are accessed by whom and how this happens.
- Communication and interaction happens with more or less anonymous service consumers and providers based on open standards. For example, this makes it necessary to verify exchanged information concerning contents and origin.
- Various layers of abstraction are involved, ranging from the business process layer down to legacy systems behind services. This makes end-to-end-security necessary, as point-to-point-security is no longer sufficient.

These differences have led to new challenges in designing an SOA to meet the common security requirements such as authentication, authorization, confidentiality, integrity, availability, non-repudiability etc. For example, one question involves deciding where user authentication should take place, i.e., against the user's application frontend, against the enterprise's SOA platform, or against each service itself. Another question involves where to encrypt which information, i.e., on the transport-level, on the message-level, or only parts of the message. [KBS04, Jos07]

Yet another important aspect is the administration of an SOA's security mechanisms. It is mandatory to have these mechanisms as flexible as the business processes they secure. To achieve this, security must not be introduced ad hoc into an SOA but the SOA must already be designed and implemented with security requirements already in mind. This makes it often necessary to decouple security functionality from the application logic, i.e., by providing security as a service or as part of the general SOA infrastructure [SSK08]. In this context, how to model security mechanisms has to be considered from both a business and a technical point of view – a process which could benefit extensively from the use of design patterns for security aspects as building blocks or validation models [Sch03].

An approach which includes security aspects as early as possible in the development process of an SOA is crucial towards maintaining an SOA's greatest advantages such as agility and integration. These must not be lost again by imposing an inflexible and heterogeneous security architecture on top of the SOA.

2.5 SOA Architecture Description and Evaluation

The business requirements and goals of an enterprise are the determining factors in the development of new applications. They define not only the functionalities of a new product but also characterize its non-functional qualities, e.g., availability, usability, performance, or security. All these requirements have to be identified primarily in the development life-cycle [BCK03].

The architecture of the new product is the first artefact that models both its functional and the non-functional characteristics. It serves as the first document for communication between customers and developers for clarifying the properties of the new application. Thus, it influences the success of the product from the very beginning of its life-cycle. Since it is used as means of communication between people from different domains the architecture has to be described in a language undestandable for all stakeholders. One of the most common modelling languages nowadays is UML (Unified Modelling Language) [Kre03]. However, not all of its constructs are easy understandable and comprehensible. Moreover, the UML constructs are insufficient for the modelling of domain specific characteristics. For this reason many different domains have defined their own architecture description languages (ADL). A state-of-the-art analysis has returned only two languages (C2SADL and Darwin) in the large set of ADL which support the modeling of highly-distributed dynamic systems like SOA [KRB07]. Yet, neither C2SADL nor Darwin is sufficient for describing SOA. C2SADL allows the modeling of communication between components only in layers and Darwin supports only constrained dynamism – the modeling of workflows at run-time is impossible – every component has to be known a priori [MRT99, MK96]. An ADL spezialized definition for the SOA domain, which is understandable for all stakeholders involved in the development of service-oriented applications, should be a general future research interest.

The architectural structure determines the quality and thus the final success of the new products. An important step for the early estimation of the quality of every new software solution is the evaluation of the architecture. Software engineering has already realised this need and there exist many software architecture evaluating methods, i.e., SAAM, ATAM, CBAM and many more [CKK02]. Some of them analyze specific quality characteristics, others evaluate also the trade-offs between the separate quality attributes. The three mentiond methods – SAAM, ATAM, and CBAM – are all scenario-based and allow the evaluation of the software architecture at any phase of the development process. SAAM (Software Architecture Analysis Method) is a simple and easy to learn method that can be used for the evaluation of single architectures or for the comparison of two or more architecture candidates. Although it was created to estimate the modifiability of an architecture, it has also been proven to be good for the estimation of other quality attributes such as portability, extensibility, integrability, and functional coverage

[CKK02]. SAAM indicates areas of high potential complexity, but it does not analyse the tradeoffs between the different quality attributes. This feature is covered by ATAM (Architecture Trade-off Analysis Method) which is a successor of SAAM. Thus, ATAM is one of the most widespread evaluation methods applied in practice. Unlike SAAM and ATAM that consider the design decisions with respect to architectural quality attributes, CBAM (Cost-Benefit Analysis Method) estimates the return on investment by analysing the costs, benefits, and schedule implications of the architectural decisions [BCK03]. However, because of the highly-distributed character of SOA the application of the existing evaluation methods on SOA is yet insufficient for analysing all of the consequences of the selected architectural configuration on the quality of the future application. Since services in a process can cross organizational boundaries and be offered by many different providers it is not that simple to analyse the quality of every service in a workflow with the present evaluation algorithms. An adaptation of the existent methods to the SOA approach is also a goal of our research.

2.6 SOA Governance

The SOA paradigm describes a way to realize agile implementations of business structures being able to flexibly adjust to changing environments. However, SOA introduces new challenges. Although it reduces a company's heterogeneity in IT to one enterprise architecture type, it introduces a new kind of complexity – numerous SOA services. While services as the smallest parts of SOA systems provide the appropriate means to enable an enterprise architecture to flexibly adjust to changing business processes, at the same time, they implicitly introduce system complexity. The homogenization and control of this emerging complexity is the central challenge to a SOA governance approach.

Scientific literature on SOA governance is scarce. Software vendors provide papers with strong marketing perspectives. In the discussion on the classification of SOA governance in the scope of corporate governance, most authors agree that it is a subset [Web06], extension [HPG06, Woo06] or specialization [ScS07] of IT Governance. Although SOA governance addresses special SOA-related issues, such as the issues service ownership or cross-company service deployment, it is still a part of the IT in an enterprise. Hence, IT governance mechanisms apply to an SOA.

A short definition of SOA governance is given by Windley [Win06]: "The development and enforcement of SOA policies and procedures goes by the name SOA governance." Main parts of SOA governance are the definition of defaults for organizational structure, standards, roles, responsibilities, policies, and measures that ensure transparency and conformity of the SOA [ScS07, SBB+06]. According to Marks and Bell, two general parts of SOA governance can be distinguished: an organizational and a process level [MB06]: The *organization* part of SOA governance defines the organizational structure of SOA and how it can be implemented in the existing corporate organizational structure. This includes the definition of roles and responsibilities that are important in the context of an SOA. An example could be the implementation of an SOA competence center that executes centralized all SOA activities [ScS07]. The *processes* part consists of all procedures for managing SOA processes and activities, including design, development, publishing, and maintenance. Additionally, respective responsibilities and roles have to be defined. Marks and Bell identify design-time governance, publishing and discovery governance, and run-time governance as phases of SOA governance processes [MB06].

Concerning practical realization, there are a number of different approaches for SOA governance. However, being based on different perspectives of SOA and its governance needs, they notably differ in scope and capability. A short overview and comparison is given in [NERS08]. However, a number of elements can be identified that all of them have in common:

- SOA governance goals are mostly derived from goals defined in IT governance frameworks. For SOA, *compliance to internal, normative, and legal regulations*, optimal *alignment of IT to business processes* (business-IT alignment) and *reliable long-term operation* are considered the most important goals.
- Common to all approaches is the *organizational integration*. Fabini proposes a “Center of Excellence” (CoE) that coordinates and “runs” the SOA governance [Fab07]. It consists of representatives of all parts of the company and enacts policies that are to be applied to the system [Kal07, BBF+06, Afs07, Sof05]. This also implies the redefinition of accountabilities and decision rights, following the standard definition of IT governance by Weill and Ross [WR04].
- As stated above, *policies* are the central means that support a governance model. All restrictions, regulations and guidelines including the target groups, processes, and systems are defined as so-called policies. Policies can be defined in numerous domains, such as architecture, security, organization, a.s.o. On its classifications in literature no consensus has been reached so far [BK05, Sof05].
- *Best practices* are considered the foundation of the majority of the governance models. Mostly they are described as catalogs where gained experience is collected [Afs07].

- SOA maturity models independently assess the overall state of a SOA system. They are used to provide general feedback to the CoE and trigger abolishment, creation or adjustment of governance policies [JG07].

Summarizing, SOA governance is a wide, mostly “unexplored” research field. A common structure, core elements, concrete control circles or supporting technical mechanisms have not yet broadly been investigated. Ongoing and future research focuses on general models for SOA governance, governance policy enforcement means, and automated compliance verification techniques, i.e., process verification.

3 Conclusions and Outlook

SOA is a paradigm that enables enterprises to deal with the integration of heterogeneous systems and to achieve highly flexible workflows. Recent surveys have shown that SOA is gaining importance for enterprises, thus strengthening the foundation for SOA applications and research in the field.

This paper presented selected topics in service engineering and management for enterprise systems, discussing various topics on different layers of abstraction and linking the self-contained sections via their respective positions in the SOA pyramid (as depcited in Figure 2). For each of these topics its necessity was shown in the context of cross-organizational, service-based workflows and current research results as well as remaining future challenges were identified. These challenges should be considered to be important and industry-relevant for reducing the gap between the business side and the technology side which is still existing in many enterprises.

But research should not focus on the challenges alone. It is at least equally important to integrate and consolidate the results achieved so far and to get them to work together. The main reason for this is that most challenges are not likely to occur in isolation but rather in highly heterogeneous and interwoven scenarios, a typical condition for enterprise systems.

References

[Afs07] Afshar, M.: SOA Governance: Framework and Best Practices, Oracle Whitepaper, May 2007, http://www.oracle.com/technologies/soa/docs/oracle-soa-governance-best-practices.pdf (last access: October 12, 2008).

[BBF+06] Bieberstein, N.; Bose, S.; Fiammante, M.; Jones, K.; Shah, R.: Service-Oriented Architecture (SOA) Compass - Business Value, Planning, and Enterprise Roadmap. IBM developerWorks, 2006.

[BCK03] Bass, L.; Clements, P.; Kazman, R.: Software Architecture in Practice, 2nd edition, Addison-Wesley, 2003.

[BG05] Baresi, L.; Guinea, S.: Towards Dynamic Monitoring of WSBPEL Processes, in: Proceedings of the 3rd International Conference on Service Oriented Computing, 2005, pp. 269-282.

[BGR+05] Berbner, R.; Grollius, T.; Repp, N.; Heckmann, O.; Ortner, E.; Steinmetz, R.: An Approach for Management of Service-oriented Architecture (SoA)-based Application Systems, in: Proceedings of the Workshop Enterprise Modelling and Information Systems Architectures (EMISA 2005), Klagenfurt, Austria, 2005, pp. 208-221.

[BK05] Brauer, B.; Kline, S.: SOA Governance: A Key Ingredient of the Adaptive Enterprise, HP Whitepaper, February 2005, http://www.managementsoftware.hp.com/products/soa/swp/soa_swp_governance.pdf (last access: October 12, 2008).

[BSR+06] Berbner, R.; Spahn, M.; Repp, N.; Heckmann, O.; Steinmetz, R.: Heuristics for QoS-aware Web Service Composition, in: 4th IEEE International Conference on Web Services (ICWS 2006), pages 72–79, 2006.

[CKK02] Clements, P.; Kazman, R.; Klein, M.: Evaluating Software Architectures Methods and Case Studies, Addison-Wesley, 2002

[CPE+05] Canfora, G.; Penta, M. D.; Esposito, R.; Villani, M. L.: An Approach for QoS-Aware Service Composition Based on Genetic Algorithms, in: Proceedings of Genetic and Evolutionary Computation Conference, pages 1069–1075, 2005.

[ESN+08] Eckert, J.; Schulte, S.; Niemann, M.; Repp, N.; Steinmetz, R.: Worst-Case Workflow Performance Optimization, in: 3rd International Conference on Internet and Web Applications and Services (ICIW 2008), 2008.

[ESR+08] Eckert, J.; Schulte, S.; Repp, N.; Berbner, R.; Steinmetz, R.: Queuing-based Capacity Planning Approach for Web Service Workflows Using Optimization Algorithms, in: IEEE

International Conference on Digital Ecosystems and Technologies (DEST 2008), 2008.

[Fab07] Fabini, M.: Governance für komplexe SOA Unternehmungen: Eine Vision für das Schweizer Gesundheitswesen, in: Gernot Starke and Stefan Tilkov (editors), SOA-Expertenwissen, Chapter SOA-Governance, pages 309–323. dpunkt.verlag Heidelberg, 2007.

[FL07] Farrell, J.; Lausen, H.: Semantic Annotations for WSDL and XML Schema, WC3 Recommendation, 2007, http://www.w3.org/TR/2007/REC-sawsdl-20070828/ (last access: October 12, 2008).

[HPG06] Holley, K.; Palistrant, J.; Graham, S.: Effective SOA Governance, IBM Whitepaper, March 2006, ftp://ftp.software.ibm.com/software/uk/flexible/wp/effective_soa_governance.pdf (last access: October 12, 2008).

[JG07] Johannsen, W.; Goeken, M.: Referenzmodelle für IT-Governance - Strategische Effektivität und Effizienz mit COBIT, ITIL & Co, dpunkt.verlag Heidelberg, 1. Edition, 2007.

[Jos07] Josuttis, N. M.: SOA in Practice: The Art of Distributed System Design (Theory in Practice) O'Reilly Media, Inc., 2007.

[Kal07] Ulrich Kalex. Von der Geschäftsarchitektur zur SOA Governance, in: Gernot Starke and Stefan Tilkov (editors), SOA-Expertenwissen, pages 325–340. dpunkt.verlag Heidelberg, 2007.

[KBS04] Krafzig, D.; Banke, K.; Slama, D.: Enterprise SOA: Service-Oriented Architecture Best Practices (The Coad Series), Prentice Hall PTR, 2004.

[KRB07] Kabzeva, A.; Repp, N.; Berbner, R.; Müller, P.; Steinmetz, R.: Software Architecture – Description Languages and Evaluation Methods, Technical Report, Technische Universität Darmstadt, KOM, August 2007.

[Kre03] Krebs, J.: Sag's einfach mit Diagrammen: Visuelles Modellieren, ObjektSpektrum, Januar 2003.

[LAP06] Lazovik, A.; Aiello, M.; Papazoglou, M.: Planning and Monitoring the Execution of Web Service Requests, International Journal on Digital Libraries, vol. 6, no. 3, 2006, pp. 235-246.

[LDK04] Ludwig, H.; Dan, A.; Kearney, R.: Cremona: An Architecture and Library for Creation and Monitoring of WS-Agreements", in: Proceedings of the 2nd International Conference on Service Oriented Computing, 2004, pp. 65-74.

[LY05] Lin, K.-J.; Yu, T.: Service Selection Algorithms for Composing Complex Services with Multiple QoS Constraints, in: 3rd International Conference on Service-oriented Computing (ICSOC 2005), 2005.

[MK96] Magee, J.; Kramer, J.: Dynamic Structure in Software Architectures, in: Proceedings of the 4th ACM SIGSOFT Symposium on Foundations of Software Engineering, San Francisco, California, United States, October 1996.

[MB06] Marks, E.; Bell, M.: Service-Oriented Architecture: A Planning and Implementation Guide for Business and Technology, John Wiley & Sons, Inc., New Jersey, 2006.

[MBM+07] Martin, D.; Burstein, M.; McDermott, D.; McIlraith, S.; Paolucci, M.; Sycara, K.; McGuinness, D. L.; Sirin, E.; Srinivasan, N.: Bringing Semantics to Web services with OWL-S, World Wide Web, vol. 10, iss. 3, September 2007, pp. 243-277.

[MRT99] Medovici, N.; Rosenblum, D.S.; Taylor, R. N.: A Language and Environment for Architecture-Based Software Development and Evolution, in: Proceedings of the 21st International Conference on Software Engineering, Los Angeles, CA, May 1999.

[MSZ01] McIlraith, S. A.; Son, T. C.; Zeng, H.: Semantic Web services, IEEE Intelligent Systems, vol. 16, no. 2, pp. 46-53, 2001.

[NT07] Naseri, M.; Towhidi, A.: QoS-Aware Automatic Composition of Web Services using AI Planners, in: Proceedings Second International Conference on Internet and Web Applications and Services, 2007.

[NERS08] Niemann, M.; Eckert, J.; Repp, J.; Steinmetz, R.: Towards a Generic Governance Model for Service-Oriented Architectures, in: Proceedings of the Fourteenth Americas Conference on Information Systems (AMCIS 2008), Toronto, ON, Canada, 2008, Aug 2008.

[NL04] Newcomer, E.; Lomow, G.: Understanding SOA with Web Services (Independent Technology Guides), Addison-Wesley Professional, 2004.

[OF07] Orleans, L. F.; Furtado, P. N.:Optimization for QoS on Web-Service-Based Systems with Task Deadlines, in: Third International Conference ion Autonomous Systems (ICAS 2007), 2007.

[Pap03] Papazoglou, M. P.: Service-Oriented Computing: Concepts, Characteristics and Directions, in: Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE 2003), IEEE Computer Society, Washington, DC, 2003, pp. 3-12.

[PYY+04] Peng, D.; Yuan, Y.; Yue, K.;Wang, X.; Zhou, A.: Capacity Planning for Composite Web Services Using Queuing Network-Based Models, LNCS (2004) 3129, pages 439–448, 2004.

[Rep08] Repp, N.: Monitoring of Services in Distributed Workflows. Doctoral Consortium of the 3rd International Conference on Software and Data Technologies (ICSOFT), 2008.

[Rob05] Robinson, W.N.: A Requirements Monitoring Framework for Enterprise Systems, Journal of Requirements Engineering, vol.11, no. 1, 2005, pp.17-41.

[SBB+06] Shah, R.; Bieberstein, N.; Bose, S.; Fiammante, M.; Jones, K.: SOA Project Planning Aspects, 2006, http://websphere.sys-con.com/read/168398.htm (last access: October 12, 2008).

[SC08] SOA Check 2008, http://www.soa-check.eu/ (last access: October 12, 2008).

[Sch03] Schumacher, M. Security Engineering with Patterns: Origins, Theoretical Models, and New Applications (Lecture Notes in Computer Science), Springer, 2003

[ScS07] Schelp, J.; Stutz, M.: SOA-Governance, in: Hans-Peter Fröschle and Stefan Reinheimer (editors), Serviceorientierte Architekturen (SOA), number 253 in HMD Praxis der Wirtschaftsinformatik, pages 1–10. dpunkt verlag, February 2007.

[SER+08] Schulte, S.; Eckert, J.; Repp, N.; Steinmetz, R.: An Approach to Evaluate and Enhance the Retrieval of Semantic Web Services, in: Proceedings of the 5th International Conference on Service Systems and Service Management (ICSSSM'08), IEEE, June 2008.

[SM06] Spanoudakis, G.; Mahbub, K.: Non Intrusive Monitoring of Service Based Systems, International Journal of Cooperative Information Systems, vol. 15, no. 3, 2006, pp. 325-358.

[Sof05] Software AG. SOA Governance - Rule Your SOA, 2005, http://www.softwareag.com/de/Images/WP_SOA_Governance_D_tcm17-22130.pdf (last access: October 12, 2008).

[SSK08] Bundesamt für Sicherheit in der Informationstechnik. SOA-Security-Kompendium: Sicherheit in Service-orientierten Architekturen, 2008, http://www.bsi.bund.de/literat/studien/soa/index.htm (last access: October 12, 2008).

[Web06] WebMethods. SOA Governance - Enabling Sustainable Success with SOA, October 2006, http://www1.webmethods.com/PDF/whitepapers/SOA_Governance.pdf (last access: October 12, 2008).

[Win06] Windley, P.J.: SOA Governance: Rules of the Game, INFOWORLD.COM, 01.23.06:29–35, 2006.

[Woo06] Woolf, B.: Introduction to SOA Governance, IBM Developer Works, June 2006, http://www.ibm.com/developerworks/library/ar-servgov/ (last access: October 12, 2008).

[WR04] Weill, P; Ross, J.W.: IT Governance - How Top Performers Manage IT Decision Rights for Superior Results. Harvard Business School Press, Cambridge, MA, 2004.