Foundations of a Reference Model for SOA Governance

Christian Ott[1], Axel Korthaus[2], Tilo Böhmann[3],
Michael Rosemann[2], and Helmut Krcmar[1]

[1] Technische Universität München, Lehrstuhl für Wirtschaftsinformatik, München, Germany
[2] Queensland University of Technology, Business Process Management, Brisbane, Australia
[3] ISS International Business School of Service Management, Hamburg, Germany
christian@coonet.de, axel.korthaus@qut.edu.au,
boehmann@iss-hamburg.de, m.rosemann@qut.edu.au,
krcmar@in.tum.de

Abstract. Although the lack of elaborate governance mechanisms is often seen as the main reason for failures of SOA projects, SOA governance is still very low in maturity. In this paper, we follow a design science approach to address this drawback by presenting a framework that can guide organisations in implementing a governance approach for SOA more successfully. We have reviewed the highly advanced IT governance frameworks Cobit and ITIL and mapped them to the SOA domain. The resulting blueprint for a SOA governance framework was refined based on a detailed literature review, expert interviews and a practical application in a government organisation. The proposed framework stresses the need for business representatives to get involved in SOA decisions and to define benefits ownership for services.

Keywords: Service-Oriented Architecture (SOA), SOA governance.

1 Introduction

Governance has been seen as one of the key success factors of IT for many years and enterprises currently invest considerable resources into the implementation of IT governance frameworks such as Cobit [1, 2]. In their seminal work, [3] define IT governance as the process of "specifying the decision rights and accountability framework to encourage desirable behaviour in the use of IT." The purpose of such a decision rights and accountability framework is to address the three basic questions of IT governance: "What decisions must be made to ensure effective management and use of IT?", "Who should make this decisions?" and "How will these decisions be made and monitored?" [3]. Many enterprises presently face the challenge of developing adequate governance mechanisms for Service-Oriented Architectures (SOAs), which introduce new complexities due to the amount of services to be managed [4]. The SOA paradigm has become widespread and is often considered an important concept to drive the evolution towards an IT architecture focusing on business processes, flexibility and reuse [5, 6, 7]. Moreover, some proponents envision that organisations will begin to open up their architecture to their business ecosystem, i.e. their network of customers, suppliers and even competitors,

achieving increased interoperability, i.e. the ability to exchange information and to use the information that has been exchanged, through the use of open standards as postulated by the SOA paradigm [8, 9]. The decomposition of today's business applications into reusable business process components that may be marketed to external customers creates novel challenges for IT governance. To date, however, no widely accepted framework for SOA governance has emerged [4]. Given that the lack of a comprehensive governance approach has been cited as the most common reason for failures of post-pilot SOA projects [10], work in this area is highly relevant.

Notwithstanding the urgent need, delineating SOA governance and building a corresponding framework is not an easy task. Already the question of how SOA governance relates to IT governance lacks a consistent answer. While for Malinverno, "SOA governance isn't simply a subset of IT governance" [11], some authors do make this very assumption [12]. For others, SOA governance is an "extension" [13] or "specialisation" [14] of IT governance. In spite of the discussions about a precise definition of the term SOA governance, most authors agree on the basic elements a governance framework should address, namely the organisational structure, processes, policies and metrics [14, 15, 16]. To provide a working definition for the rest of this paper, we build on definitions in [4] and [17]:

SOA governance focuses on the decisions across the entire service lifecycle to enable organisations to realise the benefits of SOA. It is an approach to exercising control and mitigating risk by establishing organisational structures, processes, policies and metrics suitable to ensure that the adoption, implementation, operation and evolution of an SOA is in line with the organisation's strategies and objectives and complies with laws, regulations and best practices.

For reasons of scope, we concentrate on the organisational aspects in this paper by deriving a set of activities and roles that are required in an SOA context and by proposing their responsibilities along the service lifecycle. The resulting framework can guide organisations in designing or evaluating their own governance structure.

The paper is structured as follows. In sections 2 and 3, we point to related work and explicate our research approach. Section 4 outlines the identified activities along the service lifecycle. Section 5 describes the roles involved, to which responsibilities are assigned in section 6. Section 7 includes lessons learned from an application of the framework in a case study. The paper concludes with summary and further research opportunities in section 8.

2   Related Work

The knowledge bases of corporate and IT governance form obvious points of references for research into SOA governance. While from an IT governance perspective, standard works like [3] and well-received frameworks such as Cobit [1] and ITIL [18] are the most prominent examples, the OECD Principles of Corporate Governance are among the most influential guidelines in the area of corporate governance [19].

Academic literature on SOA governance (e.g. [4, 14]) is still relatively scarce, whereas numerous IT solution vendors and analysts have addressed the topic in recent years (e.g. [10, 11, 20, 21, 25, 26, 27]). SOA governance solutions presented by IT vendors tend to address only fragments of a holistic approach. While some IT vendors propose advanced governance models that attend to design-time governance, organisational and business aspects, many of them focus on technical management and run-time governance, i.e. the governance of services that are already in production, and do not address the whole service lifecycle including aspects of adequate design-time governance and higher level corporate governance.

Open standards organisations such as OASIS, OMG and The Open Group have produced a large amount of technical specifications and standards on the subject of SOA, often overlapping in contents. Kreger and Estefan [25] give an overview of the documents for SOA reference models and ontologies, reference architectures, maturity models, SOA modelling profiles, and open standards related to the topic of SOA governance. Not only is the OMG SOA Governance RFP development group [26] exploring the standardisation of SOA governance, the topic has also been included as a chapter in the OASIS Reference Architecture for SOA Foundation [27], and a SOA governance framework is being developed by The Open Group (SOA Governance Framework [28]). These consortia address the topic from different perspectives, and the most promising for the research focus taken in this paper, i.e. organisational aspects of SOA governance, is the work of The Open Group. Their SOA Governance framework, which as at December 2009 has draft status only, however still lacks essential elements such as a specification of detailed accountabilities of roles along the service life cycle.

Bernhardt and Seese [4] propose a conceptual SOA governance framework striving to cover the complete SOA lifecycle. Their approach differs from the one taken in this paper as it uses the standardised OASIS SOA reference model [29] as a starting point for the identification of SOA governance aspects to be considered. They do not make use of empirically tested best practises from related IT governance literature, whereas the framework we propose is primarily derived from the much wider area of successful IT governance frameworks in order to leverage existing knowledge and revise it against the background of SOA-specific characteristics. Bernhardt and Seese [4] have not yet investigated the relationships between their approach and these common IT frameworks.

3   Research Approach

The SOA governance reference framework partly presented in this paper is a "design artefact" in the sense of the design science-based approach to IS research as described in [30]. According to them, IS research is concerned with two design processes, i.e.

- to 'build' purposeful artefacts to address heretofore unsolved problems, and
- to 'evaluate' these artefacts with respect to the utility provided in solving those problems (cf. Fig. 1).
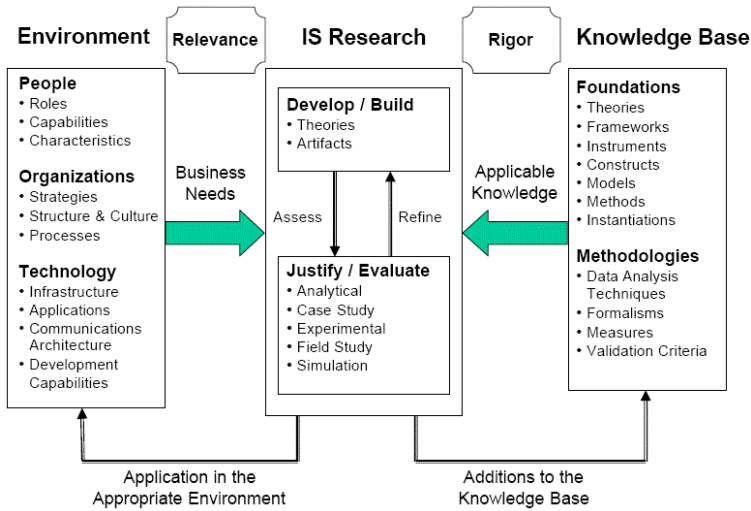
Fig. 1. Information Systems Research Framework according to [30]

Hence, as opposed to behavioural science, design science aims at providing utility and relevance to practice by innovatively designing an artefact that meets an existing business need or "problem" [30]. With regard to the work on SOA governance presented here, the claim of relevance to practice can be justified not only through the Gartner Research study mentioned earlier [10], but also, for example, through a recent SOA governance user survey by Software AG [21]. This survey indicates that most users view SOA governance as important, acknowledge the need for improvement and emphasise the demand for a holistic, business objective-driven lifecycle approach from the start. Rigour in the research process has to be assured by the appropriate application of existing foundations from the knowledge base of the field in the 'build' phase and of suitable methodologies in the 'evaluate' phase [30].

Starting from the existing knowledge base in the 'build' phase of the proposed SOA governance framework, we analysed the widely-used IT governance frameworks Cobit and ITIL and provided an initial evaluation of its utility in a case study in order to derive the core of the SOA governance framework. We selected Cobit and ITIL for the following reasons. Cobit has practically become the global de facto standard for IT control and governance, and most frameworks somewhat align with. ITIL, on the other hand, is a most prominent IT Management framework that primarily defines management and support processes. While ITIL primarily addresses IT efficiency that relates to the effective operation of IT, Cobit is primarily addressing effectiveness and strategy of IT in the context of an organisation, where effectiveness relates to producing a decided, decisive, or desired effect and strategy relates to the strategic planning and adaptation (e.g. of structure or behaviour) that serves the core function of IT to contribute to desired business outcomes.

Mapping the roles and activities proposed by the two frameworks to an SOA environment revealed a need for extensions, as some criteria that are specific to SOAs are not covered in these two popular frameworks (cf. sections 4, 5 and 6). Furthermore, this mapping necessitated a re-naming and re-grouping of activities into a service

lifecycle. In a second step, we conducted a detailed review of literature related to service lifecycle management and SOA governance. Academic articles as well as industry white papers about SOA roles and responsibilities are scarce and often use diverging terminologies or present ideas in an unstructured way, so we focused on the identification of main concepts. We also conducted a series of interviews with carefully selected experts in the field of service management. For the identification of the relevant roles and their responsibilities, we conducted a comprehensive content analysis using published job profiles from Seek.com, Australia's leading recruitment website.

In order to critically evaluate the utility of the framework, we applied it at Landgate, a public sector organisation. Landgate is the Statutory Authority responsible for Western Australia's land and property information and seeks to evolve its IT business applications to implement new services for its clients and to collaborate more closely with partners. The application of the governance framework to Landgate showed how the model supports organisations in identifying new IT management activities when moving into a service-oriented paradigm and which consequences this new paradigm has for the establishment of accountabilities.

4   The Service Lifecycle

4.1   Overview

Cobit and ITIL are very detailed and widely used frameworks that propose a large number of best practises and processes as well as measures, roles and responsibilities to aid management in the planning and organisation, acquisition and implementation, delivery and support, operation, monitoring and evaluation of IT systems. In Cobit alone, there are 197 single steps grouped in 34 processes, which are part of 4 main phases, offering an extensive repository of relevant activities and a highly elaborated set of assignments to roles. Some of the issues covered, such as infrastructure, data or technology and support, will not change significantly independent of the underlying paradigm (e.g. when SOA is replaced by another IT design paradigm) and therefore have not been further analysed. Besides that, the structures of Cobit and ITIL do not allow for an explicit representation of different decision levels. Thus, we looked at management models to find a suitable high-level structure. Drawing from IT-management, we suggest that decision rights can be distributed into distinct layers. Among these, strategy management, portfolio management, program management and project management are mentioned by most authors [31, 32]. Furthermore, operations management had to be considered as well, since governance is not just relevant during the identification and development of services, but for operating services as well (run-time governance). Due to space constraints, this paper covers only three of the five layers (shaded in Fig. 2): service portfolio-, service project- and service operation management.

While acknowledging that there is a broad variety of definitions, we agree with [33] who stress that portfolio management deals with selecting and prioritising the best projects to proceed with. Portfolio management is about choosing the right project, whereas project management is about doing the project right [34]. Hence, in

the portfolio management stage of our proposed framework, the goal is to identify the most relevant services from a larger service portfolio and decide if and when to implement them. Program management, which we do not cover here, represents the connecting link between the two and aligns strategy and execution to deliver the whole SOA by managing interdependent projects [32]. Once a business sponsor has been identified and accepts responsibility for the service, a project is started and the service can be developed. The development process and the publishing or deployment of the service are governed in the service project management stage. Once in place, the operation management of a service covers operation and use, including performance and change management, as well as the retirement phase.
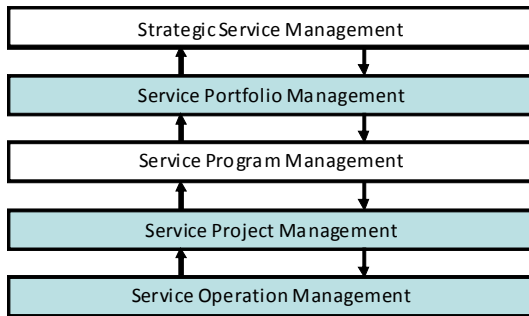

Fig. 2. Layers of management comprising decisions relevant for SOA governance (adapted from [31]). The layers covered in this paper are shaded.

A significant amount of research has been published regarding the lifecycle of a single service (cf. [35] for a comprehensive overview) with a more or less common understanding of what should be part of it. Starting with a service analysis and design phase, most authors include service implementation, service publishing, service operation as well as service retirement or withdrawal. In addition to that, [35] mention a negotiation phase. The latter is primarily relevant if a service or part of its sub-services are provided or sourced externally. For many organisations using SOA today, this is not yet an option but will become more important once emerging service brokers have leveraged the discovery of available services and provide required functions such as pricing and contracting [8]. From today's perspective, it is also not clear to what extent bargaining will happen at all, or if, for example, prices and quality standards are specified solely by the service broker. For these reasons, we have not yet considered negotiation activities in this paper.

4.2 Detailed View

In this section, we focus on the main differences as compared to traditional IT governance by introducing new activities that provide managers with a foundation upon which SOA-related decisions can be based and by discussing those that require changes. Fig. 3 gives an overview and shows how management layers, lifecycle stages and activities are interrelated. The first two columns in Fig. 3 map the three management layers introduced above (service portfolio management, service project

management and service operations management) to the main steps of the service lifecycle (service analysis, service design, service implementation, service publishing, service operation and service retirement). Service analysis occurs both on a portfolio level, e.g. to determine which new services to add to the service portfolio, and on the service project level in order to define the requirements for a single service to be built in the course of the service project. Service design, implementation and publishing are all part of a service project and therefore can be mapped to the service project management level. Service operation and retirement are mapped to the service operations management layer here, although service retirement could also be seen as belonging to the service portfolio view, as it typically requires the assessment of the service to be retired in relation to the other services in the portfolio. The third column of Fig. 3 refers to SOA-specific activities that differ from conventional IT governance-related activities. These activities are part of the service lifecycle step they are aligned with in the diagram (column two). In the following, Fig. 3 will be discussed in detail.

Management Layers	Service Lifecycle	SOA specific Governance Activities		
Service Portfolio Management	Service Analysis	Create a SOA roadmap	Assure the consultation of potential users of services	Find business sponsor / service owner
Service Project Management	Service Design	Decide on granularity and orchestration		
	Service Implementation			
	Service Publishing	Determine access rights	Develop pricing model	
Service Operation Management	Service Operation	Develop and implement a process to consistently record, assess and prioritise change requests		
	Service Retirement			

Fig. 3. Interrelationship of management layers, lifecycle stages and the activities addressed in this paper

4.2.1 Service Portfolio Management

As a first step within the service portfolio management phase, a service roadmap is developed by identifying and prioritising service candidates (e.g. by analysing business processes). The proposed services are subsequently analysed further. In this step, all potential users should contribute to the definition of requirements to ensure high reusability of the service. After the feasibility study has yielded a positive outcome and a business case has been developed, identifying a business sponsor who is willing to fund the development and operation of the service [36] is an essential activity before a project can be started. Besides that, portfolio management is also responsible for the development of an overarching service taxonomy and service

descriptions as well as for monitoring across projects. The following items represent a (certainly not exhaustive) list of Cobit activities that need to be adapted to a SOA environment:

- *Create an SOA roadmap:* The implementation of an SOA requires a significant change of both the IT landscape and the mindset of business and IT people within an organisation [37]. Due to time and budget constraints, in most cases a gradual transition towards SOA is more likely to be successful than a "big-bang" implementation. Guidance on where to start and which subsequent steps to follow is therefore essential and should be provided by creating a suitable SOA roadmap. This roadmap suggests a certain sequence in which proposed services should be analysed and developed. Identification and prioritisation of services are therefore key elements of creating a roadmap. Service identification can be conducted in top-down approaches, such as capability analysis or domain decomposition [38], or more bottom-up as in tracing business processes [36]. Prioritisation should be based on estimated business value, reuse potential (e.g. by implementing business process patterns) and IT complexity reduction potential [36]. In many real world organisations, however, services are created out of "immediate needs" (cf. section 5) either due to a lack of coordination between business and IT or simply out of aiming at short term returns. This is not surprising, as budget constraints or other obstacles may prevent a detailed analysis at this stage. In these cases, an evolutionary approach [36] can be helpful, meaning that smaller IT projects with positive business cases are defined that comply with a target application landscape as well. This will balance both short-term financial results and long-term efficiency of the SOA. We believe that the quality of the roadmap will be a crucial determining factor for the effectiveness of the whole SOA investment. The formulation of a SOA roadmap should therefore be seen as a core activity within the governance approach.
- *Assure the consultation of potential users of services:* As suggested by Cobit, all stakeholders should be included in the process (e.g. for determining requirements or assessing risks). In an SOA environment, the consultation of stakeholders becomes a common, yet more complex task, as aiming for reusability of services on a broad basis is seen as one of the core characteristics of an SOA [11, 37]. Nevertheless, many SOA initiatives fail to leverage reuse and therefore do not yield the expected financial results, leading to a drop of management support [11]. While consulting potential users is crucial to the realisation of the expected benefits, it requires a solid ground of knowing who the potential users are, putting even more emphasis on the service identification step.
- *Find business sponsor / service owner:* Another important step refers to the issue of funding [39]. Adapting services to the requirements of different users will be more expensive than developing them for the sole purpose of a single user [40]. In many cases, the benefits might outweigh the cost so that a mechanism is required for identifying those services that are worth adapting. This mechanism, however, cannot make a perfect distinction, as there is uncertainty involved in the estimation of development and maintenance cost and possible revenues. Considering this, an enterprise architect (see section 5) can identify potential users, help them express their needs and recommend a certain design of a service, but should not appoint a

business sponsor or owner. The latter should be found in a less hierarchical manner, because to enable performance measurement and encourage a high quality of decision making, the holder of the decision right should bear the economic risk as well. As multiple ownership would cause an increase in coordination effort, it will be helpful if services are owned by one of the potential users. The enterprise architect can encourage this by promoting a business case for the adapted service. If none of the potential users is willing to sponsor the service, the enterprise architect or another centralised committee could ultimately own the service as well and should therefore be provided with a dedicated budget.

4.2.2 Service Project Management

Most steps of the basic service lifecycle, as mentioned above, are part of service project management. These include analysis, design, implementation and deployment/ publishing. The analysis phase is fragmented, as this task is to a large extent conducted in the portfolio management phase, before a service sponsor can be found. In this paper, we focus on particularly interesting differences compared to traditional software development. We located them in the following activities that belong to the service lifecycle phases "Service Design" and "Service Publishing":

- *Decide on granularity and orchestration:* Although an initial analysis is conducted within the portfolio phase, different options remain for the realisation of the required functionality after a service project has been started. Sub-services that are available from the internal repository or could be bought from a service broker can serve as building blocks and reduce development cost. On the other hand, a finer granularity of service components than proposed by the requirements of internal users might also help promote services and sell them to external customers. Consequently, an optimal level of granularity is no longer just subject to technical requirements but also to market supply and demand. Thus, the availability of and the demand for services both externally and internally determines how fine or coarse a service should be and how atomic services can be combined into molecular services. This is referred to as the "economic level of granularity" [40].
- *Determine access rights:* Before a service is published, access rights need to be specified. This does not just refer to users within the organisation, but, in contrast to traditional software development, also to potential external customers. This is a strategic decision, for if cutting-edge knowledge is made accessible to competitors, comparative advantages might be lost. Therefore, key executives should be responsible for this decision.
- *Develop pricing model:* Among traditional IT cost accounting methods (for an overview see [41]), activity-based costing is seen as one of the most effective representatives [42]. Under the SOA reuse paradigm, where services are shared among several business units or departments, new mechanisms like negotiation [41] between service owners and consumers should be considered. In addition, a pricing model for the external market has to be developed if the service is also offered to external customers. It differs from the internal pricing model as it does not aim at discouraging over- or underutilisation, but aims at maximising profit.

As in the context of Service Portfolio Management, this list should not be considered complete, as there are other important aspects, such as issues regarding service contracts, Service Level Agreements (SLAs), business object governance etc. that require specific approaches.

4.2.3 Service Operation Management

Within operation management, the actual service operation, which involves activities such as training, monitoring of Service Level Agreements (SLAs) and change management, as well as the retirement phase are governed. Incident and capacity management have not been included in the service operation phase as they are not service-specific. Retirement is a responsibility of the portfolio manager; however, it strongly affects the service owner as well. It could therefore be included in the portfolio management phase as well as in the operation management phase. One main difference compared to traditional software development refers to change management. We describe the following activity in detail:

- *Develop and implement a process to consistently record, assess and prioritise change requests:* The change management process in an SOA is complex due to the distance between service providers and service consumers [39] and the high coordination effort that is required as every change affects not just the one who requested the change but the other users as well. Risk assessment should also consider side effects, because if the responses of a service are modified, other services that invoke the changed service may require changes as well [39]. Once the decision has been made and changes are authorised, all customers must be informed about the details and how their service usage requires adaptation.

5 Roles

Most of the roles proposed by Cobit (e.g. Board, CEO, CIO, CFO) are on a top management level. Additionally, architects, developers and operation managers are mentioned, but many roles that become relevant within an SOA environment are not included. Academic articles as well as industry white papers about SOA roles and responsibilities are sparse ([43] and [44] provide comprehensive frameworks, which, however, lack validation). SOA literature with a management or lifecycle focus mentions some additional roles, but mostly in an unstructured or anecdotal way [39]. Due to the lack of widely accepted terminology, definitions and descriptions, comparing or even consolidating different terms is not easy.

In this section, we give a brief overview of roles that are either not mentioned in Cobit or whose focus changes significantly under a SOA paradigm. We conducted a literature review and a comprehensive content analysis of more than 300 published job profiles at Seek.com (keyword: "SOA"), Australia's number one job site with over 100,000 jobs online. We believe that Seek.com provides a comprehensive source for an overview of the roles to be found in contemporary SOA projects in Australia. However, we did not crosscheck the findings by approaching major IT companies directly, which might have further refined the results. Here, we focus on defining the most important and accepted roles and show corresponding references.

- *Business Analyst* (Seek.com, [44, 45, 46]): The business analyst elicits domain knowledge. The business analyst understands the language of business users and providers and can translate the functional and non-functional requirements into processes and services. Among the business analyst's main responsibilities are the identification and analysis of services, but s/he is also consulted for the development of test cases.
- *Enterprise Architect* (Seek.com, [20, 36, 46, 47]): Within a traditional IT context, the enterprise architect focuses on the application of technology to increase operational effectiveness and efficiency, e.g. based on the identification of patterns in business processes [12]. In the more modern, holistic view, the enterprise architect integrates the business plan with the technical capabilities, e.g. by establishing technical policies. Within an SOA context, the enterprise architect is responsible for the development of an SOA framework and strategy. S/he ensures an optimal use as well as the performance level of services. The uptake of dedicated service architects is not visible yet.
- *Service Owner* [20, 36]: Although the service owner is mentioned as a key role, there is no definition of corresponding responsibilities and tasks in any of the literature or the published job profiles we reviewed. We define the service owner as the one who sponsors the development and operation of the service, in other terms, the benefits owner. This might be the business unit that launched the request or a centralised committee if none of the potential users is willing to fund the service or the organisation is structured hierarchically and business units or departments do not hold decision rights for the investment. As the one bearing the financial risk of the service project, the service owner must hold the right to determine a pricing model and "sell" it to other users as well as to make decisions about changes.
- *Service Librarian* [43, 48]: The service librarian is a new role in SOAs. The service librarian is responsible for the service repository and ensures the quality of published (meta-)data about as well as ease of discovery of and access to registered services.
- *Project Manager* [1, 43, 44, 46]: Compared to its traditional counterpart, an SOA project manager needs to plan for much shorter delivery cycles. This role is responsible for defining project plans, implementing the plans and monitoring the project as well as establishing the appropriate service-level agreements and resource usage. With an increased use of aggregated services (composed of other services), the relevance of this role will most likely rise.

6   Assignment of Responsibilities

The assignment of responsibilities calls for a detailed mapping of the involvement of the different roles in the activities of SOA governance. We use so-called RACI charts for each of the management layers in our proposed initial SOA governance framework to show the recommended responsibilities. The RACI charts map activities of the SOA lifecycle to roles of stakeholders in a SOA initiative and propose their responsibilities by specifying which roles are (r)esponsible, (a)ccountable, (c)onsulted or (i)nformed regarding specific activities. Roles are represented as columns and service lifecycle activities as rows. By providing these RACI charts, our framework offers a tangible and easy-to-apply tool for the analysis of responsibilities along the whole service lifecycle.

While a detailed discussion of the RACI charts is beyond the scope of this paper, two aspects of the assignment of responsibilities became particularly prominent. The first aspect is the involvement of top management and business executives in SOA development, the second aspect is the alignment of ownership for individual services. The involvement of business executives documents the degree to which the design of a service-oriented architecture is backed and driven by business concerns. In many organisations, SOA is seen as "yet another way" of software development. Consequently, few responsibilities have been changed since it was introduced. The business potential of this new paradigm is often not realised and SOA remains a means of integration for an organisation's software architecture. If this is to be changed, business representatives, especially business executives, have to be involved in decision making even more than proposed by Cobit for a traditional IT environment [1]. At first sight, this seems to increase the complexity of decision making, which would contradict executives' striving for reduction of information. Yet, management is not required to look at technical details but to understand the business implications. They can provide support for the development of interdepartmental services to leverage the reuse potential of SOA and promote the utilisation of services by selling them to external customers. Within the proposed framework, it is recommended that executives be involved in the development of an SOA roadmap and the prioritisation of services by evaluating the business potential and business value. Moreover, they can help find a business sponsor and should receive accountability for determining access rights. The business executives are expected to evaluate if a service contributes to the competitive advantage of the organisation, which could be lost once the service is offered to competitors.

Turning to ownership, the framework proposes to designate either individual service users or a central committee as service owner. A single owner that bears all cost but also appropriates all benefits of a service has several advantages. Single service ownership facilitates performance management for services and encourages owners to look for business opportunities of their internal processes, turning them into marketable services to expand their business case.

In section 7, we discuss changes to the assignment of responsibilities as compared to what traditional IT governance proposes and present the results of an application of our framework in a practical case.

7   Initial Evaluation of Utility and Lessons Learned

As a first step to validate its utility and applicability and to inform its further development, we discussed our framework with enterprise architects and other employees involved in the SOA initiative at Landgate (www.landgate.wa.gov.au). Landgate started to engage in an SOA initiative in 2006. Since then, the organisation has made some crucial organisational changes to realise SOA benefits, most remarkably the introduction of an Enterprise Architecture Office (EAO). The EAO is responsible for an impact review of proposed service projects as well as for recommending changes that leverage business investments. Thus, compared to other organisations with limited experience in SOA, Landgate's approach can be seen as quite advanced in terms of governance. In the context of this work, Landgate was very much interested in contributing to the development of the SOA governance framework and willing to learn

from that process as well. In the following, we show that the practical application of the framework not only led to an evolution of the framework itself but also helped Landgate to identify alternative approaches to assigning responsibilities within their SOA initiative and clarify the decision rights for key stakeholders.

When we discussed our recommendations and their own approach with Landgate, we perceived the limited support for interdepartmental projects due to the lack of authority and involvement of business executives in the SOA as an issue. Landgate is currently limited to a bottom-up approach to service development due to limited resources, and service development is initially based on "what services the organisation needs". Their approach is primarily project-focused, meaning that priority is given to building services as they are required by larger scale projects, which are product development projects in most cases. An impact analysis based on how the services might be shared is only an afterthought and does not affect portfolio decisions. Once a project has been started, the EAO can make suggestions on how the services should be developed. Now, in a planned re-engineering of core systems, a portfolio and program management approach is anticipated. This discussion with Landgate not only motivated the introduction of the different SOA governance layers (see section 4.1) in our framework but also strengthened the emphasis on the proposed top-executive involvement in developing a SOA roadmap and the recommendation of a more top-down oriented approach to leverage the potential of SOA for reuse and flexibility and thus long-term benefits [29].

A pain point perceived by Landgate was their unsatisfactory solution to the concept of service ownership, which differed significantly from the notion we propose in our framework. First, Landgate determines ownership at the project level. This effectively allows for multiple ownership at the project level as Landgate's scope of projects typically comprises several software services that are jointly offered as one product to their customers. Second, the ownership concept is more seen as a technical or operational role at Landgate. As a consequence, project owners usually do not seek to capture the business benefits of development projects. The direct attribution of benefits is further reduced if the EAO requires a proposed service to be changed in order to increase reusability. The EAO can then sponsor the additional cost. This, however, impedes performance measurement problems, since determining the additional cost as well as the reduction of development cost in reuse scenarios can only be calculated based on budgeted cost, as the actual cost is only known in the scenario that has been chosen. Measuring performance based on budgeted cost is not objective and can be subject to manipulation. Thus, we learned from the observed circumstances that a single service sponsor or benefits owner is required before a project is started. If no sponsor can be found, we recommend that the EAO (in Landgate's case) or another centralised committee should have the budget and authorisation to sponsor and "own" the service. If they can sell the service to the internal users and potentially external customers as well, their performance can be measured based on actual and therefore objective financial data.

8   Summary and Outlook

This paper has presented selected parts of a new framework for SOA governance. We focused on what changes to traditional IT governance approaches are required in order

to utilise the business potential of service-orientation. Initial validation at a Western Australian government agency showed that the framework can assist organisations in evaluating their own governance structure and in identifying the main obstacles to financial returns on their SOA investments. By comparing their own organisational governance model to the roles, activities and their alignment as proposed by our framework, organisations can identify divergences, which might point to weaknesses in their own approach. Once obstacles have been identified, however, major changes within the organisational structure as well as a change in mindset are often required. Therefore, it has to be borne in mind that opposition from within the organisation is likely to arise and that the implementation of required changes might take a considerable amount of time, potentially necessitating the involvement of external consultants with experience in the fields of SOA governance and change management. The proposed framework should be seen as a starting point for the research community and, at this stage, stays below the level of elaboration of its archetypes Cobit and ITIL. Its current limitations include the preliminary empirical evidence in Australia only at this stage, the emphasis on organisational aspects of SOA governance at the expense of other governance aspects such as policies, processes and metrics, and its yet untested economic efficiency. To arrive at a fully-fledged reference model for SOA governance, further work is required to evaluate the framework in real world organisations and to inform its refinement. This will be part of our future work. Moreover, we have started to derive and define a SOA governance meta-model, which will provide a formalised foundation for the framework presented in this paper and will facilitate comparison with related frameworks, the implementation of tool support and the customisation of the framework for organisation-specific needs. In addition to that, we see research opportunities in broadening the scope by integrating the different players of a service ecosystem, such as service brokers, service consumers and service providers, into the model and examine who will have the market power to set standards and force other players to comply with them in an ecosystem environment.

References

1. IT Governance Institute (ITGI). Cobit 4.1. Isaca, Rolling Meadows (2007)
2. Ridley, G., Young, J., Carroll, P.: COBIT and its Utilization: A Framework from the Literature. In: Proc. of the 37th Hawaii Int. Conf. on System Sciences (HICSS 2004), pp. 1–8. IEEE, Los Alamitos (2004)
3. Weill, P., Ross, J.W.: IT Governance: How Top Performers Manage IT Decision Rights for Superior Results. Harvard Business School Press, Boston (2004)
4. Bernhardt, J., Seese, D.: A Conceptual Framework for the Governance of Service-Oriented Architectures. In: Proc. of the 3rd Workshop on Trends in Enterprise Architecture Research (TEAR 2008), Sydney, Australia, December 1 (2008)
5. Erl, T.: SOA Principles of Service Design. Prentice Hall, Upper Saddle River (2008)
6. Barnes, M., Sholler, D., Malinverno, P.: Benefits and Challenges of SOA in Business Terms. Gartner Research, 1–6 (2005)
7. Barnes, M., Malinverno, P.: Learn the Key Success Factors for SOA Deployments. Gartner Research 8 (2005)
8. Barros, A., Dumas, M.: The Rise of Web Service Ecosystems. IT Professional 8(5), 31–37 (2006)

9. Schulz-Hofen, J.: Web Service Middleware - An Infrastructure For Near Future Real Life Web Service Eco-systems. In: Proc. of the IEEE Int. Conf. on Service-Oriented Computing and Applications (SOCA 2007), Newport Beach, California, pp. 261–270. IEEE, Los Alamitos (2007)
10. Malinverno, P.: Service-Oriented Architecture Craves Governance. Gartner Research, 1–6 (January 20, 2006)
11. Malinverno, P.: Gartner Research Index on SOA Governance. Gartner Research, 1–5 (August 16, 2006a)
12. Webmethods. SOA Governance - Enabling Sustainable Success with SOA (October 2006), http://www1.webmethods.com/PDF/whitepapers/SOA_Governance.pdf
13. Holley, K., Palistrant, J., Graham, S.: Effective SOA Governance (2006), ftp://ftp.software.ibm.com/software/rational/web/whitepapers/soagov-mgmt.pdf
14. Schelp, J., Stutz, M.: SOA-Governance. HMD-Praxis der Wirtschaftsinformatik 253, 66–73 (2007)
15. Bell, M.: Service-Oriented Modeling: Service Analysis, Design, and Architecture. Wiley, Hoboken (2008)
16. Shah, R., Bieberstein, N., Bose, S., Fiammante, M., Jones, K.: SOA Project Planning Aspects (2006), http://websphere.sys-con.com/read/168398.htm
17. Dodani, M.H.: Who Took the Cookie from the Cookie Jar? Journal of Object Technology 5(4) (May-June 2006), http://www.jot.fm/issues/issue_2006_05/column3/
18. ITIL. IT Infrastructure Library v3. OGC, TSO (2007), http://www.itil-officialsite.com/home/home.asp
19. OECD. OECD Principles of Corporate Governance. Org. for Economic Co-Operation and Development (2004), http://www.oecd.org/dataoecd/32/18/31557724.pdf
20. Malinverno, P.: Sample Governance Mechanisms for a Service-Oriented Architecture. In: Gartner Research, April 27, pp. 1–5 (2006b)
21. Software AG. Best Practices for SOA Governance – User Survey. SOA Governance Survey (2008), http://www.infoq.com/zones/centrasite/res/download/SOAGovernanceBestPracticesSurvey.pdf
22. Biske, T.: SOA Governance. Packt Publishing (2008)
23. Marks, E.A.: Service-Oriented Architecture (SOA) Governance for the Services Driven Enterprise. Wiley, Chichester (2008)
24. Brown, W.A., Laird, R.G., Gee, C., Mitra, T.: SOA Governance - Achieving and Sustaining Business and IT Agility. IBM Press (2008)
25. Kreger, H., Estefan, J.: Navigating the SOA Open Standards Landscape Around Architecture, White Paper W096, The Open Group (July 2009), http://www.opengroup.org/pubs/catalog/w096.htm
26. OMG. OMG SOA Governance Metamodel and Profile (SGMP) Standard Wiki (2009), http://www.omgwiki.org/soagov/doku.php
27. OASIS. OASIS Reference Architecture for Service-Oriented Architecture, Version 1.0, OASIS Public Review Draft 1 (April 23, 2008), http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf
28. The Open Group, The Open Group SOA Governance Framework, Draft Technical Standard (2009), http://www.opengroup.org/projects/soa-governance

29. MacKenzie, C.M., Laskey, K., McCabe, F., Brown, P.F., Metz, R.: OASIS Reference Model for Service Oriented Architecture 1.0 (October 2006),
    http://docs.oasis-open.org/soa-rm/v1.0/
30. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Quartely 28(1), 75–105 (2004)
31. Morris, P.W.G., Jamieson, A.: Moving from Corporate Strategy to Project Strategy. Project Management Journal 36(4), 5–18 (2005)
32. Martinelli, R., Waddell, J.: Program Management: It's About the Business. PM World Today 4(1), 1–6 (2007)
33. Archer, N.P., Ghasemzadeh, F.: Project Portfolio Selection and Management. In: Morris, P.W.G., Pinto, J.K. (eds.) The Wiley Guide to Management Projects. John Wiley & Sons, NY (2004)
34. Cooke-Davies, T.: Project Success. In: Morris, P.W.G., Pinto, J.K. (eds.) The Wiley Guide to Management Projects. John Wiley & Sons, New York (2004)
35. Riedl, C., Boehmann, T., Rosemann, M., Krcmar, H.: Quality Management in Service Ecosystems. Information Systems & E-Business Management, 1–23 (2008)
36. Deb, M., Helbig, J., Kroll, M., Scherdin, A.: Bringing SOA to Life: The Art and Science of Service Discovery and Design. SOA Web Services Journal 27, 42–47 (2005)
37. Zhao, L.J., Goul, M., Purao, S., Vitharana, P., Wang, H.J.: Impact of Service-Centric Computing on Business and Education. Comm. of the Association for Information Systems 22, 295–310 (2008)
38. Kohlborn, T.: A Consolidated Approach for Service Analysis. Master's Thesis, BPM Group, Faculty of IT, Queensland University of Technology, Brisbane (2008)
39. Schepers, T.G.J., Iacob, M.E., Van Eck, P.A.T.: A Lifecycle Approach to SOA Governance. In: Avanzi, R.M., Keliher, L., Sica, F. (eds.) SAC 2008. LNCS, vol. 5381, pp. 1055–1061. Springer, Heidelberg (2009)
40. Ren, M., Lyytinen, K.: Building Enterprise Architecture Agility and Sustenance with SOA. Comm. of the Association for Information Systems 22, 75–86 (2008)
41. Gerlach, J., Neumann, B., Moldauer, E., Aro, M., Frisby, D.: Determining the Cost of IT Services. Comm. of the ACM 45(9), 61–67 (2002)
42. Ross, J.W., Vitale, M.R., Beath, C.M.: The Untapped Potential of IT Chargeback. MIS Quarterly 23(2), 215–237 (1999)
43. Kajko-Mattsson, M., Lewis, G.A., Smith, D.B.: A Framework for Roles for Development, Evolution and Maintenance of SOA-Based Systems. In: Proc. of the Int. Workshop on Systems Development in SOA Environments (SDSOA 2007), pp. 7–12 (2007)
44. Bieberstein, N.: Service-Oriented Architecture Compass: Business Value, Planning, and Enterprise Road-map. Prentice Hall, Englewood Cliffs (2006)
45. Papazoglou, M.P.: Web Services: Principles and Technology. Prentice Hall, Harlow (2008)
46. Matsumura, M.: Delivering Business Agility with SOA: View from the Corporate Trenches. In: The Online Conference Series on Keys to SOA, pp. 21–30 (2008)
47. Strano, C., Rehmani, Q.: The Role of the Enterprise Architect. Information Systems and E-Business Management 5, 379–396 (2007)
48. Haines, M.N.: The Impact of Service-Oriented Application Development on Software Development Methodology. In: Proc. of the 40th Hawaii Int. Conf. on System Sciences (HICSS 2007), pp. 172–180. IEEE, Los Alamitos (2007)