Software Engineering Institute

A Scenario-Based Technique for Developing SOA Technical Governance

Soumya Simanta
Ed Morris
Grace A. Lewis
Sriram Balasubramaniam
Dennis B. Smith

June 2009

TECHNICAL NOTE
CMU/SEI-2009-TN-009

Research, Technology, and System Solutions Program

http://www.sei.cmu.edu

Carnegie Mellon

This report was prepared for the

SEI Administrative Agent
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about purchasing paper copies of SEI reports, please visit the publications section of our website (http://www.sei.cmu.edu/publications/).

Table of Contents

List of Figures

List of Tables

Abstract

A well-known problem within the service-oriented architecture (SOA) community is the need to establish effective SOA governance procedures to enable an organization-wide SOA initiative. A number of organizations and vendors address this problem through SOA governance frameworks that provide models, procedures, and tools for SOA governance. Many of these SOA frameworks are general purpose because they are intended to be useful for a diverse customer base. However, while designed for a wide customer base, vendor SOA frameworks tend to be narrowly focused to work with the specific tools of the vendor. A critical problem for an organization when implementing SOA governance is to customize vendors' offerings to its specific technological and management context. In this technical note, a lightweight and extensible technique is proposed, one that employs scenarios to tailor existing SOA governance frameworks to the specific needs of an organization.

1 Introduction

Service-oriented architecture (SOA) is a way of designing, developing, deploying, and managing systems characterized by coarse-grained services that represent reusable business functionality. Service consumers compose applications or systems using the functionality provided by these services through standard interfaces. At a high level, SOA is a way to design, develop, deploy, and manage systems in which

- Services provide reusable business functionality.
- Service consumers are built using functionality from available services. Examples of services consumers are end-user applications, portals, internal systems, external systems, and composite services.
- Service interface definitions are first-class artifacts.
- An SOA infrastructure enables discovery, composition, and invocation of services.
- Protocols are predominantly, but not exclusively, message-based document exchanges.

From a more technical point of view, SOA is an architectural style or design paradigm—it is neither a system architecture nor a complete system. As an architectural style, it is characterized by a set of components and connectors, situations when the style is applicable, and benefits associated with implementing the style.

If implemented correctly, SOA adoption can provide business agility, reuse of business functionality, and leverage of legacy systems for an organization. Many organizations recognize these potential benefits and are adopting SOA—some more successfully than others. SOA has indeed "crossed the chasm,"[1] according to a recent Software AG user survey in which 90% of the respondents claim to have made some commitment to SOA adoption [17].

But there are concerns, chief of which appears to be SOA governance. The Software AG user survey also reveals that "an overwhelming percentage of respondents (over 90%) view governance as significant with 54% calling it critical" [17]. Further, Gartner has identified the lack of governance as the most common reason for failure of SOA projects [1], and an *InfoWorld* 2007 SOA Trend Survey labels the lack of governance is the main inhibitor for SOA adoption (50%) [2]. An ebizQ survey on SOA governance funded by Oracle covering 118 companies, reveals that most organizations believe that SOA governance is a critical part of their SOA strategy, with 49% believing that without governance their SOA plans will fail [18].

There are several definitions of SOA governance. For example, IBM defines SOA governance as the process of establishing the chain of responsibilities and communications, policies, measurements, and control mechanisms that allow people to carry out their responsibilities in SOA projects [3]. eBizQ states that SOA governance provides organizations with the processes, policies, and solutions/technologies that can help to manage increasingly complex SOA deployments in an effective and efficient manner [18].

[1] This term was coined by Geoffrey Moore in his book *Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers* to refer to the chasm that exists between visionaries (early adopters) and pragmatists (early majority), from a technology adoption perspective.

In general, SOA governance provides a set of policies, rules, and enforcement mechanisms for developing, using, and evolving service-oriented systems, and for analysis of the business value of those systems. SOA governance includes policies, procedures, roles, and responsibilities for design-time governance and runtime governance. Design-time governance includes elements such as rules for strategic identification, development, and deployment of services; reuse; and legacy system migration. It also enforces consistency in use of standards, SOA infrastructure, and processes. Runtime governance develops and enforces rules to ensure that services are executed only in ways that are *legal* and that important runtime data is logged. From a life-cycle point of view, design-time governance applies to early activities such as planning, architecture, design, and development. Runtime governance applies to deployment and management of service-oriented systems.

Often, organizations begin their exposure to SOA with small pilot projects that provide the environment to experience and learn about various aspects of service orientation. However, these pilot projects are often pragmatically limited to experimentation with technical feasibility and do not address SOA governance. Even when SOA governance is addressed, the results are unlikely to scale to organization-wide SOA efforts, leaving many critical SOA governance questions unanswered, such as:

- What services should be implemented?
- Does a proposed service represent a new, reusable capability?
- How are service changes and upgrades decided and communicated?
- What are expectations for service verification and validation?
- Who pays for maintenance and development of services?
- Who owns a service and the data it uses?

There are multiple SOA governance frameworks that provide basic governance concepts to support SOA adopters. Some of these are offered by

- IT market leaders such as IBM and Oracle [4], [5]
- niche SOA vendors and consultants such as SOA Software, Software AG and AgilePath [6], [7], [8]
- independent consulting companies such as CBDi [9]
- industry organizations, even if they are not SOA specific, such as the Information Technology Infrastructure Library (ITIL) [10]

These frameworks can be very useful; they define specific vocabulary and identify the basic capabilities of SOA governance. Some even suggest architectures and identify best practices, particularly for use of vendor-specific tools. They were developed based on experiences of actual organizations, and as such represent abstracted and generalized critical features of SOA governance. The frequent failures of SOA governance are not due to these frameworks, but to the inability of organizations to relate and adapt them to their specific contexts. While the frameworks suggest in general what is involved in SOA governance, they do not tell organizations specifically what to do and how to do it.

This report proposes a scenario-based technique for tailoring existing SOA governance frameworks in order to establish a customized SOA governance strategy that addresses the needs of a

specific organization adopting SOA. Although the technique could be applied in a multi-organizational context, and some elements of the established strategy could involve relationships with other organizations, the focus of this report is the implementation within a single organization. Section 2 presents concepts related to SOA governance and analyzes several existing SOA governance frameworks to identify common elements in these frameworks. Section 3 presents a scenario-based technique for identifying organization-specific SOA governance needs and for tailoring SOA governance frameworks to address these needs. Section 4 summarizes and concludes the report.

2 Existing Governance Frameworks

Existing SOA governance frameworks are diverse, employ unique vocabularies, and possess different strengths and limitations. However, there are many common elements across these frameworks that are often obscured by their unique vocabularies and marketing jargon. This section presents an abstract architecture of existing SOA governance frameworks that removes vendor- and implementation-specific details and summarizes the core concepts using a simple, generic vocabulary. The SOA governance frameworks listed in Section 1 were considered for this analysis because of their widespread popularity in the SOA community.

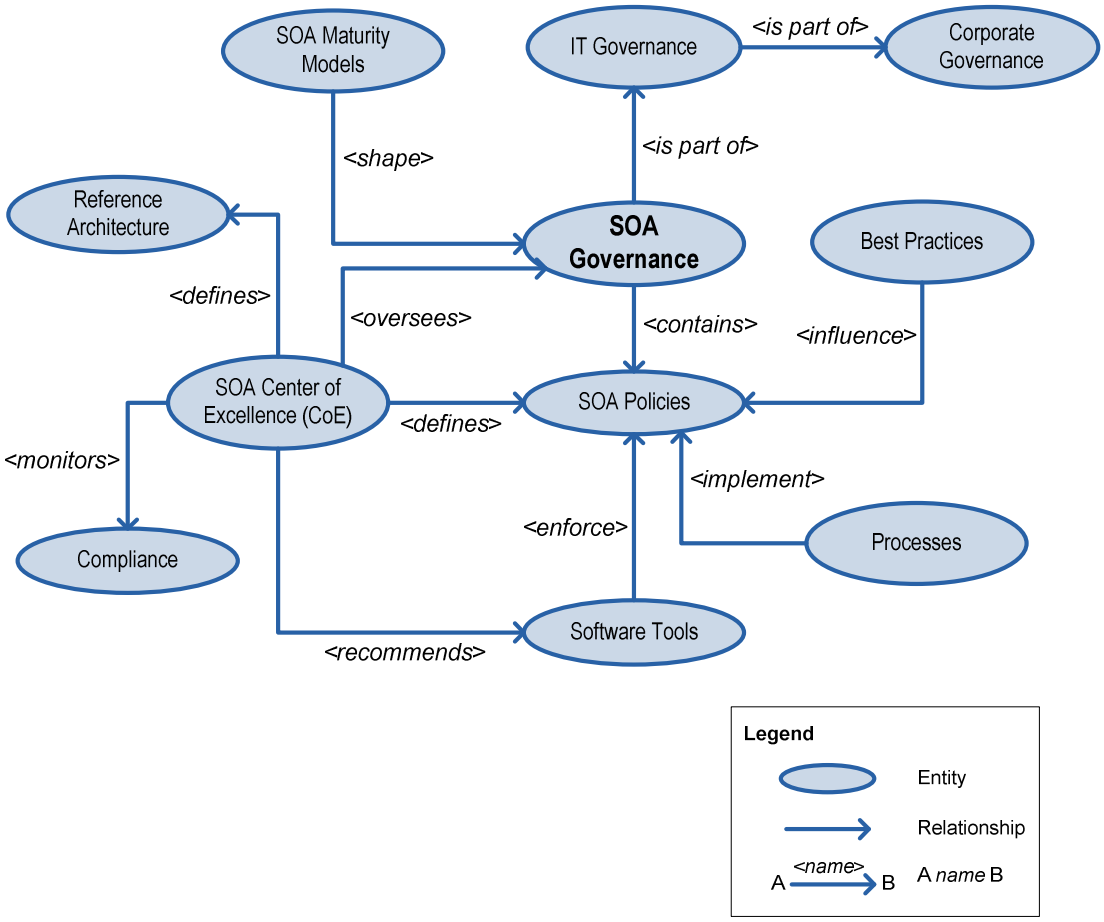

Figure 1: Entity-Relationship Diagram of SOA Governance Framework Elements

Figure 1 shows and abstraction of the following common elements in these SOA governance frameworks, expressed as a simple entity-relationship diagram.

- SOA Governance
- IT Governance
- Corporate Governance
- Best Practices

- Processes
- Software Tools
- SOA Policies
- Compliance
- SOA Center of Excellence (CoE)
- Reference Architecture
- Maturity Models

Sections 2.1 through 2.5 discuss the entities presented in Figure 1, as well as some important relationships among these entities.

2.1 SOA Governance is Part of Corporate and IT Governance

All of the governance frameworks studied for this report assume that SOA governance is part of a larger corporate governance or IT governance framework. For example, not allowing the installation of certain software products on office machines due to security concerns is an example of IT governance that is not related to SOA governance. This implies that any implementation of a new SOA governance framework must be done in the context of corporate and IT governance models and practices and must fit the organization's unique culture. Organizations adopting an SOA governance framework should be clear about how it supports broader business and IT goals. Some frameworks strongly suggest creating an SOA CoE to oversee the strategy and implementation of SOA governance mechanisms organization-wide. The scenario-based technique proposed in this report could be orchestrated by an SOA CoE.

2.2 SOA Governance is Incremental and Based on Maturity Models

Across the frameworks we considered, there is agreement that an organization should implement SOA governance incrementally. An incremental approach involves (1) analyzing the organization's existing IT and SOA governance structures (e.g., policies, processes, tools, organizational units) and the future governance required for SOA and (2) determining a step-wise approach for implementing SOA governance or improving and enhancing if it already exists. An incremental approach will allow an organization to adopt best practices gradually—and learn from early mistakes to improve subsequent practices. In addition, an organization can put in place SOA governance strategies that support early phases of migration to SOA. For example, an organization can determine that the first step in migrating will involve implementing services that are exactly analogous to existing, non-SOA capabilities. SOA governance practices can verify the analogous nature of the new services and support their deployment. Applications can then code calls to these services in place of existing calls (or whatever mechanism was used). At this stage, it may be appropriate to put in place practices to ensure that services are invoked and used appropriately by applications.

A key feature of SOA governance frameworks is the incorporation of a maturity model that evaluates the sophistication of the policies and practices in place within an organization. The rationale for maturity models is that as organizations become more mature with respect to SOA adoption, governance aligned with the increased maturity is warranted. Some form of maturity model is present in all the considered frameworks, although the actual models are quite different. IBM has

perhaps the most comprehensive model [11]. It is based on the type and maturity of integration (data integration, application integration, functional integration, process integration, supply-chain integration, virtual infrastructure, and eco-system integration) inside an organization. The CBDI maturity model is focused on the maturity of policies, processes, organization, and infrastructure [12].

2.3 Policies, Processes, and Best Practices

An SOA policy, as shown in Figure 1, is a directive or strategy that is defined independently of how it will be carried out. For example, a policy might require that all services undergo certification before they can be used. Policies are implemented through processes and ideally should be influenced by best practices.

Policies are important elements of all SOA governance frameworks, but policies differ widely in terms of their importance, scope, specificity, and potential implementations. Some policies such as those related to service usage and quality of service monitoring can be formally specified and enforced automatically. Others require the commitment and discipline of people to adhere to specific processes that support the policy in order to achieve desired outcomes and compliance.

Not all policies are equally constraining—some are merely guidelines. For example, guidelines are often created for identifying preferred granularity of services. However, a particular service may be an exception to the guideline due to specific quality requirements (e.g., performance). Other policies are more stringent and can result in critical consequences when violated. For example, failure to comply with a policy requiring registration of new services results in duplication of services across the organization. This violation is counterproductive to the goal of organization-wide reuse.

2.4 Policy Compliance and Enforcement Mechanisms

All of the frameworks we considered include an emphasis on compliance and enforcement. In the case of vendor-created SOA frameworks, compliance and enforcement strategies are consistent with and supported by the SOA products they market. For example, vendors with strong repository/registry products advocate runtime monitoring and enforcement of service usage that can be accomplished by their products, while those vendors with strong modeling and design support prefer to emphasize governance controlling the design of services.

There are numerous commercial software tools that implement aspects of SOA governance by supporting specific policies and monitoring compliance. While tools supporting compliance and monitoring for the majority of the service life cycle are available, no single vendor provides a full suite of tools or an integrated solution. One of the biggest challenges is creating a “best-of-breed” solution by integrating various products [13]. Unfortunately, there are no SOA-specific governance standards [14]. There are also no standards that allow governance tools to be easily tied together into an integrated solution. Thus, building a best-of-breed solution for SOA governance and compliance will remain a difficult task, which might be eased if vendors adopted standards. On the other hand, standardizing all aspects of SOA governance may be unrealistic and may not yield all the benefits associated with standardization. First, given that SOA governance is part of IT governance and each organization has specific SOA needs, standardizing all aspects of SOA governance will require all organizations to follow the same IT governance approach, which is

unlikely and unrealistic. Secondly, most aspects of SOA governance are limited to a single organization, which means that standardization in the context of SOA governance would be more beneficial where inter-organization interoperation is important, which is not always the case.

2.5 SOA Center of Excellence

Many frameworks strongly suggest the creation of SOA CoE that not only promotes adoption of SOA best practices organization-wide but can also

- evaluate and recommend SOA governance tools and products best suited for the organization
- keep track of the latest technologies and vendor products
- help various units inside the organization adopt existing governance policies and create new ones, if required.
- monitor the compliance to SOA governance policies across the organization, as well as provide the business with a realistic analysis of the maturity of the service-oriented initiatives.

Although creating a CoE is widely recommended, it is not necessary for establishing SOA governance.

3 Developing SOA Governance Using Scenarios

Many organizations considering SOA adoption are at least somewhat aware of the governance frameworks of various vendors and industry groups. However, the SOA governance needs of these organizations vary across a wide spectrum. On one end are organizations where extreme agility in producing new applications and services is key (either directly by the organization or by other interested parties). On the other end are SOA implementations in the U.S. Department of Defense (DoD) that involve military command and control and real-time surveillance where there is a high need for control.

Clearly, the same SOA governance solution or framework is not appropriate for everyone. Some differentiators are cultural—technical sophistication, application domain, process disciplines, business/mission goals, and market profile. In most cases, it is neither desirable nor possible for organizations to directly adopt a solution from a single vendor. In some cases, organizations may find the vendor governance frameworks overwhelming. Also, organizations may find that their unique SOA governance needs and characteristics of existing infrastructure, applications, and IT governance approaches impose constraints that preclude single vendor strategies. These organizations may consider building their own SOA governance implementation.

Thus, while organizations may be aware of the available governance frameworks, they also might be confused about what SOA governance can do for *them*, what SOA governance actually entails for *them*, which specific governance capabilities *they* should implement, and what strategies for adopting those capabilities *they* should employ. They probably understand the need for SOA governance, but they do not understand how to implement SOA governance within their specific context.

We believe there is a need for an impartial technique that distills and simplifies the key elements of the existing commercial models, and allows organizations to quickly understand what they need to do to build appropriate SOA governance. The approach that we propose supports organizations in understanding their SOA governance needs and the context in which they need to implement SOA governance. This approach focuses only on the technical aspects of SOA governance. For example, human resource allocation or funding issues related to an SOA project are outside the scope of this approach.

In addition, many organizations play multiple roles or address multiple perspectives in their SOA implementations. They build services (service provider), develop applications that consume services (service consumer), and also build their SOA infrastructure (infrastructure provider), often because of the unique demands of the SOA environment they are establishing. Other organizations are playing only one or two of these roles, as in the example of organizations that are developing or using third-party services. Effective SOA governance for these organizations has to consider each of the relevant perspectives operative in a given context. The scenario-based technique proposed here takes into account these perspectives without going into specific roles.[2] Each sce-

[2] The SOA governance frameworks analyzed go into very specific details about roles and responsibilities (e.g., SOA Architect, SOA Designer) but do not consider the distinction between the perspectives of service consumer, service provider, and infrastructure provider that we believe are important.

nario provides the context and requirements for a specific situation related to technical aspects in an SOA project. The elements of a scenario are defined in Section 3.5.

Characteristics of the technique include

- vendor-neutral and applicable regardless of which vendor-provided or custom SOA governance framework is adopted
- scenario-based to capture organization-specific governance concerns
- risk-aware to support organizations in the analysis and remediation of potential problems in governance

The technique is not intended to replace existing commercial offerings, but to provide a starting point to help organizations understand their specific SOA governance needs and navigate the available offerings. Thus, the technique can be applied in conjunction with the governance frameworks we have considered or any other existing frameworks.

3.1 Overview of the Technique

The technique employs six activities to understand the organization's context and to create scenarios relating to an existing SOA governance framework. These activities include

1. Establish context—SOA governance drivers, framework, and scope.
2. Develop classification schemes.
3. Create affinity groups by SOA governance needs.
4. Create and refine scenarios of SOA governance needs.
5. Consolidate scenarios.
6. Customize policies to fit SOA governance framework.

A visual representation of the technique is presented in Figure 2, in which the direction of the arrows illustrates the flow of activities. This figure shows that the first three activities can be performed concurrently. However, for a single iteration they should be completed before the individual groups can start developing scenarios (even if there is a need to revisit these activities at some point in the process). The remaining activities are sequential. The loop shown in Figure 2 from Activity 6 back to Activity 1 reflects an incremental approach to SOA governance that requires continuous revision of the context, the groupings of SOA governance elements, and the grouping of organizational units.

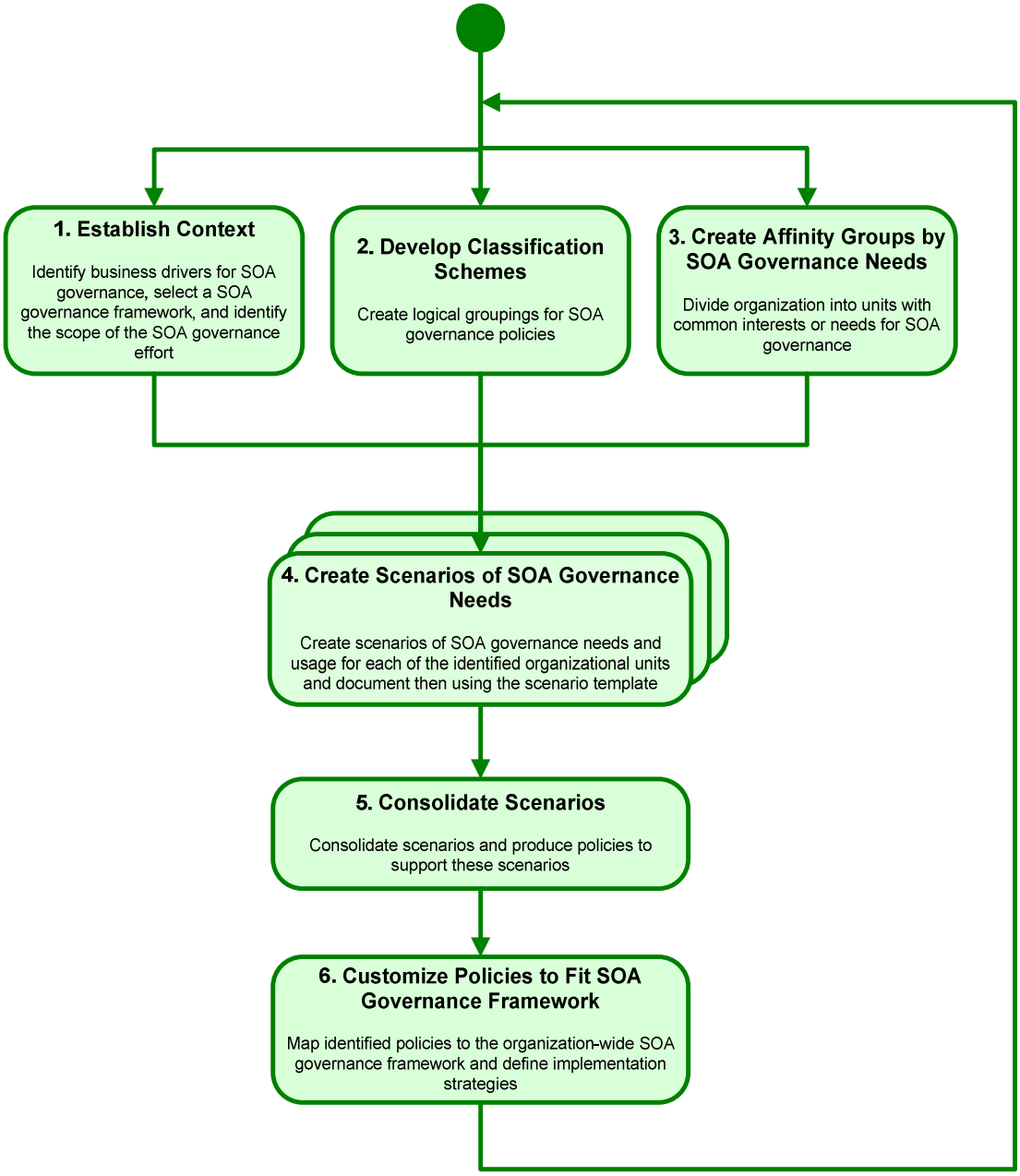

Figure 2: Visual Representation of a Technique for SOA Governance Development

Activities 1 and 2 in Figure 2 (Establish Context and Develop Classification Schemes) are organization-wide actions that should be performed by a central authority (such as the SOA CoE) to foster general agreement. Create Affinity Groups, Activity 3, requires interaction between the central authority and the various lines of business involved (or planning to be involved) in SOA efforts. Activity 4 (Create Scenarios) is repeated for each of the organizational units identified in Activity 3. The final two activities, Consolidate Scenarios and Customize Policies, build on the results of previous ones. Activity 5 consolidates all the scenarios identified in Activity 4 by each group. Activity 6 relates the work of the previous activities to the organization's chosen SOA governance framework. Each activity in the technique is described in a subsequent section.

3.2 Activity 1: Establish Context

The goal of establishing context is to collect and record information that will guide the scenario and policy generation activities. One part of establishing context is determining the business drivers or justification for SOA governance. For example, the organization may be driven to ensure that all users of a business service access the same capability—that there is consistent and widespread use of the same service within the organization. Or, the organization may be driven to ensure that no "rogue" versions of services can be used and that all services are completely vetted for security, reliability, and other qualities (i.e., certified) prior to deployment. If these drivers have not previously been made explicit, they must be made so at this point to inform all participants in the scenario generation process about the expectations of the organization.

Another part of establishing context is identifying the scope of the SOA governance effort. If it is impossible to develop a consistent set of drivers for all participants (i.e., if various participants are expected to respond to different goals), the context of the scenario-based activity should be scaled to assure that all participants are responding to consistent drivers. One result of scaling the context may be that multiple implementations of the scenario-based technique are needed for groups within the organization that must address disjointed SOA governance drivers.

The final part of establishing context is selecting an SOA governance framework that addresses the identified drivers. This SOA governance framework can be an existing framework from a commercial vendor, one based on a standard or a widely recommended approach such as ITIL, one custom-built for the organization, or a hybrid of all of the preceding. For organizations that have IT governance in place, the most logical step is to select an SOA governance framework that is consistent with existing IT governance.

3.3 Activity 2: Develop Classification Schemes

Classification schemes are used to categorize scenarios and the policies designed to address problems they raise. The schemes can often be simply based on the selected SOA governance framework. Common classification schemes group scenarios according to goals (e.g., service certification), life-cycle phase (e.g., service design time), focus of activity (e.g., financial, technology), or usage (e.g., externally visible services, internally visible services). For example, CBDi provides a model of the service life-cycle phases (feasibility, approval, design, build/integrate, and launch) that can serve as a classification scheme [9]. Oracle provides a framework for policies related to people, financial, portfolio, operation, architecture, information, technology, and project execution as shown in Figure 3 [5]. This framework can be the impetus for an alternate classification scheme.

Classification schemes also provide a shorthand for discussing groups of governance scenarios and policies and support efficient communication among participants. In addition, to ensure broad coverage of governance concerns in the scenarios and simplify the consolidation of the efforts of multiple participants, classification schemes should be developed as input to scenario generation.[3]

[3] It may be the case, especially in organizations new to governance, that classification schemes identified as scenarios are generated in the next step. If this is the case, the classification schemes need to be updated for all other groups so that everyone uses the same classification—an aid also for the consolidation step.

It is often useful to employ multiple classification schemes. For example, an organization may find value in applying both a life cycle phase scheme like that of CBDi to identify the major stages a service will go through, as well as an activity-focused scheme like that of Oracle to ensure that all aspects of governance are covered in each life-cycle phase.

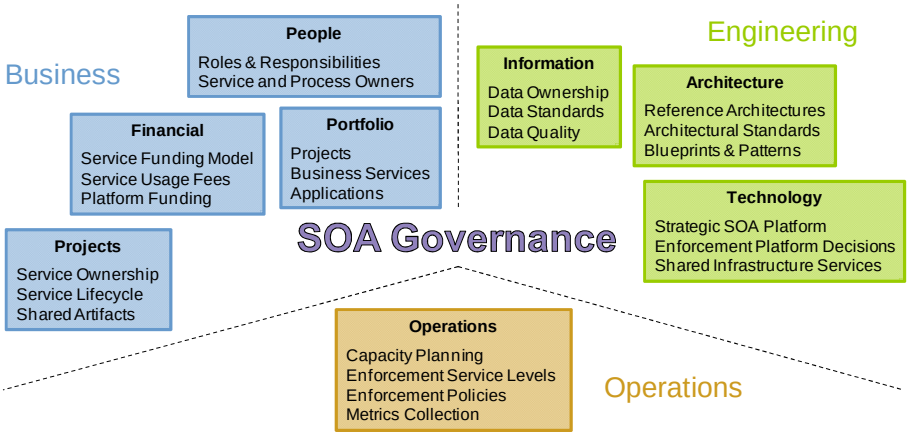

Figure 3: SOA Governance Elements in Oracle's SOA Governance Framework

3.4 Activity 3: Create Affinity Groups by SOA Governance Needs

The first two activities are best accomplished by employing a top-down approach under control of a central authority such as a CoE. However, central authority insight is limited by the differing SOA governance needs within the organization. Indeed, it is difficult (and some would argue impossible) for a central authority to identify the full range and scope of governance issues within the organization or to understand the ramifications of the selected SOA governance frameworks on all parts of the organization.

These different needs are driven by varying views of the governance problem and different ways that various groups must respond to solve problems. Groups with similar and related needs (affinity groups) are created and tasked with creating scenarios to capture these different viewpoints. One common and logical approach, for example, is to group according to lines of business.

We recommend two guidelines for affinity group membership. First, the grouping process should lead to group sizes that are manageable and allow the productive generation of scenarios.[4] Second, each group should incorporate representatives of the three perspectives of service provider, infrastructure provider, and service consumer, if they all exist in the given organization whose governance scope is under consideration.

3.5 Activity 4: Create and Refine Scenarios of SOA Governance Needs

Scenarios are generated independently by each of the affinity groups established in Activity 3. Generating appropriate scenarios that provide broad coverage of SOA governance will likely require multiple brainstorming sessions. Each session is typically held as a workshop to encourage insightful thinking about the real problems that the group will face in SOA governance. Similar

[4] Literature on brainstorming and team exercises claims that ideal group size is 3-5 people, with a maximum of 12-15 people, plus a facilitator.

scenario-based approaches have been successfully applied in many engineering contexts, including requirements analysis and architecture tradeoff analysis [15], [16].

The initial round of brainstorming is intended to elicit broad coverage of governance problems, a goal supported by requiring groups to consider a set of scenarios that address as many categories within the classification scheme as possible. To accomplish this, the facilitator goes through each category in the classification scheme and asks the group to think about situations that could lead to problems which could be alleviated if there were some kind of policy or control in place. For example, using Oracle's classification scheme, a concern under *Operations* could be access to unauthorized data, a concern under *Technology* could be inconsistent use of a particular technology by service developers, and a concern under *People* could be the definition of the role responsible for registry maintenance.

Following this initial round, the groups may need to reconsider the list of scenarios by validating them against the drivers and scope of SOA governance. In an extreme case, it might be necessary to revisit the first activity in the process because the scenario(s) may not be achievable in the context. At this point, the group is asked to remove, merge, add, or reword scenarios, if necessary. Each final scenario is then documented using the template provided in Table 1.

Table 1: SOA Governance Scenario Template

Scenario Element	Description
Concern	Very informal description of a concern with respect to SOA adoption
Scenario Description	Concise description of a situation, the context in which the situation occurs, and the participants in the situation that would address the above concern
Governance Drivers	The specific SOA governance drivers (defined in Section 3.2) that must be addressed in any policies that are created to address the scenario.
Scenario Categories	Each service should be related to at least one category that is part of the classification scheme defined previously. For example, if the classification scheme is based on service life-cycle phases, then a scenario involving avoiding redundancy in services may be categorized as *planning phase governance*. A single scenario may be categorized into more than one category (e.g., design-time and runtime) or there may be multiple classification schemes (e.g., service life cycle and business line).
Perspectives	We identify three primary perspectives involved in SOA governance: service provider, service consumer, and infrastructure provider. Separating these three perspectives offers a valuable abstraction for understanding governance issues. Each perspective will likely map to multiple roles within the organization.
Policies	Policies are a starting point for implementing governance. A scenario may map to multiple policies. For example, a scenario involving release of new versions of a service may map to policies for service providers, policies for service consumers, and policies for infrastructure providers, because each must respond in predefined ways during a release.
Implementation Mechanisms	Implementation mechanisms are suggested approaches to implementing policies. For example, implementing a policy for a trusted registry may involve mechanisms such as digital signatures for interfaces and the restriction of access to the registry to select, trusted parties. There may be multiple mechanisms required to implement a single policy.
Risks and Mitigations	Multiple risks are potentially associated with each policy and mechanism, and multiple mitigations may be associated with risks. Mitigations may be assigned to one or more perspectives and may ultimately lead to additional tasks or policies.

Scenario Element	Description
Implications	Implications relate the work of the individual groups back to SOA governance within the larger organization. Implications of policies, implementation mechanisms, and associated risks may potentially affect the SOA governance framework, the CoE, best practices, tool support, or any other aspect of governance.
Exceptions	Exceptions are situations where proposed policies are inappropriate or where alternate policies are needed. Exceptions are rarely considered when developing SOA governance strategy, but almost all attempts to implement SOA governance will be faced with exceptions.

Table 2 maps each element of a scenario to the activity in which it is identified, created, or updated. (Note that no scenario element is identified, created, or updated in the *create affinity groups* activity.)

Table 2: Scenario Elements Mapped to Activities

Activity / Scenario Element	Establish context	Develop classification schemes	Create affinity groups	Create scenarios	Consolidate scenarios and create policies	Customize policies
Concern				Identify		
Scenario Description				Create		
Governance Drivers	Identify					
Scenario Categories	Create	Update				
Perspectives	Identify	Update				
Policies					Create	Update
Implementation Mechanisms				Suggest	Suggest	Suggest
Risks and Mitigations				Identify	Update	Update
Implications				Identify	Identify	Identify
Exceptions				Identify	Identify	Identify

A scenario defined using the template is shown in Table 3. The scenario is on avoiding services with little or no potential for use.

Table 3: Scenario Example Using Template

Scenario Element	Description
Concern	Departments within the organization start to deploy services and there is no control or centralized knowledge of deployed services.
Scenario Description	A service provider that tries to expose a service that corresponds to capability with limited or no use is prevented from doing so.
Governance Drivers	Organization-wide reuse of services to reduce development costs.
Scenario Category	Service Planning Phase, Service Deployment Phase
Perspectives	Service provider, infrastructure provider (infrastructure provider may be involved if part of the verification process occurs inside the infrastructure).
Policies	• Before starting development, any service provider needs to prove the business need for the service and that the service is not redundant with

Scenario Element	Description
	existing services. • All available services have to be published in the centrally accessible service repository.
Implementation Mechanisms	• Service providers, before starting service development, must fill out a form in which they describe the capability provided, related services already available in the registry, and business processes that will use the service. This form is validated by business process owners. • Services providers must check in their services to a centrally accessible registry.
Risks and Mitigations	**Risk**: Service providers and service consumers override the central registry. **Mitigation**: Create an audit process that periodically monitors the service usage and checks for non-registered services. **Risk**: The service metadata does not provide enough information about the service to make a decision on redundancy. **Mitigation**: Service providers can provide examples of service usage and the business process where the service is being used. The service consumers should also update the metadata based on feedback from service consumers.
Implications	All groups and the SOA CoE have to agree on a tool (e.g., registry) and the processes that should be suggested in the implementation mechanisms.
Exceptions	None

3.6 Activity 5: Consolidate Scenarios

The goal of consolidation is to reconcile and merge the work of the various groups in order to identify SOA governance policies for the organization. This process is carried out by the central authority responsible for overall SOA governance (e.g., the SOA CoE) in conjunction with representatives from the various affinity groups. Several cases may arise during consolidation of scenarios generated by different groups:

1. **A scenario is consistently included by multiple groups** (i.e., multiple groups recognize a similar problem and address it in similar ways). In this case, the goal is to merge the scenarios and establish a unified policy that addresses the concerns of most of the groups.
2. **A scenario is in conflict with another scenario**. In this case, various groups have different understandings or prioritizations of SOA governance drivers. It is important to understand whether the conflict is at the level of implementation or the business goal. Conflicts affecting the implementation can be resolved by recommending a policy and an appropriate implementation mechanism. However, if the business goals related to each policy are in conflict, the scenarios should be treated separately.
3. **A scenario is unique**. In this case, the need is to ensure that the scenario is appropriately addressed by SOA governance policy.
4. **Several scenarios across groups appear to be similar, although they are actually different**. In this case, the scenarios should be further refined to differentiate between them.
5. **Elements of a scenario from one group are in conflict with one or more scenarios from other groups**. For example, the details of various scenarios may suggest conflicting policies or mechanisms to address the same situation. In this case, the need is to reconcile the work of the groups on a case-by-case basis. For example, two groups may identify identical scenarios that require some certification of services prior to deployment. One

group advocates central certification while the other argues for local certification. The team must use its judgment and experience to select a solution (e.g., central certification, local certification, or a "third way"). A different approach may be necessary when differing mechanisms reflect real differences in context or SOA governance needs across groups. In this case, the organization may support different certification strategies for different contexts.

The result of these activities is a single set of consistent, implementable policies across the various groups that have been fully validated against governance drivers.

3.7 Activity 6: Customize Policies to Fit SOA Governance Framework

The primary goal of the customization activity is to fold the findings of the various teams, now reflected in a single set of implementable policies and other elements, back into the organization-wide SOA governance approach. Customizing policies requires the consolidation team to analyze whether the consolidated scenarios and policies are consistent with the selected SOA governance framework. Several interesting cases may occur:

1. Where the policies do not provide complete coverage of the framework, a decision must be made whether policies should be added or are not needed because the missing coverage represents areas that are unnecessary for the organization.
2. Where policies have been developed that do not map to the framework, the framework may need to be extended. This need may arise when the organization has specific SOA requirements that are not addressed by the selected SOA governance framework. One possible solution is to look at other frameworks which may provide elements missing from the selected framework.
3. Where policies are inconsistent with the framework, either the policies or the framework needs to be modified.

A secondary goal of the customization activity is to identify how the policies can be implemented and supported across the organization. The information captured in the scenario template (see Table 1 on page 13) related to policy and policy implementation is a valuable starting point for developing the organization's overall SOA governance strategy.

3.8 Evolve and Iterate

It is unlikely that all the necessary policies required for SOA governance can be established in a single iteration of the process for these reasons:

- The SOA effort in an organization matures and creates governance needs in new areas.
- There are changes in technology.
- Business goals and needs change.

The organization can accommodate these areas of change by periodically iterating through the process to identify new scenarios and the corresponding SOA governance policies and implementation mechanisms to support them.

4 Conclusions

Vendor and other existing SOA governance frameworks are a useful starting point for an organization adopting SOA. However, mandating an existing framework without considering the unique needs of the organization may result in inefficiencies, overkill, or, even worse, complete failure.

Rather than by fiat, adoption of an SOA governance framework should start with an analysis of organizational needs for SOA governance. We believe that the scenario-based technique described in this report provides a mechanism that is vendor neutral, compatible with existing SOA governance frameworks, simple, and easily scalable. The technique can be used in the early phases of SOA adoption to lay the foundation for organization-specific SOA governance and can also provide insight into a sequencing of policies based on actual needs. The technique can also be used to capture the inevitably changing needs for governance as SOA is deployed. Where needs are changing, groups can be asked to consider modifications to existing scenarios and to generate new scenarios that capture their improved and changing understanding. These changes can then be related back to the existing SOA governance framework.

Appendix Examples of SOA Governance Frameworks

This appendix is a list in alphabetical order of some of the SOA governance frameworks that were studied for the creation our approach. The list is included simply to show examples of existing frameworks. The SEI does not endorse the products or services listed below.

AgilePath's SOA Governance Model

According to AgilePath, "SOA governance is the definition, implementation and ongoing execution of an SOA stakeholder decision model and accountability framework that ensures an organization is pursuing an appropriate SOA strategy, aligned with IT and business goals, and is executing that strategy in accordance with guidelines and constraints defined by a body of SOA principles and policies." The AgilePath four-tier governance model is represented in Figure 4 [8]. This model is tied to an integrated policy enforcement model and to the concept of governance performance management. Additional information can be found at http://www.agile-path.com.

Figure 4: AgilePath's Governance Model

CBDi SAE SOA Governance Framework

According to CBDi, "SOA governance is the part of IT governance that refers to the organizational structures, policies, and processes that ensure that an organization's SOA efforts sustain and extend the organization's business and IT strategies, and achieve the desired outcomes." The CBDi governance model is represented in Figure 5 as a set of different views that address the how, what, who, and when aspects of SOA governance [9]. Additional information can be found at http://www.cbdiforum.com.

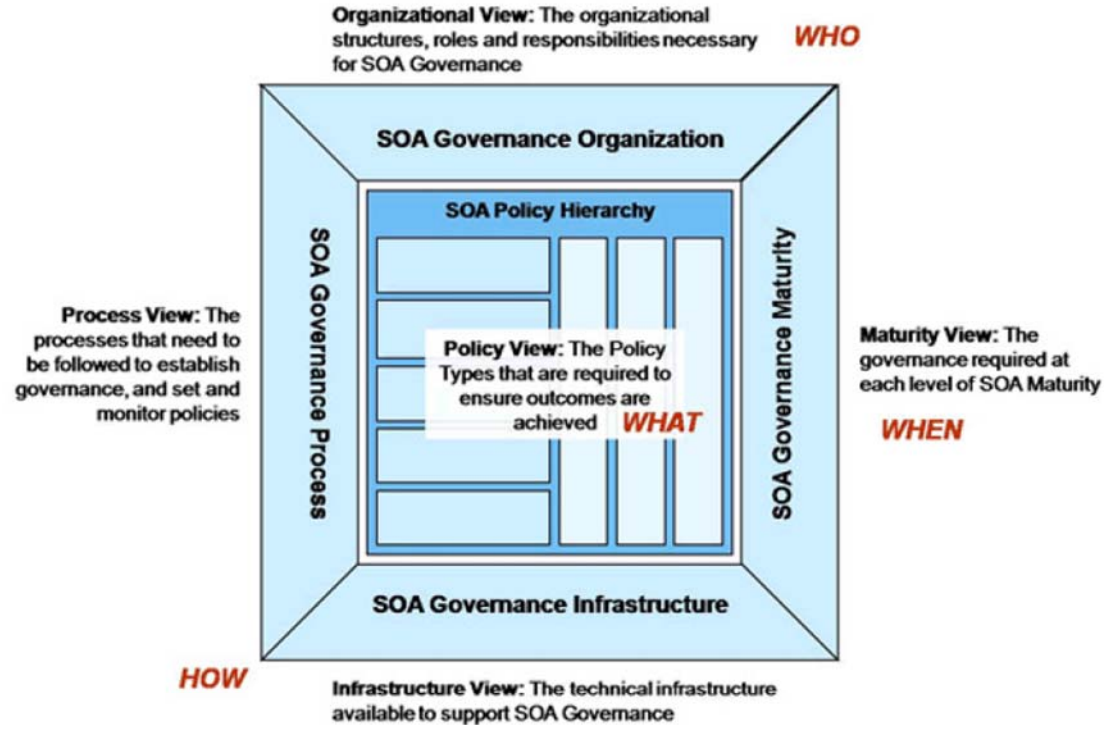

Figure 5: CBDi-SAE SOA Governance Framework

IBM SOA Governance and Management Method

According to IBM, "SOA governance is an extension of IT governance specifically focused on the life cycle of services, metadata and composite applications in an organization's service-oriented architecture." Consistent with this definition, the IBM SOA governance model is based on a life cycle for SOA governance, in which each phase has defined steps and deliverables, as shown in Figure 6 [4]. Some of the specific elements that are covered in the framework are centered on the life cycle for services, from identification to deployment to consumption. Additional information can be found at http://www-01.ibm.com/software/solutions/soa/gov/.

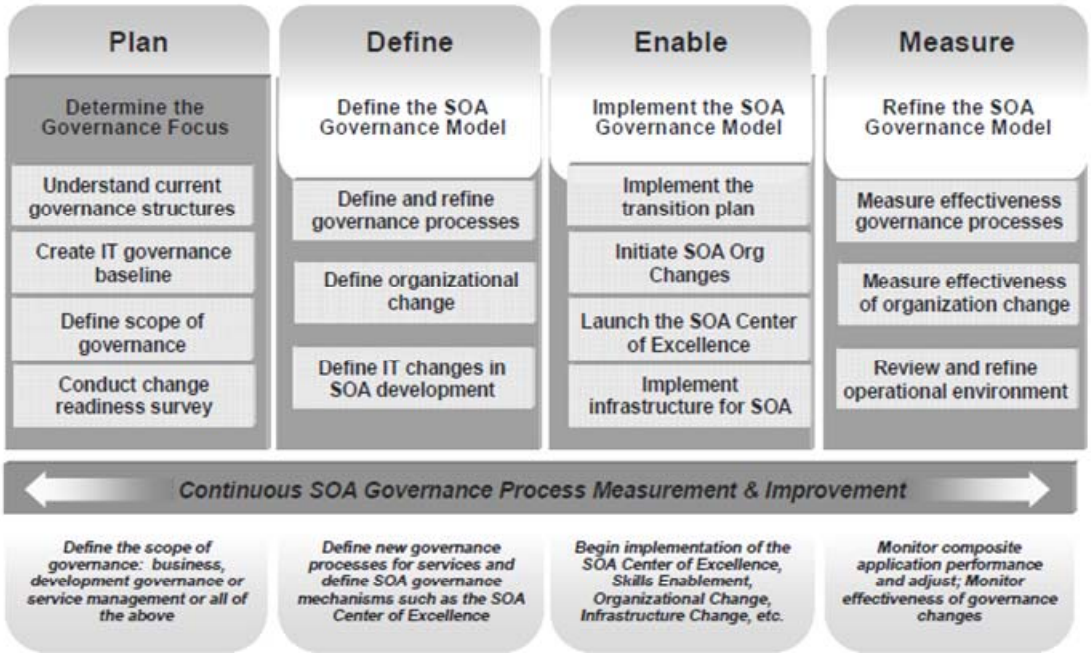

Figure 6: IBM SOA Governance and Management Method

Information Technology Infrastructure Library (ITIL)

ITIL is a framework developed by the OGC (Office of Governance Commerce) on behalf of the British government. ITIL is composed of best practices and policies for IT service management. ITIL v3 is divided into Service Strategies, Service Design, Service Transition, Service Operation, and Continual Service Improvement, as shown in Figure 7 [19]. Even though the scope of IT services is beyond services in the SOA context, much of the guidance in ITIL has application to SOA service development, deployment, and management. Additional information can be found at http://www.ogc.gov.uk/, http://www.itil.org, and http://www.itil-officialsite.com/.

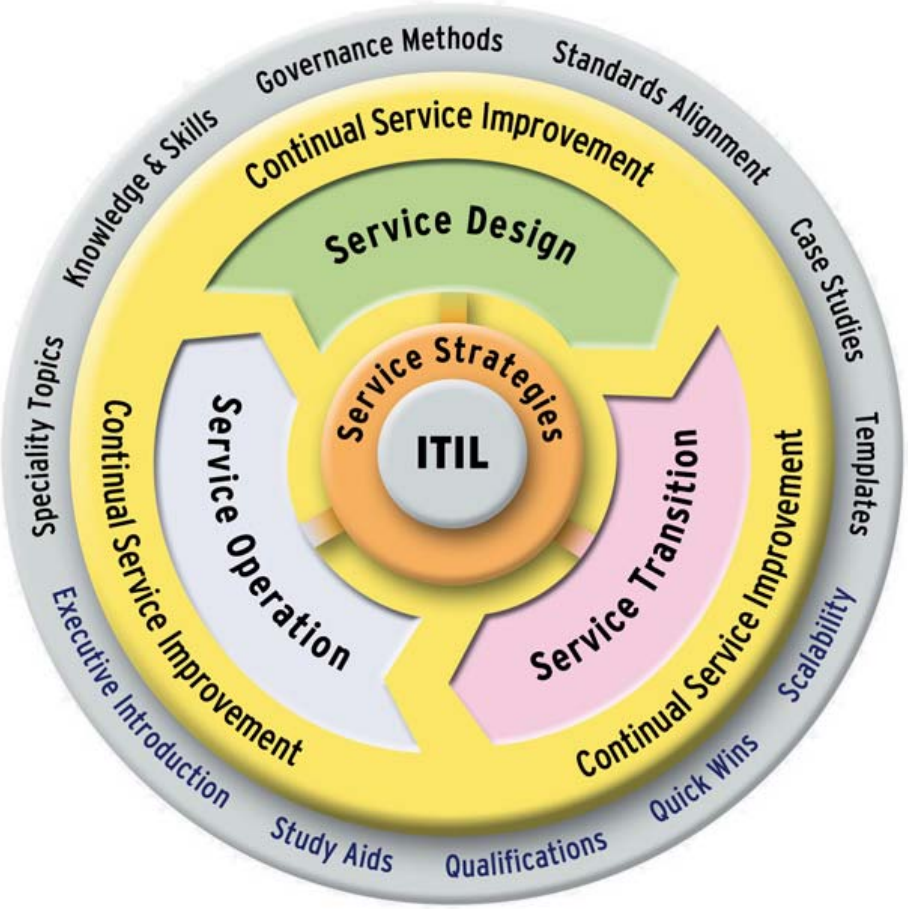

Figure 7: ITIL Core Framework

Oracle SOA Governance Model

According to Oracle, "SOA Governance can be defined as the interaction between policies (what), decision-makers (who), and processes (how) in order to ensure SOA success." The Oracle SOA governance model includes a set of best practices and a six-step process to define SOA governance. It is organized around key leverage points for SOA governance policies, as shown in Figure 8 [5]. Additional information can be found at http://www.oracle.com/technologies/soa/soa-governance.html.

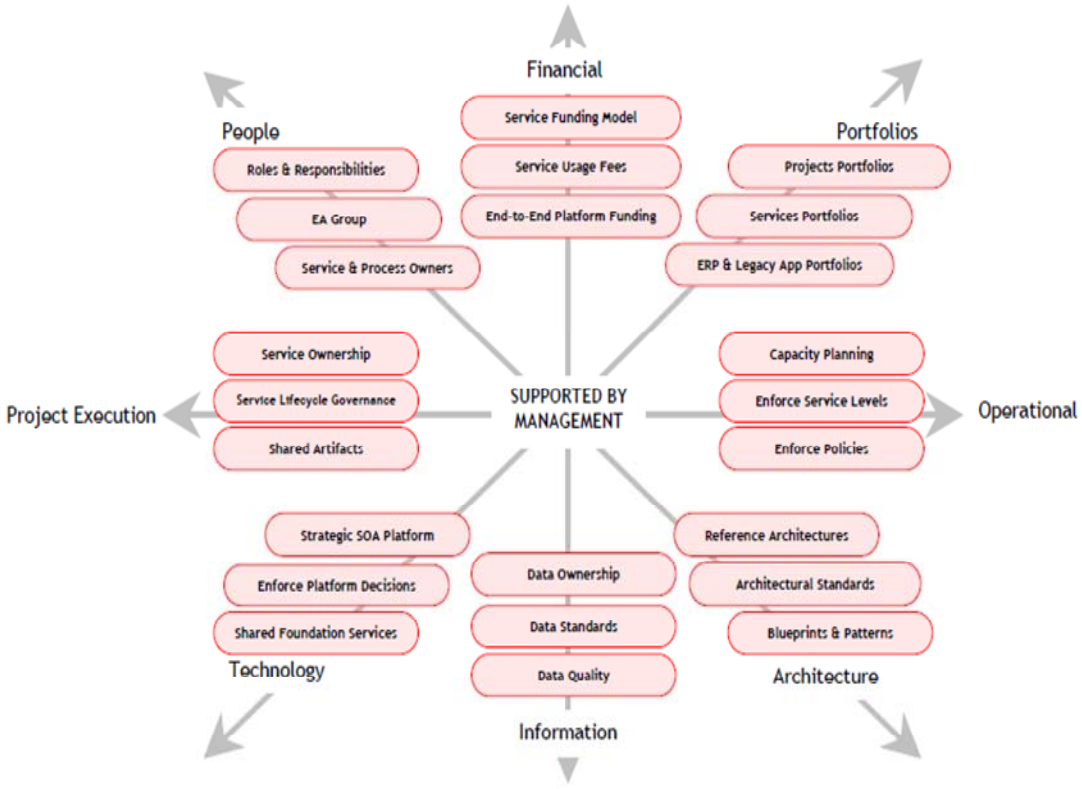

Figure 8: Oracle SOA Governance Model (Key Leverage Points for Policies)

SOA Software SOA Governance Model

According to SOA Software, "SOA Governance is about making sure that the enterprise builds the right things, builds them right, and makes sure that what it has built is behaving right." SOA Software's SOA governance model is organized around an integrated SOA governance solution that is automated by its product portfolio. At a high level, the model is organized around planning, development, operational, and policy governance, as presented in Figure 9 [6]. It includes a set of best practices and use cases to aid in SOA governance implementation. Additional information can be found at http://www.soa.com/.

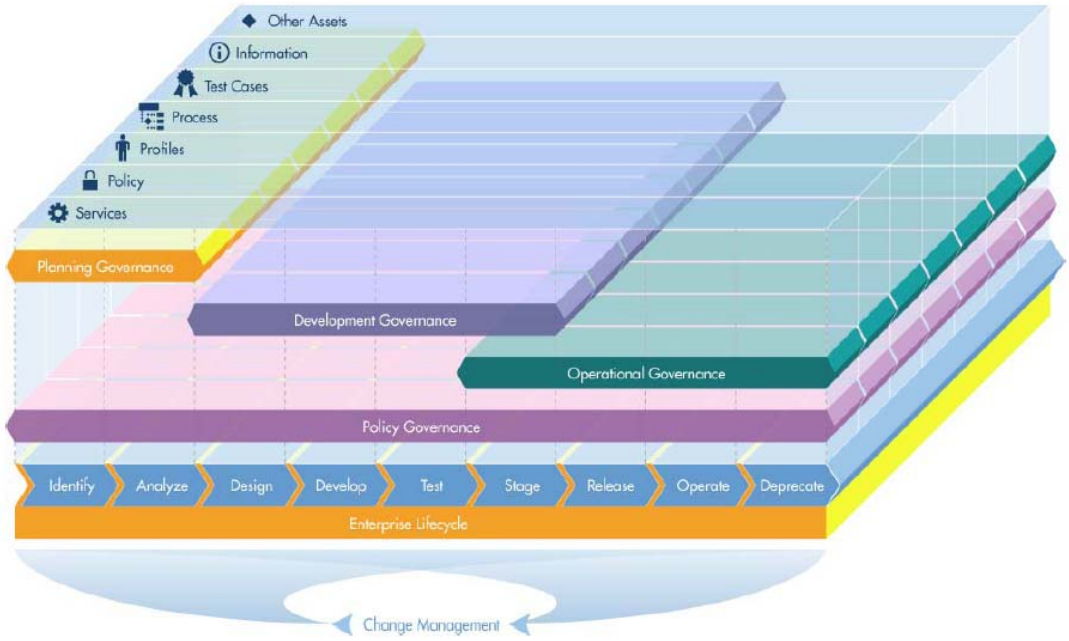

Figure 9: SOA Software Integrated Governance Model

Software AG SOA Governance Model

According to Software AG, “SOA governance is [a] subset of IT governance. It establishes policies, controls, and enforcement mechanisms required for successful SOA adoption by giving IT visibility and control over their SOA development and deployment.” The model is based on service-level life cycle governance that is supported by their product suite [7]. Additional information can be found at http://www.softwareag.com/Corporate/products/wm/soa_governance/default.asp.

References

URLs are valid as of the publication date of this document.

[1] P. Malinverno, “Service-Oriented Architecture Craves Governance,” Gartner Group, Research Report, 2006.

[2] “InfoWorld Research Report: Service Oriented Architecture (SOA),” 2007. [Online]. Available: http://www.s2.com.br/s2arquivos/403/multimidia/197Multi.pdf. [Accessed: May 1, 2009].

[3] B. Woolf, “Introduction to SOA Governance,” June 13, 2006. [Online]. Available: http://www.ibm.com/developerworks/library/ar-servgov/index.html. [Accessed: May 1, 2009].

[4] W. A. Brown, G. Moore, and W. Tegan, “SOA Governance - IBM’s Approach,” 2006. [Online]. Available: ftp://ftp.software.ibm.com/software/soa/pdf/SOA_Gov_Process_ Overview.pdf. [Accessed: May 1, 2009].

[5] M. Afshar, “SOA Governance: Framework and Best Practices, Version 1.1,” Oracle, May 2007. [Online]. Available: http://www.oracle.com/technologies/soa/docs/oracle-soa-governance-best-practices.pdf. [Accessed: May 1, 2009].

[6] “SOA Software White Paper: Integrated SOA Governance,” 2007. [Online]. Available: http://www.soa.com/index.php/user_account/download.php?r=Resource_Center/White_Papers/SOASoft_Int_SOA_Governance.pdf. [Accessed: May 1, 2009].

[7] F. Castaldini, “Software AG White Paper: SOA Governance and CentraSite: Ensuring SOA Success with Effective, Automated Control Throughout the Lifecycle,” 2008. [Online]. Available: http://www.softwareag.com/Corporate/Images/SAG_SOAGov_ CentraSite_WP_Apr08-web[1]_tcm16-40610.pdf. [Accessed; May 1, 2009].

[8] E. Marks, SOA Governance: Evolving Governance in the Services-Driven Enterprise. New York: John Wiley & Sons, 2008.

[9] P. Allen, “SOA Governance: Challenge or Opportunity?” SOA Best Practice Report, CBDI Forum, 2008. [Online]. Available: http://www.cbdiforum.com/report_ summary.php3?page=/secure/interact/2008-04/challenge_opportunity.php&area=silver. [Accessed: May 1, 2009].

[10] S. Taylor and M. N. Iqbal, ITIL Service Strategy. London: The Stationary Office, 2007.

[11] A. Arsanjani and K. Holley, “Increase flexibility with the Service Integration Maturity Model (SIMM),” Sept. 30, 2005. [Online]. Available: http://www-128.ibm.com/developerworks/webservices/library/ws-soa-simm/ [Accessed: may 1, 2009].

[12] D. Sprott, “The SOA Maturity Model—Part II,” SOA Best Practice Report, CBDi Forum, March 22, 2006. [Online]. Available: http://www.cbdiforum.com/report_summary.php3?page=/secure/interact/2006-03/The_SOA_Maturity_Model_Part2.php&area=silver. [Accessed: May 1, 2009]

[13] F. I. Kenney, D. C. Plummer, and K. Harris, K., “No 'Leader' Exists in SOA Governance ... At Least Not Yet,” Gartner Group, Research Report, 2007.

[14] L. Thé. “SOA Governance - 'Not Much' Success, Panelists Say,” Application Development Trends, Feb. 14, 2008. [Online]. Available: http://www.adtmag.com/article.aspx?id=22055. [Accessed: May 1, 2009].

[15] A. G. Sutcliffe, N. A. M. Maiden, S. Minocha, and D. Manuel, “Supporting scenario-based requirements engineering,” IEEE Transactions on Software Engineering, vol.24, no.12, pp.1072-1088 (Dec. 1998).

[16] R. Kazman, M. Klein, and P. Clements, “ATAM: Method for Architecture Evaluation,” Software Engineering Institute, Carnegie Mellon University, CMU/SEI-2000-TR-004, 2000.

[17] “SOA Governance User Survey: Best Practices for SOA Governance User Survey,” Software AG, 2008. [Online]. Available: http://www.softwareag.com/Corporate/res/SOAGovernanceSurvey.asp. [Accessed: May 1, 2009].

[18] “Increasing the Effectiveness and Efficiency of SOA Through Governance, 2008 SOA Governance Survey Report,” Oracle, 2008. [Online]. Available: http://www.ebizq.net/white_papers/10075.html. [Accessed: May 1, 2009].

[19] “ITIL Knowledge,” ITIL.org, 2009. [Online]. Available: http://www.itil.org/en/. [Accessed: May 1, 2009].

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	June 2009	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
A Scenario-Based Technique for Developing SOA Technical Governance	FA8721-05-C-0003

6. AUTHOR(S)

Soumya Simanta, Ed Morris, Grace A. Lewis, Sriram Balasubramaniam, and Dennis B. Smith

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2009-TN-009

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116	

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

A well-known problem within the service-oriented architecture (SOA) community is the need to establish effective SOA governance procedures to enable an organization-wide SOA initiative. A number of organizations and vendors address this problem through SOA governance frameworks that provide models, procedures, and tools for SOA governance. Many of these SOA frameworks are general purpose because they are intended to be useful for a diverse customer base. However, while designed for a wide customer base, vendor SOA frameworks are narrowly focused to work with the specific tools of the vendor. A critical problem for an organization when implementing SOA governance is to customize vendors' offerings to its specific technological and management context. In this technical note, a lightweight and extensible technique is proposed, one that employs scenarios to tailor existing SOA governance frameworks to the specific needs of an organization.

14. SUBJECT TERMS	15. NUMBER OF PAGES
SOA, SOA governance, scenario-based technique, service management, policy, best practices, risks	37

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL