zapthink
white paper

# THE VALUE OF SOA GOVERNANCE

ZapThink, LLC • 108 Woodlawn Road • Baltimore, MD 21210 • info@zapthink.com • www.zapthink.com

# THE VALUE OF SOA GOVERNANCE

June 2007

*Analyst: Jason Bloomberg*

**Abstract**

The definition of *corporate governance* is creating, communicating, and enforcing policies in a corporate environment. Governance is the key to balancing executive control with employee and customer empowerment across the enterprise. While many corporate governance activities don't directly involve the information technology (IT) department, the enterprise does call upon IT to provide tooling for automating policy creation and enforcement, when it's possible to represent policies in a machine-understandable format.

*Service-Oriented Architecture* (SOA) is an approach to organizing IT resources to meet the changing needs of the business in flexible ways. Governance is an essential part of any SOA implementation, because it ensures that the organization applies and enforces the policies that apply to the Services that the organization creates as part of its SOA initiative. But more importantly, organizations can leverage SOA best practices to represent policies broadly in such a way that the organization can achieve better policy management, flexibility, and visibility into policy compliance across the enterprise.

## I. The Business Motivation for Governance

The concept of governance is drawing substantial attention in corporate boardrooms and technical meetings alike, as companies struggle with complex regulatory compliance pressures, increasing globalization, enhanced competition, and the maturation of their markets. Many organizations have learned that one key differentiator between top performing organizations and their less successful counterparts is the ability to monitor their performance, and then evaluate those results and act on the lessons they've learned. As a result, governance both empowers and controls. It empowers stakeholders to make and implement decisions, while it provides for management controls to maintain corporate policies. Of course, the notions of corporate governance broadly, and IT Governance in particular nothing new. As this paper will illustrate, SOA Governance is a subset of IT Governance with an emphasis on SOA and a new way of looking at the problem of governance that SOA brings to the table

*People don't particularly like to be governed.*

Perhaps the greatest challenge, however, facing organizations who are implementing some kind of corporate governance initiative is simply the fact that people don't particularly like to be governed. From the corner office, governance appears to be a clear win, and yet the rank and file tend to resist the implications of governance, especially when they perceive the controls in place to be too onerous.

And yet, management requires control, and in the absence of an effective governance program, such control typically devolves into micromanagement—which is every bit as bad, if not worse than an overbearing governance initiative. The challenge for management, therefore, is to scale management control while avoiding micromanagement—empowering users while avoiding policy breaches.

### What is governance?

To achieve this balance between empowerment and control—and leverage performance monitoring and decision making without dissention in the ranks—requires effective corporate governance, which we define in this way:

- Establishing and communicating the policies that employees must follow
- Giving employees the tools they need to be compliant with those policies
- Enforcing policies
- Providing visibility into the levels of compliance in the organization
- Resolving any deviations from established policy

There's nothing in the above five bullets that requires that management involve technology in any way, and in fact, most managers today handle corporate governance in an essentially manual fashion, or where IT serves a secondary

support role. For example, let's consider the case of a corporate nondiscrimination policy. Corporate council establishes the policy by basing it on the law. They then communicate it to employees via a memo. Next, human resources prepares some instructions on following the policy, which are the tools they provide to employees. The organization then handles enforcement through a formal complaint and investigation process, which also affords management visibility into the levels of compliance with the policy. Finally, the organization institutes a reprimand and dismissal policy for dealing with violations policy breaches.

Note that in the example above, technology is not involved, but technology likely supported or enabled the governance processes. Perhaps someone sent an email to inform HR about the violation or posted the policy on an intranet,. And yet, while a policy like a corporate nondiscrimination policy doesn't lend itself to IT involvement, other policies can better take advantage of the resources IT can provide. For example, today's businesses must keep confidential customer and employee information that resides in various databases private and secure. Enforcing confidentiality will then be a combination of manual tasks like educating medical personnel, combined with automated processes like requiring password protection for documents and processes.

*One challenge for SOA governance becomes identifying which policies are automatable, and then leveraging the benefits of SOA to automate the enforcement of those policies.*

Therefore, one challenge for IT governance—and consequently, for SOA governance—becomes identifying which policies are automatable, and then leveraging the appropriate technologies to automate the enforcement of those policies in a flexible way, And yet, even for the most mature SOA implementations, many governance tasks fall outside the realm of automation. Even so, when architecture drives IT governance, taking a Service-oriented approach to such architecture can improve the policy management, flexibility, and visibility necessary for IT governance, and more broadly, corporate governance.

### What is a policy?

In order to identify which policies lend themselves to automation, it's worthwhile to take a look at the definition of *policy*. According to Merriam-Webster, a policy is "a high-level overall plan embracing the general goals and acceptable procedures," which is essentially a broad business definition that would apply in the United Nations as well as the corporate boardroom. However, it's clearly quite difficult to automate a policy given that definition. Another definition of policy is "a set of rules that apply to the performance or behavior of a system and its users," which is a more technical definition of policy that is easier to automate.

This dichotomy between different perspectives on the nature of policies in the organization presents challenges across the organization as both business and IT managers get a handle on automating governance. Before SOA, business and IT managers shared little common ground with respect to policy enforcement. SOA, however, helps automate policy activities by treating policies as external artifacts that serve as what ZapThink refers to as *metadata* – policy information represented in a standard, machine readable format. Once policies appear as metadata, it becomes possible to bridge the gap between the business and IT perspectives on policies. Note that we're using the term *metadata* broadly to include artifacts external to the application code and the data the applications deal with. The physical representation of a policy might be an object like an XML file.

### Defining the governance framework

The starting point for any SOA governance initiative, therefore, centers on the policies that the organization values and requires and how they will enforce those policies in order to effectively balance empowerment and control. In order to get a handle on the scope of such an initiative, it is essential to put together a *governance framework*. To create a governance framework, you should answer the following questions:

- Which policies are within the scope of the current iteration? Which policies should you implement first?
- Who in your organization is responsible for creating policies?
- Which policies are automatable?
- How will you create and communicate policies?
- How will you represent policies? In other words, what is the format for your policies?
- How will people within your organization discover policies?
- What tools will people use to follow policies?
- How will management get visibility into policy compliance?
- How will you deal with policy violations? What mitigation approaches will you use?

*The governance framework becomes an outline of your governance initiatives.*

This governance framework then becomes an outline of your governance initiatives. In early iterations, it will be a simple document, but in each successive iteration, it is important to flesh it out, delineating in increasing detail how you will define and enforce policies as your governance initiative matures.

## II. IT's Dual Governance Role

Once your initial governance framework is in place, it soon becomes important to identify the role IT has in implementing the governance initiative. It's important to note, however, that there are several different activities that organizations must undertake to tackle corporate governance, including the following:

- *Communication* – the simple act of communicating policies, either face-to-face, via one-to-one remote communication media like email, or via one-to-many approaches like the corporate Intranet.
- *Training* – formal and informal training on policies and procedures
- *Human management* – people working with their direct reports to ensure understanding of and compliance with corporate and it policies.
- *Knowledge management* – leveraging a centralized repository of policies and associated best practices.
- *Automation* – taking advantage of IT infrastructure to implement policy enforcement directly.

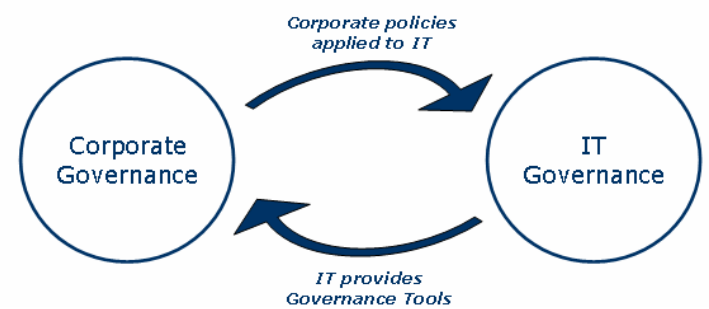
*IT governance describes how people entrusted with the authority over some aspect of the business will consider IT in their supervision, monitoring, control and direction of that business entity.*

Of these activities, IT clearly focuses on automation, and may also provide various communication and knowledge management capabilities as well. IT governance describes how people entrusted with the authority over some aspect of the business will consider IT in their supervision, monitoring, control and direction of that business entity. IT governance is an integral part of enterprise

governance and consists of the leadership and organizational structures and processes that ensure that the organization's IT sustains and extends the organization's strategies and objectives.

And yet, there is more to IT governance than leveraging IT capabilities for corporate governance. It's also important to remember that the IT department is part of the organization, just like any other division, and corporate policies apply to IT as well. As a result, there is a dual role for IT governance, as the following figure illustrates:

Governance Relationships (Step 1)




Source: ZapThink

In fact, this dual role of IT governance carries over into the discussion of architecture governance, as well as SOA governance.

### The role of architecture in IT governance

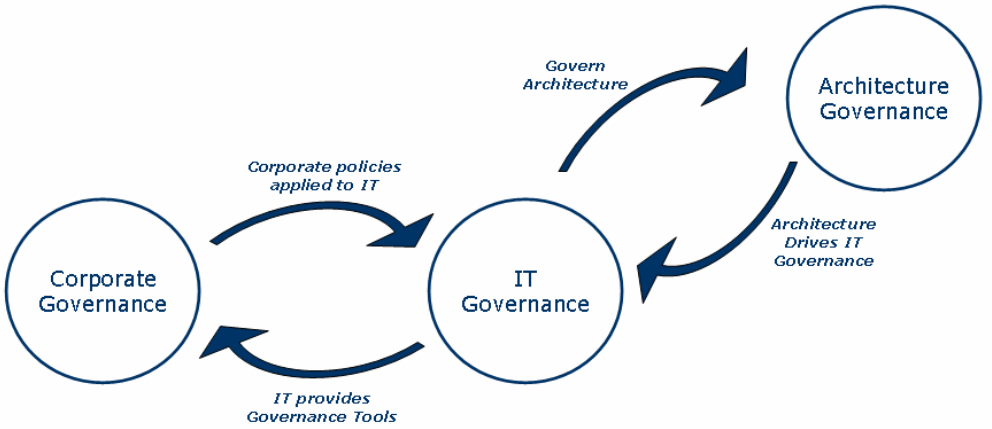
As organizations grow, either organically or via acquisitions, their IT efforts tend to decentralize. Such decentralization typically leads to redundant, incompatible approaches to solving business problems. On the other hand, IT centralization efforts often run into roadblocks of their own, as well. As a result, one of the key goals of IT governance initiatives is how to decentralize IT responsibility without leading to redundant or incompatible capabilities, and while maintaining sufficient centralized control, in addition to specifying the decision rights and an accountability framework for encouraging desirable behavior in the use of IT resources .

The answer to this question centers on *architecture*, which IEEE defines as the fundamental organization of a system embodied by its components, their relationships to each other and to the environment and the principles guiding its design and evolution. Architecture is in many ways is the cornerstone of IT governance, because it provides the overall organizational guidelines for all of IT. In addition, architectural processes are the best way for the IT organization to implement IT governance. It is also necessary for an architecture board to drive IT governance within the organization.

In fact, it is possible to extend the dual role IT governance has for corporate governance to the consideration of architecture governance as well. After all, not only does architecture drive IT governance, it is also important to govern the architecture initiatives, as shown in the figure below:

Governance Relationships (Step 2)




Source: ZapThink

*SOA governance at its core focuses on establishing a framework for assuring Service quality over the course of the SOA lifecycle.*

It is within the context of architecture governance as the above figure illustrates that SOA governance takes place. SOA governance at its core focuses on establishing a framework for assuring Service quality over the course of the SOA lifecycle. To ensure proper SOA governance, organizations must manage Services in the context of specific business, IT and regulatory policies that apply to those Services and the consumers that interact with them.

Fundamentally, however, this core of SOA governance is SOA Governance "in the narrow," in that it focuses on governance of Services in the context of the SOA initiative, rather than on IT governance more broadly. SOA governance "in the broad," however, focuses on how the transition to Service-oriented approaches affects the broader area of corporate IT governance. SOA governance in the narrow, therefore, focuses on the creation, communication, and enforcement of policies that apply both to the design time aspects of Service artifact creation, publication, and reuse, as well as the runtime aspects of Service operations, including service levels and the management of Quality of Service (QoS) metrics.

### Achieving SOA benefits with proper governance

One way of thinking about the benefits SOA provides to the organization is to consider direct benefits for the business, like improved regulatory compliance, indirect business benefits, including the optimization of business processes in environments of business change, and direct benefits to the IT organization, including reducing documentation, increasing IT transparency, and improving efficiency and reuse. To fully understand the importance of SOA governance, then, let's take a look at what happens when a SOA initiative does not have proper governance in place.

*Lack of governance can lead to unexpected noncompliance with regulations.*

First of all, an ungoverned SOA implementation can lead to unintended consequences, including the lack of sufficient reuse, inconsistent, duplicate, or incompatible Services, increased support costs, failure to satisfy service-level agreements, and challenges with updating the versions of Services. Furthermore, because regulatory compliance can be very complex, with multiple regulations and jurisdictions in play at once, and also because regulations are essentially arbitrary, lack of governance can lead to unexpected noncompliance with regulations, by failing to associate key policies with Services.

Lack of sufficient governance can also lead to various security breaches by allowing arbitrary access to Services, leading to the exposure of confidential information, unwanted access to internal systems, and other security threats. Fundamentally, without proper governance, all of the business benefits of SOA, including increased agility, reduced costs, increased asset reuse, and improved visibility, are all at risk. In other words, the agility benefit of SOA is a two-edged sword

*A poorly organized SOA governance team can become a bottleneck or even a single point of failure for a SOA initiative.*

Organizational challenges can also impede a SOA initiative—or even, a SOA governance initiative. In fact, a poorly organized SOA governance team can become a bottleneck or even a single point of failure for a SOA initiative. Runtime performance can also derail a SOA rollout. Performance is always important for production applications, and adding a Service abstraction increases the performance overhead. Runtime SOA governance must address, not contribute to, such issues.

Fundamentally, however, SOA initiatives must cross the business/IT divide, and SOA Governance must lead the way since it establishes and controls the business value that SOA delivers. Insufficient business management support can also be an issue, if the lines of business relegate consideration of SOA entirely to IT. As a result, it's important to consider business-centric policies for leveraging the value of SOA as part of the governance framework.

## III. SOA Governance Starting Point

Since SOA involves enterprisewide architectural change, SOA governance should not focus solely on certain technologies or IT projects. And yet, organizations should still take an iterative approach to SOA broadly, as well as with SOA governance initiatives, which generally require a focus on SOA governance in the narrow before moving onto SOA governance in the broad.

One way to describe SOA governance in the narrow is the structuring of decision making authority for developing and modifying various SOA artifacts. Furthermore, SOA governance encompasses people and their roles, technologies and tools, as well as processes. Knowing where to start, and how to implement an initial SOA governance framework, then, become key questions early on in any SOA initiative.

### First steps for SOA governance

Proper planning is important to implementing SOA Governance successfully. First, create your governance framework, which should then lead to the following activities:

1. Publish the goals of your SOA initiative – define the business value of each phase of the SOA rollout and inform stakeholders and other interested parties about the initiative.
2. Define the SOA organizational structure – put together the SOA Center of Excellence, which should help to inform the SOA Governance team by fleshing out the necessary roles for the SOA initiative, including the SOA architect, Service developers, the SOA quality team, business analysts and process specialists, and operations and security personnel. Assign each member of this team a clear set of responsibilities. The SOA governance team should be a subset of this broader SOA team, but at the least the SOA governance team should include one representative from both business and IT, have the ear of upper management, and the mandate to decide SOA-related issues.

*The SOA governance team should include one representative from both business and IT, have the ear of upper management, and the mandate to decide SOA-related issues.*

3. Define SOA Governance processes – consider basic issues like Service design, development, deployment, operation, and change processes. Define the roles that are relevant for the various tasks in your SOA initiative.
4. Evaluate technical challenges for SOA governance – first, leverage SOA governance tools, including knowledge management and collaboration tools, as well as a SOA registry/repository for storing and managing SOA assets.
5. Determine governance of the SOA infrastructure –How will you integrate new technologies and tools into the existing IT landscape? How will you govern the various parts of your SOA infrastructure? You should consider design artifacts, interfaces, policies, and other artifacts that you should store in a SOA registry/repository, distributed intermediaries or an integration infrastructure like an ESB, management tools for runtime SOA Governance, and security tooling for handling authentication and authorization as well as security policy enforcement.

**Design time, runtime, and change time governance: governing the SOA lifecycle**

As part of your early SOA governance planning, it's critical to recognize that design time and runtime are two very different parts of the SOA lifecycle, and have divergent, yet overlapping needs, as decisions you make during design time directly influence runtime results. Design time spans Service planning, design, discovery, implementation and testing phases, while runtime encompasses deployment, operations, support, and versioning tasks.

There is another aspect of runtime governance that in many ways is a separate phase altogether: change time governance. By change time we mean changes to configurations and compositions of running Services. Such changes do not impact the underlying execution of the Services (which happens at design time), and yet managing and enforcing policies during this change time phase is an especially important part of the SOA governance picture.

To better delineate these SOA lifecycle phases, here are some of the activities various personnel might undertake in each phase:

*Design time:*

- An architect designs a Service contract or plans a Service implementation
- A developer searches for an existing Service before building a new one
- A developer requests that the SOA Competency Center approve the creation of a new Service
- A tester simulates a new Service to execute a test plan

*Runtime:*

- A business executive monitors a business process
- A manager confirms compliance of a Service with particular policies or service level agreements
- The CIO monitors levels of Service reuse
- The operations manager handles Service exceptions

*Change time:*

- A business analyst plans a change within a certain business process
- An IT administrator adjusts the Quality of Service requirements for a Service

*Governance of the SOA lifecycle combines the design time, runtime, and change time phases with the SOA governance framework.*

Governance of the SOA lifecycle combines the design time, runtime, and change time phases with the SOA governance framework. Basically, for each of the phases, the steps are to determine which policies are within scope, and then delineate how you will create, communicate, and enforce those policies in each phase, as well as obtain visibility into levels of compliance.

The goals of governance the SOA lifecycle include defining the relevant authority and providing transparency, defining and enforcing rules for Service creation, use, and management, managing change, measuring results, and optimizing Service behavior. It's important to note that the SOA lifecycle involves Services, related artifacts and roles. In practice, a business process analysis and optimization request could result in either creation of a new Service, modification of an existing Service or reuse of an existing Service as-is. SOA Governance provides guidance by describing how new Services move from planning and design to production, mandating the consideration of Service reuse or modification as appropriate, ensuring that necessary reviews are an integral part of each phase, and defining each person's role within the SOA lifecycle.

### Enforcing SOA policies: the importance of active governance

Organizations handle policy enforcement and noncompliance resolution for most business policies through various types of human interactions. For example, an executive team might enforce a nondiscrimination policy through management instructions and communication, and then enforce any deviations of that policy by censuring or dismissing the offender. In such instances, however, management encourages policy compliance, but is not able to fully prevent breaches from occurring.

In the world of SOA, however, automated policies may have a greater level of enforcement, depending upon the policy management infrastructure in place. Enforcement of a security policy, for example, should ideally offer no alternatives to compliance—in other words, the infrastructure should prevent undesirable behavior. We refer to this level of policy enforcement runtime governance, because the governance infrastructure must take a direct, active role in ensuring the compliance with particular policies during runtime. Some examples of active governance include enforcing limitations on the number of Services in a composition, enforcing specific Service reuse metrics, as well as most security policies.

### Timing for initial deployment of SOA governance

It's important to tackle the implementation of SOA governance iteratively, as an integral part of an overall iterative approach to the SOA rollout. Nevertheless, it's critical to incorporate some aspects of SOA governance into the earliest stages of SOA planning. At the least, initial SOA governance iterations should include defining and publishing your SOA goals, defining the SOA governance organizational structure, and sketching out initial SOA Governance processes.

Early iterations should also include the evaluation of certain technical issues, and often the purchase of a SOA governance tool. Ensure that such tools will scale through future iterations. It also makes sense for your architect team to

formulate a standard approach to representing policies, in order to facilitate interoperability into the future.

### The SOA lifecycle and the need for constant governance optimization

Managing the SOA during change time means managing the change control of artifacts throughout the planning to production cycle. Remember that one of the primary motivations for SOA is agility, and thus the SOA implementation should support ongoing change. As a result, there is no final or complete state for any SOA implementation; rather, organizations must constantly seek to optimize their Services, processes, and associated governance tasks.

In particular, organizations should focus on the continual optimization of the lifecycle processes, as well as the various SOA artifacts that will continue to change over time. To support this change, it's essential to implement flexible integration with different metadata stores. Furthermore, organizations should realize that policies themselves have a lifecycle as well. Be sure to hammer out how to create, implement, version, and deprecate policies just as you would the Services that they apply to.

## IV. Expanding the Context for SOA Governance

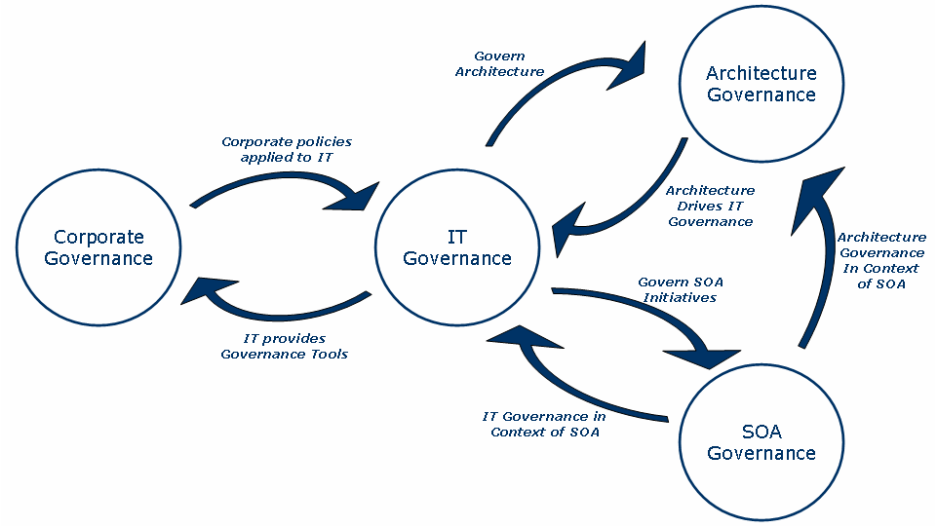
You've taken the advice of this paper and implemented SOA governance as a core part of your early SOA planning and rollout. As your SOA implementation matures, however, it becomes more straightforward to leverage the benefits of SOA for governance more broadly than simply the governance of Services as part of the SOA lifecycle. This SOA governance in the broad goes beyond the governance of Services within a SOA initiative (SOA governance in the narrow), and essentially considers how having SOA in place will improve IT governance overall, and more broadly, corporate governance as well. Basically, SOA governance in the broad involves governance in the context of SOA more so than governance of SOA initiatives. The question is, therefore, what benefits does SOA provide that are particularly appropriate for satisfying governance requirements.

*SOA offers three core capabilities that enable governance: policy management, visibility, and flexibility.*

In fact, SOA offers three core capabilities that enable governance: policy management, visibility, and flexibility. Because SOA represents policies as metadata, it's possible to represent a broader set of policies as metadata than simply those that apply to Services. Because SOA abstracts heterogeneous data sources across an organization, SOA techniques can provide visibility into levels of compliance with policies across the company. And finally, SOA's core agility benefit helps organizations deal with policy change.

In fact, while SOA governance in the narrow applies IT governance to the governance of SOA initiatives, SOA governance in the broad both places IT governance in the context of SOA, as well as architecture governance in the context of SOA, as the figure below illustrates:

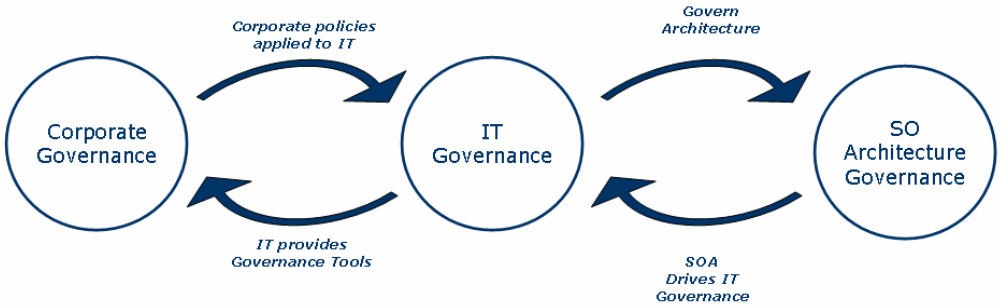
Governance Relationships (Step 3)




Source: ZapThink

The figure above, however, suffers from an element of serious shortsightedness. While architecture governance enables IT governance as we discussed earlier, ZapThink believes that over time, Enterprise Architects will take a Service-oriented approach to architecture governance, as SOA best practices become the standard approach for organizing IT resources to meet the needs of the business. At that point, the architecture governance and SOA governance circles will merge into a Service-oriented approach to architecture governance, as the final illustration below suggests:

Governance Relationships (Step 4)




Source: ZapThink

In fact, ZapThink believes the figure above illustrates the true future of SOA governance for enterprises around the world, as they fully implement SOA.

### Policy management, visibility, and flexibility

SOA governance requires that organizations take business policies, typically in written form, and transform them into metadata-based rules that can help automate the process of validating and enforcing compliance with those policies in both design time and runtime environments. Companies must then manage policies through their entire lifecycle. In general, policy lifecycle management

within SOA focuses on ensuring the quality, performance and applicability of available Services, enabling Service consumers to discover and reuse Services as well as other artifacts, managing Service versions, handling the security of Services and other SOA artifacts, and assessing and managing the impact of change across all Service consumers. Managing policies also includes providing visibility into whether people are following policies, as well as handling policy infractions. Such policy management tasks are also an inherent aspect of IT governance, as well.

*At design time, the goal of SOA policy management is to detect and resolve quality issues before putting the Services into production.*

At design time, the goal of SOA policy management is to detect and resolve quality issues before putting the Services into production. At runtime, then, organizations must also implement runtime policy management for monitoring and automatically enforcing policies during the usage of Services. Such runtime policies may focus on security, QoS, or other requirements for the behavior of the Services in a production environment.

SOA policies require the following capabilities:

- *Policy Management* – defining and maintaining reusable policies over the course of the Service lifecycle. Moving proprietary policies from siloed systems to central policy management that can interface with the SOA infrastructure in a standards-based way.
- *Policy Association* – applying policies to Services and other artifacts, often through the use of a SOA registry. It's usually preferable to publish such policies to the registry similar to Service contracts.
- *Policy Enforcement* – Enforce SOA policies in practice, either via the registry/repository at design time, via SOA management tools at runtime, or via various types of policy enforcement points on the network, depending upon the type of policy.
- *Policy Reporting* – providing visibility into policy compliance via reports that the registry/repository can store.

Here are some practical steps for automating policies. First, conduct a policy inventory to uncover the policies that are a priority. Next, decide which policies are automatable. In other words, identify those policies that you can represent as metadata that your policy management tools can understand. Then, decide on level of granularity for your policies. Note that not every policy management or enforcement tool represents policies with the same level of detail, so it's important to develop a consistent format for representing the policies.

At this point you must translate policies into a system-understandable format. Standards like WS-Policy and WS-SecurityPolicy can aid somewhat with this formatting issue, but unfortunately, these standards can only help in rather narrow situations. In the general case, it will be important to either develop your own XML-based policy specification, or encode the policies directly into the policy enforcement system, which represents policies according to the tools' own internal specifications.

Once you have fully defined your policies, you must figure out how to enforce policies in practice. Policy enforcement essentially depends on the type of policy. For example, an XML firewall might enforce a security policy, while a registry/repository might enforce a Service reuse policy. SOA management tools enforce many runtime policies, while identity and access control solutions are adept at enforcing access management policies.

Finally, it's important to identify techniques for long-term policy maintenance, as the organization creates, modifies, and retires its policies. In addition to these

policy lifecycle considerations, you should also consider how you will scale policy management, either through a centralized or federated approach.

### Balancing empowerment and control with SOA governance

*As companies adopt SOA as enterprise architecture, SOA governance becomes the primary way that companies can establish principles for the control of their organizations.*

As the IT resources business users require become more flexible and generally better able to meet an increasingly broad range of business needs, IT becomes inextricably intertwined in the daily operations of the business. In such a situation, enterprise architecture becomes a critical enabler of governance, and as companies adopt SOA as enterprise architecture, SOA governance becomes the primary way that companies can establish principles for the control of their organizations.

Many organizations are pursuing SOA to abstract their IT infrastructures from inflexible, monolithic application silos to loosely-coupled, business-oriented Services that various parts of the organization can easily share and reuse. SOA can have a transformative impact on organizations by making them more agile, and it can also reduce the cost of integration and application development as well. In effect, SOA enables IT to provide more powerful, flexible tools to business users, as they leverage the power of these business Services in increasingly flexible processes.

With greater user empowerment, however, comes greater responsibility. The last thing IT management desires is to enable an increased circle of users to wreak havoc with their more powerful capabilities. Hence SOA governance becomes a critical part of any SOA initiative. Basically, it's a mistake for organizations to discount governance as optional or a later-phase aspect of their SOA initiative. It is essential that governance planning begin with the initial SOA deployment, providing the framework, processes and practices for scaling the SOA initiative.

## V. CentraSite: SOA Governance Tool

Software AG and Fujitsu teamed to develop CentraSite, a SOA governance tool and registry/repository that supports the Service lifecycle. CentraSite provides transparency, collaboration, Service utilization, change management and governance to the SOA implementation. It acts as a Web Services and SOA asset management platform, holding all metadata assets, and offering reports on usage. CentraSite incorporates an asset registry, which allows companies to reuse Web Services. CentraSite also increases collaboration between IT and business users by putting access and analysis tools into the hands of business analysts, architects and developers.

*It's possible to leverage CentraSite's open architecture and flexible plug-in architecture which enables integration of various third-party tools.*

It's possible to adopt CentraSite to the specific needs of organizations, leveraging its open architecture and flexible plug-in architecture which enables integration of various third-party tools. CentraSite collects the metadata from components across the SOA infrastructure and assets that these products create, such as Web Services metadata, process models, information models and Web Services orchestration metadata. CentraSite also provides reporting functionality and an impact analysis interface to provide discovery, promote reuse and ensure reliability of SOA assets.

CentraSite provides reduced time to market and cost by increasing the reuse of existing Services and reducing the development of duplicate functionality. As an example, a large insurance firm leveraged CentraSite to obtain the following return on investment. This organization employed 100 developers at the average cost of €100K per developer. Through the use of CentraSite, they achieved a 5% headcount reduction, resulting in €500K savings per year. Furthermore, they

also managed a 10% reduction of help desk calls, leading to €50K savings per year. They also report a reduction of runtime problems, which in turn led to savings of €100K per year through the elimination of one position.

Reducing employee expense was only one area of improvement, however. They also improved documentation and transparency of distributed systems, improved their change management processes through the use of sophisticated impact analysis of interdependencies, and increased quality and operational excellence.

### Beyond Web Services

Software AG fully realizes that SOA governance is more than governing Web Services. That aspect of SOA governance that focuses on governing Web Services in part involves enforcing the contracts that apply to the Services. Yet, WSDL is only one small part of the metadata necessary for adequately defining a Service; we also need information on security, process, quality of service, commercial requirements, and other information.

Software AG's CentraSite SOA registry/repository provides the logical hub where customers can store Service-related components and policy assertions, providing the logical foundation for applying SOA governance. The Service registry references metadata that apply to Services, enabling the critical abstraction layer that separates the Service from its implementation. Likewise, the Repository stores assets associated with the Service.

CentraSite thus combines the capabilities of a registry and a repository, storing all the assets that are relevant to the Service, going beyond models, mappings, shared keys, and transformational schemas, with a plug-in architecture that can accept data such as process models, business rules, test plans and results, and runtime performance histories from external sources.

Further capabilities of CentraSite include:

- Versioning capabilities for artifacts stored in the registry/repository
- Full SOA lifecycle management for Services, policies, processes and other artifacts with customizable lifecycle states and stages
- Standards-based capabilities for policy and security management, auditing logs and configuration management
- Integration with existing infrastructure through an extensible and pluggable AJAX-based user interface architecture accessed through the Eclipse framework

When properly implemented, SOA becomes closely intertwined with business rules, processes, enterprise architecture, IT and corporate governance, which drive the numerous touch points across different management activities across the enterprise. Consequently, a robust SOA governance solution like CentraSite must be extensible, and should support industry standards to integrate with third party tools and solutions addressing governance across the entire Service lifecycle.

## VI. The ZapThink Take: The Business Benefits of SOA Governance

Governance is essential for the success of SOA initiatives. In fact, SOA Governance yields benefits for the enterprise well beyond IT. By properly implementing SOA Governance, companies can control and monitor Service development, use, management, and reuse. SOA Governance prevents Service

"inflation" with its associated high maintenance costs. SOA Governance also optimizes the manageability of the SOA infrastructure.

As organizations achieve levels of success with SOA, they are coming to realize that SOA should apply to the enterprise as a whole. Enabling the business to leverage IT capabilities in flexible, governed ways is a fundamental requirement of the business. As such, IT is an enabler of Service Orientation, but SOA in essence is more than an IT initiative—it is a business initiative.

Governance, as well, is more of a business imperative than a simple IT project. And yet, the enterprise calls upon IT to provide tools for governance, and IT is now calling upon the architects to leverage SOA for better governance. For this reason, SOA governance is at the eye of the SOA storm in most enterprises.

SOA governance is clearly critical for realizing the promise of SOA, and capitalizing on the agility, efficiency that Service Orientation enables. Furthermore, SOA governance helps to resolve the conflict between user empowerment and IT management control. Before SOA, IT management sought to maintain control, and doled out limited capabilities to users, because if they provided too much in the way of user capabilities, the users might violate any number of important policies.

The Service-Oriented approach allows IT to empower a wide range of users to meet the needs of the business in myriad ways, because now we have formalized the practice of SOA governance, which enables business user empowerment in the context of policy-based control. In a fundamental way, therefore, governance is the key to the success of SOA.

## Copyright, Trademark Notice, and Statement of Opinion



## About ZapThink, LLC

ZapThink is an IT advisory and analysis firm that provides trusted advice and critical insight into the architectural and organizational changes brought about by the movement to XML, Web Services, and Service Orientation. We provide our three target audiences of IT vendors, service providers and end-users a clear roadmap for standards-based, loosely coupled distributed computing – a vision of IT meeting the needs of the agile business.

ZapThink helps its customers in three ways: by helping companies understand IT products and services in the context of Service-Oriented Architecture (SOA) and the vision of Service Orientation, by providing guidance into emerging best practices for Web Services and SOA adoption, and by bringing together all our audiences into a network that provides business value and expertise to each member of the network.

ZapThink provides market intelligence to IT vendors and professional services firms that offer XML and Web Services-based products and services in order to help them understand their competitive landscape, plan their product roadmaps, and communicate their value proposition to their customers within the context of Service Orientation.

ZapThink provides guidance and expertise to professional services firms to help them grow and innovate their services as well as promote their capabilities to end-users and vendors looking to grow their businesses.

ZapThink also provides implementation intelligence to IT users who are seeking guidance and clarity into the best practices for planning and implementing SOA, including how to assemble the available products and services into a coherent plan.

ZapThink's senior analysts are widely regarded as the "go to analysts" for XML, Web Services, and SOA by vendors, end-users, and the press. Respected for their candid, insightful opinions, they are in great demand as speakers, and have presented at conferences and industry events around the world. They are among the most quoted industry analysts in the IT industry. ZapThink was founded in November 2000 and is headquartered in Baltimore, Maryland.

**ZAPTHINK CONTACT:**
ZapThink, LLC
108 Woodlawn Road
Baltimore, MD 21210
Phone: +1 (781) 207 0203
Fax: +1 (815) 301 3171
info@zapthink.com

ZapThink, LLC • 108 Woodlawn Road • Baltimore, MD 21210 • info@zapthink.com • www.zapthink.com