SOA Governance:
Balancing Flexibility and Control Within an SOA

September 2006
SOA Governance: Balancing Flexibility and Control Within an SOA
Executive Summary

Few would argue with the fact that SOA is inevitable and has become a strategic imperative for organizations today. Those without a strategy for SOA risk being outpaced and outperformed by competitors who are better equipped to serve customers, seize opportunities and respond to change. But SOA brings new challenges with respect to consistency, predictability and trust. This white paper introduces the concept of SOA governance and provides a framework for blending the flexibility of an SOA with the control, consistency and predictability of traditional IT architecture. Key areas the paper highlights include:

- The crucial elements of SOA governance, including a context for understanding and a vocabulary for discussing associated issues and requirements;
- The role SOA governance plays in corporate governance initiatives;
- The fundamental importance of quality, predictability and trust in an SOA;
- The importance of a full lifecycle approach to managing SOA artifacts; and,
- The excessive cost of inaction, and the consequences of an ungoverned SOA.

This paper offers readers a solid understanding of the role of, and the requirements for, SOA governance. Readers will be better prepared to ask the right questions and to define and implement an SOA governance strategy.

The SOA Imperative

There’s a reason why service orientation is among the most strategic and visible trends within the enterprise today. Few other innovations in computing have the transformative potential of service oriented architectures (SOA). The emergence and wide adoption of open standards, in conjunction with the advent of new SOA tools and infrastructure, has turned SOA from a distant hope to a promise well within reach.

In many respects, being service oriented is synonymous with being business oriented. IT systems are created as a set of reusable services that can easily be linked together to conform to changing business requirements. SOA also moves the focus away from the nuances of underlying technologies and toward process definition, visibility and governance. This requires a new approach to how IT is produced, shared and consumed.

SOA Requires a New Way of Thinking

Gartner estimates that by 2007, 80 percent of IT initiatives will be service oriented. This prediction is made largely because standards-based Web services technologies have made service orientation a practical possibility. Web services are not mandated for SOA, but they offer a practical road to making SOA a viable and interoperable option for broad enterprise adoption.

Despite the high profile of service orientation today, many continue to think of SOA as “just a bunch of Web services.” But SOA is far broader in scope and ambition. SOA is a structured, planned approach to the design, deployment and integration of business-oriented services—including, but not limited to, Web services—within and across lines of business and other organizational boundaries. In this context, SOA brings new challenges with respect to the assurance of service quality, consistency, performance, predictability and, perhaps most fundamentally, trust between the providers and consumers of services.
SOA Governance: Balancing Flexibility and Control Within an SOA

Why SOA Governance?

The promise of SOA is powerful. But what is apparent as organizations peel back the layers is that SOA radically changes traditional IT architectures. While SOA promises untold opportunities, it also introduces new issues around IT governance. The reality is, without a governance strategy, SOA can lead to chaos.

SOA introduces many independent and self-contained moving parts—components which are reused widely across the enterprise and are a vital part of mission-critical business processes. What happens when a service is changed? How can you be sure the service you are consuming is of high quality? What happens if a subcomponent of a composite service is retired? How can you ensure a predictable uptime of a service? These questions illustrate the need for SOA governance. SOA governance is about managing the quality, consistency, predictability, change and interdependencies of services. It’s about blending the flexibility of service orientation with the control of traditional IT architectures.

SOA Governance is All About Trust

A significant challenge to widespread SOA adoption is that while the management of service quality is paramount, simply having quality is not enough. For the first time, quality must be proven and demonstrable to consumers to gain their trust and create an effective shared-service environment.

A useful way to think about the importance of trust in SOA is to consider the example of a consumer marketplace such as eBay, where anonymous buyers and sellers are expected to come together and quickly establish some degree of trust, despite their total anonymity. According to basic economics, a market requires information in order to function efficiently. Information is the lifeblood of any market, largely because it enables buyers and sellers to make informed decisions, and provides the basis for establishing trust. Buyers and sellers on eBay trade on the basis of information. Buyers are not willing to do business unless they understand what is being offered, the terms and conditions of the sale, and the reputation of the seller; likewise, sellers want some assurances of the buyer’s ability and willingness to pay in a timely fashion.

In this respect, SOA is no different. SOA cannot be successful without trust—consumers will simply fail to reuse services if they can’t be assured of quality, predictability and transparency of terms and conditions. In the same fashion, organizations should not encourage the use of services without understanding and controlling access, provisioning, and understanding the overall fitness of reusable services.

SOA and the Nature of Change

Tightly-coupled systems define governance and control in the context of the application. SOA is different, in the sense that the application context is varied and ever-changing. This means that governance must be managed at a different level of abstraction—on the services themselves. This means that policies need to be taken out of the code and externalized as metadata associated with the services. Complicating matters is the fact that, in a loosely-coupled world, change is a constant. Loosely-coupled architectures potentially involve hundreds of services, which evolve and change based on their own unique lifecycles. With all of this change happening at once, how can an IT organization identify and manage the potential impact and interdependencies of change? This is a key domain of SOA governance.

Key Elements of SOA Governance

Full SOA governance can’t be delivered out of the box by a single technology vendor. Rather, it requires a cohesive strategy involving multiple elements that collectively ensure the quality, predictability and trust necessary for reuse. These elements include:

Who Cares About SOA Governance Anyway?

If you consider the priorities on a CEO’s agenda, SOA governance probably doesn’t make the list—at least not explicitly. But it’s important to recognize that SOA governance is a key enabler of corporate governance initiatives, which is a top concern of C-level executives. Sarbanes-Oxley, for example, requires enterprises to establish internal control guidelines and processes, auditor attestations of such controls, and senior management certification of financial results. Corporate governance, in turn, drives IT governance. To implement IT governance, companies turn to frameworks like ITIL, which prescribes setting IT objectives, providing direction to IT operations and comparing implementation to plan. IT activities are then introduced, and there is a process to measure performance.

SOA governance not only offers the means to control business services, but naturally bridges corporate and IT governance. Since organizations construct business services expressly to align IT with the business, they can use standards-based metadata (policies and descriptions) about these services to determine which services need to conform to corporate policies. They can then construct an SOA model that expresses how business services are constructed from elements of IT and enable workflow that drives the management and enforcement of corporate policy.

While corporate policy may be enforced manually using written guidelines, with the right infrastructure in place, SOA governance can automate corporate policy, reducing both risk and cost of compliance.
SOA Policies

The nature of SOA (highly distributed, heterogeneous and very dynamic) means that it is critical for SOA artifacts to be governed by specific business, technical and regulatory policies. An SOA policy defines configurable rules and conditions that affect services during both design time and run time. This means that policies must be used to validate services before they are published, and as a basis for enforcing specific standards and behaviors at run time.

Because an SOA is composed of lots of moving parts, it’s critical that service rules are electronically codified as a set of standard, reusable policies that can be associated with services. Such a linkage between service and policy enables automated validation of services and the enforcement of specific policies. Organizations must make an initial investment in taking these policies out of their dusty hard-bound binders and turning them into electronic business rules. This enables organizations to automate the process of validating and enforcing compliance in both a design-time and run-time environment.

The goal is to first focus policy management at the design-time phase to ensure that quality issues and non-conformance are detected before services are put into production. This means that problems are headed off early, which is less costly to correct and less disruptive to operations than dealing with issues in a production setting. Many organizations will also implement run-time policy management capabilities for monitoring and automatically enforcing policies during the usage of services.

The basic requirements for SOA policies are:

- **Policy Management.** Definition of reusable policies is one of the most important parts of a policy-driven SOA implementation. To define and maintain reusable policies, organizations need a system of record for policies.

- **Policy Association.** Policies are associated with their subjects (often, a business service) in an SOA registry. Policies are published to the registry in the same way as business services or XML schemas. Once policies are published, they are associated with business services, rendering specific capabilities, configurations or requirements that are imposed on specific services.

- **Policy Enforcement.** Policy enforcement is performed by specialized SOA services. Some policies might be enforced by the SOA service repository (usually, design-time policies), some by Web services management (WSM) products (for example, monitoring, logging and SLAs) and some by SOA applications themselves.

- **Policy Reporting.** Information about policies and policy enforcement is summarized in reports. These reports are stored and maintained in an SOA repository.

Service Contracts

Contracts are key architectural tools for communicating and enforcing policies, as well as other requirements in a heterogeneous and distributed IT environment. Just as a business contract ensures a healthy commercial relationship, a service contract ensures a healthy provider/consumer relationship, and helps to establish an agreement and maintain trust between these parties. In other words, a service contract should provide a precise and unambiguous agreement for how the provider and consumer interact. Contracts are typically unique to a specific provider/consumer relationship, and they act as the container for both formal policies, as well as agreements that are unique to the parties.

A useful way to understand the role of a service contract is to relate it to a common example. Virtually everyone has rented a car. In this context, the rental agency is the provider, the renter is the consumer and the car is the service. A contract details the provider (the rental agency) and the consumer (the renter) and specifies the service (the car), the terms and conditions (the policies), and any other provisions or agreements that are unique to the provider and consumer (for example, pre-paying for fuel). This contract is the basis for an agreement to bind the deal. A service contract is no different in complexity or purpose.
Because they're unique to each individual provider/consumer relationship, contracts are typically created at the point of service consumption. But this is not to say that they must be rewritten each and every time. Many contracts can and should be retained and reused to form the basis of many provider/consumer agreements. Therefore, contracts represent another important SOA artifact that should be managed for reuse.

**Lifecycle Management**

The business value of SOA is directly related to quality, predictability and, most fundamentally, trust. But the only way to achieve the promise of SOA is by managing services and other SOA artifacts, not in isolation, but across a complete lifecycle. In this sense, the management of the SOA lifecycle is an intrinsic part of SOA governance. In general, SOA lifecycle management is about:

- Ensuring the quality, performance and applicability of services that are published;
- Providing a means for consumers to discover and reuse services and other artifacts;
- Managing versions, security and state-change of services and other artifacts; and,
- Assessing and managing the impact of change across a network of consumers.

Because of the loosely-coupled nature of providers and consumers within an SOA, there are actually two parallel, but distinct lifecycles at work within SOA:

- The lifecycle of individual services as they are designed, built and deployed (which is primarily the concern of the service provider)
- The lifecycle of a network of services (in which services are accessed and used by changing populations of service consumers, and where the lifecycle primarily concerns those consumers).

The figure above illustrates a lifecycle model for service providers, and a lifecycle model for service consumers. In general, the provider lifecycle is centered on:

- Understanding and managing the requirements for services;
- Managing the access and visibility of services;
- Publishing information to support the reuse of services; and,
- Managing an infrastructure to deliver on quality of service commitments.
The consumer lifecycle is actually quite different. It involves:

- Exploring service availability and capabilities;
- Validating the conformance of services;
- Negotiating terms of usage with providers;
- Validating and reporting on quality of service; and,
- Discovering and responding to changes in services that are consumed.

Proper SOA governance is dependent on a strategy that addresses the requirements for both the provider and the consumer lifecycles. Such a strategy offers the structure, control and discipline necessary to encourage good behaviors and discourage bad behaviors. It’s a common mistake to treat the requirements of providers and consumers similarly, but the reality is that their needs are quite unique.

**Metadata**

SOA governance is ultimately the combination of policy, process and metadata. Metadata, or data about data, is the set of policies and descriptions of business services that enables discovery and appropriate usage of those services. A rich set of information about business services must be interrelated to support all of the governance and lifecycle processes, such as publication, validation and approvals, that is required to ensure that an SOA remains manageable.

Generally speaking, there are three types of metadata: business information, technical information and governance information. Business information includes information like service type (e.g., order entry) and line of business focus (e.g., retail banking). Technical information includes transport type, authentication, interfaces and implementation. Finally, examples of governance information includes the various policies and agreements discussed previously, and the relationships and dependencies between SOA elements.

In a tightly-coupled world, metadata is typically defined within the code of systems and applications. SOA requires this metadata to be externalized—separated from the native system—to enable the classification and governance of these independent services. Thus, metadata becomes a key artifact that needs to be managed within an SOA.

**The Cost of an Ungoverned SOA**

As previously mentioned, an ungoverned SOA can lead to unintended consequences, reversing the virtuous cycle and actually causing SOA to add cost and disrupt processes. The key here isn’t to forego SOA because of this risk; rather, it’s to define a strategy for SOA that builds governance into its core. Costs of an ungoverned SOA can include:

- Lack of reuse by compromising trust and causing consumers to decide against reusing services because of unpredictable quality and performance issues;
- Process disruption by publishing services that don’t fully conform to service-level requirements, or by failure to assess the impact of change;
- Escalations in support cost through an onslaught of help-desk and field service calls resulting from service issues and outages;
- Lack of interoperability creating silos of business services and perpetuating the same challenges of a traditional, tightly-coupled architecture;
- Non-compliance with regulations, by failing to associate key policies with services that have implications for industry or governmental regulations;
- Security breaches by allowing arbitrary access to data and services; and,
- Overall SOA failure by allowing chaos to reign and perpetuating a “garbage in, garbage out” environment.
Governance is Fundamental to SOA

Governance is not the same as management. Rather, it is more accurately defined as “the rules, processes and practices that affect the way in which powers are exercised.” In other words, governance can be best thought of as management architecture. SOA governance is really about creating a management architecture that blends the flexibility of SOA with the control and predictability of a traditional IT architecture.

It’s a mistake for organizations to discount governance as something that is optional, nice to have, or a later-phase aspect of SOA. Governance must begin with the initial SOA deployment, providing the framework, processes and practices for scaling out a healthy and efficient SOA. An organization can’t simply back into governance down the road, once an SOA implementation has reached a new level of maturity. In the unique context of SOA, governance doesn’t follow success; governance begets success.

SOA governance must focus on establishing a framework for assuring service quality and engendering trust between service providers and consumers as both individual services—and the service network as a whole—progress through their lifecycles. Without strategies or infrastructure for governance in place, organizations will hit roadblocks as they try to advance their SOA initiatives. By focusing on maximizing quality and trust within a service network, SOA governance allows organizations to achieve the potential system flexibility promised by SOA with a consistent and managed approach that helps ensure long-term success.

The basic equation for thinking about the value of SOA governance is:

\[
\text{SOA + governance} = \text{flexibility + consistency + trust}
\]

Systinet 2 for SOA Governance and Lifecycle Management

Systinet provides the foundation for SOA governance and lifecycle management, making IT simpler, faster and standards-based. With its suite of award-winning and proven products, Systinet enables organizations to leverage and reuse their existing applications and data assets rapidly, provide interoperability among heterogeneous systems, and better align business processes with IT.

Systinet brings visibility, trust and control to service orientation with an SOA “system of record” and a rich set of governance and lifecycle management applications. This allows organizations to capitalize on the flexibility of SOA together with the control and predictability of traditional IT systems.

The Systinet 2 product set (formerly codenamed “Blizzard”) prepares organizations to realize the full potential of an SOA, providing the foundation for business agility and service reuse without sacrificing IT control or predictability. Specifically, Systinet 2 enables organizations to achieve:

- **Reuse through trust.** Many SOA initiatives fail because reuse is hampered by a lack of trust in the quality and integrity of shared services. Systinet 2 provides the structure, control and processes to engender consumer/provider trust and to ensure reuse.
- **A platform for SOA adoption.** A methodology for designing, deploying and governing services is critical to a successful transformation to SOA. Systinet 2 provides an open and flexible platform for creating the structure, process and best practices for SOA success.
- **A foundation for scale.** An SOA simply cannot scale without governance at its core. Systinet 2 provides the foundation for a healthy SOA that is able to rapidly show momentum and provide the basis for large-scale implementations and broad reuse.
- **Complete SOA visibility.** Complete up-to-date information is key to ensuring a predictable and trustworthy SOA. Systinet 2 provides an enterprise system of record for access to all of the metadata in an SOA, from policy and contracts to run-time metrics.
- **Flexibility and control.** Traditionally seen as mutually exclusive concepts, Systinet 2 allows organizations to finally take advantage of the flexibility and business agility of an SOA without sacrificing IT quality, control or predictability.
An enterprise-class foundation, Systinet 2 delivers SOA governance and lifecycle capabilities, including the ability to:

- Standardize an approach for SOA adoption
- Publish and discover business services
- Validate conformance of services to specific policies
- Create and manage consumer/provider contracts
- Manage full lifecycle of services and other SOA artifacts
- Report on usage and understanding the impact of change
- Ensure interoperability with the broader infrastructure

Systinet 2 includes an SOA repository for storing metadata and managing relationships, and a suite of SOA governance and lifecycle management applications that can be deployed independently or together as an integrated suite. Specific capabilities include:

- **Standards-based Registry**—Systinet Registry has the widest adoption of any business service registry on the market today. It provides a simple and standards-based way to discover and publish reusable business service. It also integrates with the Systinet 2 repository to provide a complete “system of record” for all SOA information.

- **SOA Repository**—The SOA repository is the foundation for the rich governance applications Systinet 2 delivers. The repository provides a way to capture, catalog and discover all of the metadata, artifacts and relationships at the heart of an enterprise SOA. It also provides capabilities for rich reporting, impact analysis and synchronization with other repositories.

- **Service Catalog**—this capability simplifies process of publishing and discovering services with a straightforward and intuitive application for providers to publish and consumers to discover business services.

- **Policy Management**—This capability transforms design-time validation of services from a manual effort to the click of a button. This takes the time and complexity out of service validation and improves the quality and conformance of reusable services.

- **Contract/Consumer Management**—Promotes consumer/provider trust by facilitating service-level agreements and other terms and conditions that bind the service providers and the consumers who reuse services.

- **Lifecycle Management**—provides control over versioning and state-change of business services from initial introduction to final retirement.

- **Governance Interoperability Framework (GIF)**—As the most widely adopted specification for SOA governance interoperability, GIF allows run time applications and other SOA infrastructure to contribute to and reference Systinet 2 as the system of record for SOA information.

Systinet products are based on industry standards such as XML, SOAP, WSDL and UDDI. A pioneer in SOA technology, Systinet led the development of important standards at the World Wide Web Consortium (W3C), OASIS and elsewhere, while remaining consistently first-to-market with advanced and innovative products based on these standards. More than 170 Global 2000 companies rely on Systinet technology, including Amazon.com, BMC Software, Interwoven, JPMorgan, Motorola, Defense Information Systems Agency, and Société Générale.

To find out how Systinet can help your business, visit [www.systinet.com](http://www.systinet.com), or call 1.781.362.1300. E-mail us at [sales@systinet.com](mailto:sales@systinet.com).