

SOA Governance:
What's Required To Govern And
Manage A Service-Oriented
Architecture

An Oracle White Paper
October 2006

SOA Governance: What's Required to Govern and Manage a Service-Oriented Architecture.

INTRODUCTION

Service-oriented architectures (SOA) offer significant advantages, however they place additional demands on visibility, control, and overall governance.

Although enterprise SOA initiatives are generally deployed incrementally, issues around SOA governance must be addressed early in the implementation process to gain long-term value and maintain quality and consistency.

Typically, SOA governance ensures that (1) services deliver expected results based on well-defined business goals, and (2) services are published and managed throughout their lifecycle according to company rules for design, deployment, access control, and audit, as well as compliance with regulatory requirements.

Oracle provides an integrated SOA solution including multiple components addressing the following:

- Lifecycle management of services and related resources
- Definition and management of security policies
- Runtime security policy and service-level agreement enforcement
- Service availability
- Service provisioning
- Monitoring of security and management events

EVOLVING REQUIREMENTS

SOA governance may be viewed as management architecture: a framework that blends the flexibility of SOA with the control and predictability of a traditional IT architecture.

Early ungoverned web services deployments met with multiple adverse consequences:

- A disruption in processes resulting from publishing services that don't fully conform to service-level requirements.

An ungoverned SOA can become a liability for the enterprise, reversing the positive cycle and adding costs and disrupting processes. In fact, Gartner estimates that a lack of working governance mechanisms in mid-to-large-size (greater than 50 services) SOA projects is the most common reason for project failure.

Gartner, "Service-Oriented Architecture Craves Governance", January 2006

- A lack of interoperability amongst web services, creating silos of business services and perpetuating the same challenges presented by traditional, tightly-coupled architectures.
- Non-compliance with company and government regulations due to the failure to associate key policies with services.
- Security breaches caused by allowing arbitrary access to services.

Governance in current and future SOA deployments requires centralized definition of web services security policies and service-level agreements (SLAs). Security and management policies should be defined and stored centrally in a single point of control, and enforced locally where the web services actually run.

SOA governance policies must be provisioned to local enforcement points from a policy management system leveraging a web services registry.

STANDARDS

A key component of a successful SOA governance implementation is a registry based on the Universal Description, Discovery and Integration (UDDI) specification.

A UDDI registry is essentially an online directory enabling service providers to advertise their offerings and allowing service consumers to find services that match their criteria. It provides a “white pages” listing of service providers, a “yellow pages” listing of the services offered, and technical information needed to access a service as defined in the Web Services Description Language (WSDL) document for that service.

In addition, Web Services Inspection Language (WSIL) provides an XML framework designed to inspect a specific site for available services and a set of rules for how inspection-related information should be made available to web services consumers. In this sense, WSIL can be seen as a "metacatalog" of services.

Unlike UDDI, WSIL works under the assumption that you are already aware of the service provider. WSIL facilitates the aggregation of references to various service description documents, which allows users to inspect sites for available web services. For example, users who choose to register their service descriptions in UDDI can create WSIL references instructing consumers to use UDDI to discover service description information.

Another set of standards relevant to SOA governance is Web Services Policy (WS-Policy) and its companion specifications (WS-PolicyAssertions and WS-PolicyAttachment).

A web service provider may define conditions (or “policies”) under which a service is to be provided. The WS-Policy framework enables one to specify policy information that can be processed by web services applications.

A policy is expressed as one or more policy assertions. WS-Policy defines a policy as a collection of policy "alternatives." In this context, a policy alternative is a set of policy assertions used by the application processing the WS-Policy message.

A policy assertion represents a capability or a requirement. For example, a policy assertion may stipulate that a request to a web service be encrypted. Likewise, a policy assertion can define the maximum message size that a web service can accept. The meaning of each assertion is specific to a particular domain, for example, security or privacy.

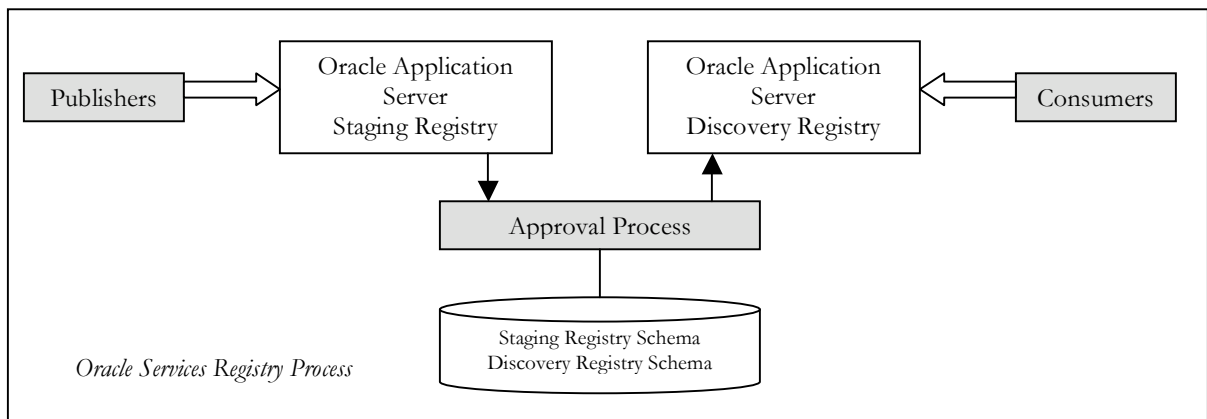
Each policy assertion is identified by its Qualified Name. General policy assertions are defined in the WS-PolicyAssertions specification, which provides an initial set of general assertions for WS-Policy.

WS-Policy doesn't specify how policies are discovered or how they are attached to a web service. WS-Policy expressions are associated with various web services components using the WS-PolicyAttachment specification.

While WS-Policy defines a model for expressing web services policies, WS-PolicyAttachment defines two mechanisms for associating policies with policy subjects:

- Allowing XML descriptions of resources to associate a policy as part of their definition.
- Allowing policies to be associated with arbitrary policy subjects independently from their definition.

WS-PolicyAttachment also defines how these mechanisms may be used to reference policies from WSDL definitions, associate policies with deployed web services, and associate policies with UDDI entities.



ORACLE'S PRODUCT STRATEGY

As noted earlier, Oracle addresses the multiple needs of SOA implementation and management with its SOA Suite, an integrated set of products.

A key component of Oracle's offering is the fully UDDI V3-compliant Oracle Service Registry. The Registry provides a central platform for such critical governance functions as lifecycle management of services and resources; ensuring quality and external and internal standards compliance; notifying stakeholders of change; adherence to policy; and access control to services.

Compliance with the UDDI standard is critical to offering a scalable governance solution. For example, UDDI compliance enables integrated development environments such as Oracle JDeveloper and Eclipse to query the Oracle Service Registry in a standard manner, enabling such tools to retrieve service interfaces for client generation.

In addition, the Registry itself is implemented as a standards-based web service, meaning that virtually all of the Registry's functionality can be accessed through standard SOAP-based application programming interface (API) calls. As such, any number of client applications can be implemented or extended to interact with the Registry at design-time or runtime.

In addition to the Registry, the Oracle SOA governance solution includes Oracle Web Service Manager (WSM), which provides the ability to enforce SOA policy at runtime for each web service registered with the environment.

Oracle WSM can be used by a developer or a security administrator. The Oracle WSM process includes:

- Defining security policies and SLAs based on corporate and regulatory requirements using Oracle WSM's Policy Manager.
- Broadcasting security policies and SLAs to enforcement points which can be gateways (analogous to proxy servers sitting in the DMZ) or agents (running in the same process space as the application they protect). Gateways and Agents execute security policies and SLAs in real time.
- Monitoring access-control events such as authentication and authorization, as well as web service requests traffic and pre-defined SLAs. Monitoring is reported in charts and other graphical representations in Oracle WSM's web-based management console.

The Oracle Service Registry integrates with Oracle WSM to enable runtime enforcement of policies set on services exposed through the Registry. Specifically, the Oracle WSM administrator can import service information published to the Registry and used by Oracle WSM to secure access to web services through its policy enforcement points (gateways and agents described above). Once a service is secured, the UDDI entry for the service within the Registry can easily be updated with the proxy URL redirecting service callers through Oracle WSM.

CONCLUSION

The promise of SOA is powerful and appealing but it radically changes traditional IT architecture approaches.

While SOA provides many benefits, it also introduces new challenges and risks that must be managed and mitigated.

The concept of SOA governance, while still nascent, is already a prerequisite for a successful SOA implementation. As with any sound management practice, SOA must be seen as a first-order concern, with requirements that must be factored into your organization's SOA strategy at the very earliest stages of implementation.



SOA Governance: What's Required to Govern and Manage a SOA.

October 2006

Author: Marc Chanliau

Contributing Author: Dan Hynes

Oracle Corporation

World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065

U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

oracle.com

Copyright © 2006, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.