SOA Policy Enforcement: Overview

This topic provides an overview of SOAtest's quality policy enforcement capabilities.

Sections include:

Policy Enforcement Details

SOAtest provides a complete SOA policy enforcement solution, enforcing policies with executable rules that can be applied to WSDLs, schemas, SOAP messages, and any other XML artifact or SOA meta-data component.

Once an organization has defined their policies to guide their SOA deployments, SOAtest can be used to enforce them throughout the development and QA process. For example, SOAtest verifies schema and semantic validity for W3C and OASIS standards compliance, validates Basic Profile 1.1, 1.2, or 2.0 for WS-I Interoperability compliance, and implements rules to enforce various other endorsed WS* Standards.In addition, SOAtest can be used to enforce compliance to best practices such as customized company guidelines, security, and maintainability and reusability.

Registry-Based Policy Management

SOAtest provides native support for multiple commercial registries. This integration enables teams to automatically execute a quality workflow and correlate quality data in the context of an SOA Governance initiative. Teams can automatically extract the information needed to create tests for design and development policies (such as standards, compliance, security, and best practices) for web services assets as they are defined in a registry. They can also select a service asset and verify the associated policies, thereby ensuring interoperability and consistency.

SOAtest is capable of querying any UDDI registry from vendors such as IBM, HP, and Microsoft. Furthermore, Parasoft offers even tighter integration with Oracle / BEA's AquaLogic Enterprise Repository (ALER) and Software AG's CentraSite. We automatically generate tests at the time the services are published to the registry–including functional test cases and WSDL verification tests that ensure WSDLs are compliant to best practices and organizational policies. Policy compliance results are then reported back to the registry and updated in real-time. This provides continuous visibility into a service's quality throughout its lifecycle.

Registry-Based Test Generation

- To learn how to create tests that enforce policies applied to Web service assets that are declared in a BEA repository, see Creating Tests From Oracle Enterprise Repository / BEA AquaLogic Repository.
- To learn how to create tests that enforce policies applied to Web service assets that are declared in a Software AG CentraSite repository, see Creating Tests From Software AG CentraSite Active SOA.

WSDL, Schema, and Semantic Verification

WSDL verification can be considered as the first step in testing web services. Although WSDLs are generally created automatically by various tools, it doesn't necessarily mean that the WSDLs are correct. When WSDLs are manually altered, WSDL verification becomes even more important. Ensuring correct and compliant WSDLs enables your service consumers to function correctly, and avoids vendor lock-in, thus achieving interoperability and realizing SOA goals of service reuse.

SOAtest can automatically generate a test suite of comprehensive WSDL tests to ensure that your WSDL conforms to the schema and passes XML validation tests. Additionally, it performs an interoperability check to verify that your Web service is WS-I compliant.

WS-* Standards Validation

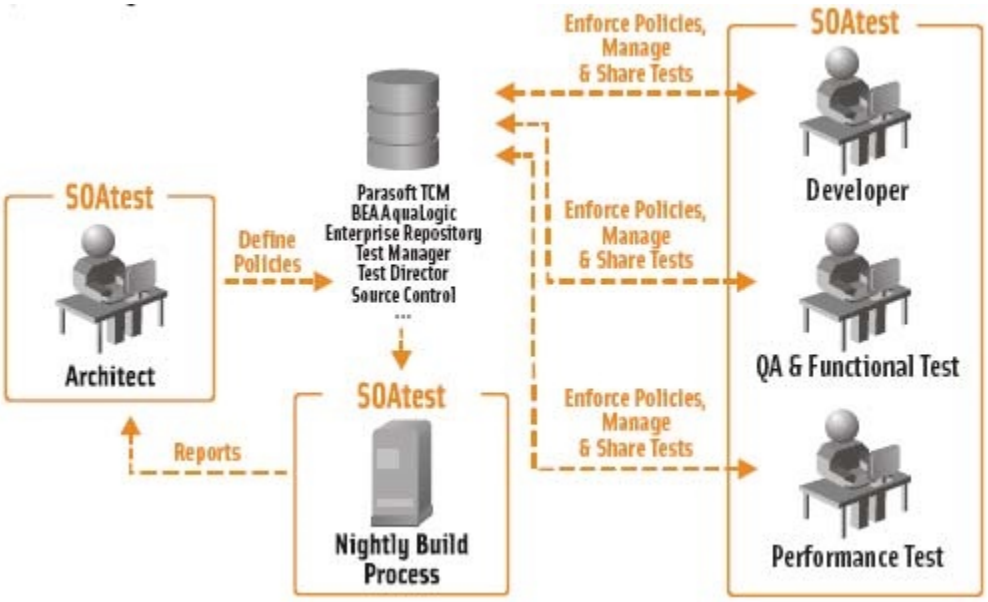
SOAtest enforces policies with executable rules that can be applied to WSDLs, schemas, SOAP messages and any other XML artifact or SOA meta-data component. For example, we verify schema and semantic validity for W3C and OASIS standards compliance, validate Basic Profile 2.0, 1.2, or 1.1 for WS-I Interoperability compliance, and implement rules to enforce various other endorsed WS* Standards. In addition, we enforce compliance to best practices such as customized company guidelines, security, and maintainability and reusability.

Interoperability Testing

SOAtest verifies the WSDL and SOAP traffic for conformance to Basic Profile 2.0, 1.2, or 1.1 using WS-I Testing Tools. Functioning as both a traffic monitor and analyzer, SOAtest enhances the usability of WS-I Testing Tools by eliminating the need to set up man-in-the-middle Monitor and configuration files for the Analyzer. The only required input is the WSDL URL.

Recommended Workflow

SOAtest's policy enforcement component enables the SOA architect to define policy rules. Using SOAtest, Web service developers, QA, and Test engineers can then verify service compliance against the Architect-defined rules early in the Web service lifecycle process. The Architect then monitors compliance to these rules, resulting in a visible and controlled development process.



In addition, the rules defined by the architect can be applied across the entire organization via Parasoft's Team Server, a server component that enables the sharing of rules and test policies among Parasoft's products.

Tutorial

For a step-by-step tutorial of how to monitor compliance to SOA policies, see Design and Development Policy Enforcement.