E-Guide

Testing Web Services: What You Need to Know
The first thing to get straight, according to ebizQ columnist Andre Yee, is that Web Services testing is decidedly different from conventional testing.

Here, Yee explains how, and then serves up a number of best practices for testing Web services.

**Testing Web Services: What You Need To Know**  
By Andre Yee

Perhaps it’s just our preoccupation with new development, but it never ceases to amaze me how little attention is paid to the entire application lifecycle. This seems especially true whenever a new technology is introduced and assimilated. Consider Web services, for example.

There are countless books and seminars, primarily targeted to the developer, focused on the design and coding of Web services and service oriented architectures (SOA). Yet, precious little information is given to the latter phases of the application lifecycle, such as the testing, deployment and management of Web services-based architectures. In this column, we will tackle the unique challenges surrounding just one of these phases - the testing of Web services, particularly as it is applied in an inter-enterprise SOA model.

**Is Web Services Testing Different?**

Before delving into Web services testing concepts, let’s explore the unique aspects of Web services that present a challenge to traditional testing models.

There are three key aspects to Web services used in SOAs that are worthy of mention.

First, a Web services-based architecture results in a distributed, loosely coupled system with dependent stakeholders.
While this architecture holds great benefit in terms of scalability and extensibility of the system, it poses a great challenge for the tester. No longer is each component of the system necessarily constructed and deployed by a single corporate entity. Potentially, the overall system may actually be derived from components that are constructed, tested and hosted by multiple stakeholders. As such, it is difficult to establish consistent code quality, availability and quality of service (QoS) across these entities.

The second difference in the Web services solution is that it provides a standards-based, service-driven model to integration. While prior object technologies such as CORBA provided facilities for building distributed architectures, they did so via complex programmatic interfaces. With Web services, standards for data format (XML), communication (SOAP) and programmatic interface (WSDL) are data-driven, simple and share a common XML-based foundation. Traditional testing tools must evolve to fully support these standards for effective testing to occur, but that hasn’t really happened yet.

The use of Web services standards also offers other implications to current testing tools. Web services-based integration highlights the importance of validating interface points and message formats rather than simply testing at the graphical user interface level. Using testing tools that utilize GUI-driven automation is simply inadequate for a Web services project.

Finally, the applicability of Web services in an inter-enterprise SOA model introduces new dimensions of functional testing. For instance, traditional, tightly-coupled architectures are often tested from a functional blackbox approach. However, in an inter-enterprise SOA model, it is essential to validate the efficacy of the “publish”, “find” and “bind” Web services relationship shown in Figure 1.
In addition, validation of end-to-end transactions is significantly more difficult in the SOA system due to the difficulty in tracking message routes through intermediaries. Yet, without the ability to monitor and validate effectively, one would be unable to tune, debug and optimize the system.

**Best Practices for Testing Web Services**

Based on our discussion about the differences in Web services testing, it is reasonable to assume that current testing methodologies should be extended to support the use of Web services in an SOA solution. Here are a few key practices to consider as part of this extended model.

**Practice #1 – Proactively Establish Service Level Agreements**

The inter-enterprise SOA system spans multiple stakeholders and intermediaries. Hence, it is important to establish service levels such as high availability metrics, Web service response times and security policies upfront, during the requirements phase. Once established, these service levels should be incorporated into test plans. Doing so proactively is not only politically expedient when dealing with
multiple stakeholders, but also less costly from a development standpoint.

**Practice #2 – Perform Component-Level Validation and Benchmarking**

Establishing the service level agreements upfront facilitates the ability to perform component-level testing. Instead of benchmarking the service level of the overall system when integrated, strive to perform testing of the individual Web service components before integration to ensure they meet acceptable parameters.

**Practice #3 - Stress Testing is Essential**

When dealing with a distributed system with multiple intermediaries, it is important to perform stress/load testing in a pre-deployment mode. Testing should be performed for steady stream throughput as well as burst mode traffic.

**Practice #4 – Invest in Web Services Monitoring Tools**

The ability to monitor message processing is fundamental to testing Web services. Monitoring SOAP message routing, error handling, and service response times are all part of the reason you want to invest in Web services monitoring tools. Examples of tools you may want to check out include products from Confluent, AmberPpoint and Empirix.

**Practice #5 – Extend Test Scenarios to Incorporate Web Services Functionality**

Test cases need to be extended to support the Web services paradigm and its accompanying functionality. This includes testing for:

- SOAP message format validation
  
  Recommended testing tool: SOAPTest from Parasoft
Contents

Is web services testing different?

Best practices for testing web services

About the author

- The publish, bind and find functionality
- Service location independence
- Validate interfaces as defined in WSDL
- Recommended testing tool: FirstACT from Empirix
- Testing for interoperability between SOAP platforms.
  Recommended testing tool: Web Service Monitor and Web Service Profile Analyzer from WS-I group.

About the Author

Andre Yee is an entrepreneur and technologist with nearly 20 years of experience in the business of technology. He has authored two books and several articles on enterprise technologies. He is currently SVP, Products for Eloqua, the category defining leader in marketing automation platform. Prior to Eloqua, he was the CEO, NFR Security, a leading intrusion prevention company which was successfully sold to Checkpoint Software. He is also the holder of two patents was noted as one of the Innovators to Watch, 2006 by Info World magazine.
Free resources for technology professionals

TechTarget publishes targeted technology media that address your need for information and resources for researching products, developing strategy and making cost-effective purchase decisions. Our network of technology-specific Web sites gives you access to industry experts, independent content and analysis and the Web’s largest library of vendor-provided white papers, webcasts, podcasts, videos, virtual trade shows, research reports and more—drawing on the rich R&D resources of technology providers to address market trends, challenges and solutions. Our live events and virtual seminars give you access to vendor neutral, expert commentary and advice on the issues and challenges you face daily. Our social community IT Knowledge Exchange allows you to share real world information in real time with peers and experts.

What makes TechTarget unique?

TechTarget is squarely focused on the enterprise IT space. Our team of editors and network of industry experts provide the richest, most relevant content to IT professionals and management. We leverage the immediacy of the Web, the networking and face-to-face opportunities of events and virtual events, and the ability to interact with peers—all to create compelling and actionable information for enterprise IT professionals across all industries and markets.

Related TechTarget Websites

- ebizQ
- SearchSOA
- SearchSoftwareQuality
- SearchContentManagement