

**PARASOFT.**

# SOAtest™

## DATA SHEET

### Parasoft SOAtest is the industry's premier testing platform for service-oriented architectures and composite applications.

Parasoft SOAtest helps QA teams ensure secure, reliable, compliant business applications with an intuitive interface to create, maintain and execute end-to-end testing scenarios. It was built from the ground up to reduce the complexities inherent in complex, distributed applications.

Since 2002, Parasoft customers such as HP, IBM, Fidelity, Lockheed Martin, and the IRS have relied on SOAtest for:

- Ensuring the reliability, security, and compliance of SOA, cloud, and web applications
- Reducing the time and effort required to construct and maintain automated tests
- Automatically and continuously validating complex business scenarios
- Facilitating testing in incomplete and/or evolving environments
- Validating performance and functionality expectations under load
- Rapidly diagnosing problems directly from the test environment



## Virtualize

### Service Virtualization with Parasoft Virtualize

Parasoft Virtualize, which is seamlessly integrated with Parasoft SOAtest, helps teams rapidly access any environment needed to develop, test, or validate an application. It dramatically reduces the time and cost of managing dev/test environments by emulating the behavior of dependent systems, which may be unavailable, evolving, or difficult-to-access.

### End-to-End Testing

<b>End-to-End Test Scenarios</b>	Promotes a building-block approach for rapid development of test suites that exercise multiple endpoints, which may span across the messaging layer, ESBs, the web interface, the database, and mainframes. This ensures the reliability of the underlying implementation.
<b>SOA-Aware Test Construction</b>	Advanced test automation and an SOA-Aware interface enable fast construction of extensible tests. Automatically generates tests from artifacts such as WSDL, WADL, UDDI, WSIL, XML, Schema, BPEL, HTTP traffic, and key industry platforms (see list on page 2).
<b>Web Interface Validation</b>	Guides the team in developing robust, noiseless regression tests for rich and highly-dynamic web interfaces—without scripts. Supports cross-browser testing, RIA/AJAX testing, and multi-level validation.
<b>Continuous Regression Testing</b>	Continuous regression tests are applied across the multiple layers of heterogeneous systems. These tests will immediately alert you when modifications impact application behavior—reducing the risk of change.
<b>Runtime Error Detection</b>	Automatically exposes defects that occur as the application is exercised—including race conditions, exceptions, resource & memory leaks, and security attack vulnerabilities. Pinpoints the root causes of application crashes, performance degradation, and unpredictable behavior.

### Event Tracing and Validation

<b>Message/Event Monitoring</b>	Visualizes and traces how messages and events flow through ESBs, message brokers, applications, databases, and more as tests execute—facilitating rapid diagnosis of problems directly from the test environment. Regression controls and validations can be applied at any point in the process.
<b>Test Flow Visualization</b>	Constructs a logical representation of the test execution sequence from SOAtest as well as remote systems under test. This aggregates all relevant information in a single location—with intuitive controls for zeroing in on the information needed to debug complex test scenarios (e.g., with many tests, multiple data sources, dynamic data, etc.).

Load Testing	
<b>Performance and Stress Testing</b>	Existing functional tests are leveraged for load testing, removing key barriers to performance monitoring. Monitors response rates as well as verifies whether functionality problems occur under load. Also supports non-Parasoft components such as JUnits or lightweight socket-based components.
<b>Expected Quality of Service Management</b>	Defines and enforces expected QoS metrics that are important for setting and measuring SLAs during development and QA. You can define success metrics before testing and set performance thresholds. "Regression performance tests" can run as part of the automated continuous testing process.
Security Testing	
<b>Penetration Testing</b>	Automatically generates tests to perform security penetration testing at the message layer and web interface. Covers parameter fuzzing, SQL injections, XPath injections, various XML issues, XSS, buffer overflow, command injection, unvalidated input, and more.
<b>Message Layer Security Policy Validation</b>	Enables execution of complex authentication, encryption, and access control test scenarios. Supports key transport and message-level standards (SSL, OAuth, Digest, Kerberos, WS-Security, etc.). Allows for token validation and negative testing to ensure proper enforcement of message integrity and authentication.
Quality Governance	
<b>Design and Development Policy Enforcement</b>	Ensures interoperability, security, and consistency across distributed systems—from application code statements to business processes. Industry-standard and custom policies are enforced automatically and continuously.
<b>Registry-Based Policy Management</b>	Automatically tests registered services and verifies compliance to the policies defined in the registry. Results are reported back to the registry—and updated in real-time—providing continuous visibility into a service's quality and compliance throughout its lifecycle.

**Platforms**

■ AmberPoint ■ HP-QC ■ IBM/Rational ■ Microsoft ■  
 ■ Oracle/BEA ■ Software AG/webMethods ■  
 ■ Progress Sonic ■ Tibco  
 Windows ■ Linux ■ Solaris ■ Mac

**Technologies and Protocols**

SOA / Web services ■ Java ■ .NET languages ■ XML ■ WSDL ■ WADL ■ UDDI ■ WSIL ■ SOAP ■  
 PoX (Plain XML) ■ REST ■ JSON ■ BPEL ■ Web Applications ■ Mobile Web Interfaces ■ RIA ■  
 AJAX ■ JSP ■ JavaScript ■ HTML ■ CSS ■ WS-\* Standards ■ MTOM(XOP) / MIME / DIME  
 Attachments ■ Oauth ■ TCP/IP ■ webMethods Broker ■ webMethods IS ■ H17 ■ FTP ■ HTTP 1.0 ■  
 ■ HTTP 1.1 ■ HTTPS ■ JMS ■ IBM WebSphere MQ ■ Sonic MQ ■ RMI ■ EJB ■ SMTP ■ Tibco  
 Rendezvous ■ .NET WCF (TCP, HTTP, WS Transaction Flow) ■ ISO 8583 ■ EDI ■ custom/other

**Parasoft SOAtest Delivers Real Results**

Here are just a few examples of how leading organizations use Parasoft SOAtest...



To audit business processes



To enforce SOA policies



To validate complex business scenarios



To create and manage emulated service assets



To validate performance expectations



Vanguard®

To ensure expected quality of service



To ensure the reliability of iPhone billing systems



To validate and load test corporate tax e-filing systems



To validate role-based specimen management systems



To ensure the reliability and performance of the world's largest travel network  
 (including Travelocity, American Airlines, and more)

**www.parasoft.com**

**Contact info:**

Parasoft Corporation, 101 E. Huntington Dr., 2nd Flr., Monrovia, CA 91016  
 Ph: (888) 305.0041, Fax: (626) 256.6884, Email: [info@parasoft.com](mailto:info@parasoft.com)