Introduction

Parasoft SOAtest

Automates thorough testing for composite applications with robust support for REST and web services, including over 120 protocols /message types.

Parasoft Virtualize

Simulates the behavior of systems that are still evolving, hard to access, or difficult to configure for development or testing.

SOAtest + Virtualize

Together, SOAtest and Virtualize enable you to lower the cost and accelerate the delivery of better software.

Why SOAtest?

Parasoft SOAtest is an enterprise-grade solution that simplifies complex testing for business-critical transactions through APIs, message brokers, databases mainframes, ERPs, browser-based UIs, and other endpoints. 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 testing APIs and API-driven transactions within complex environments.

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

- Ensuring the transaction reliability, security, and compliance
- 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

Why Virtualize?

Parasoft Virtualize enables access to realistic test environments– removing significant barriers to delivering quality software efficiently. Virtualize provides access to the dependencies that are beyond your control, still evolving, or too complex to configure in a virtual test lab. This may include third-party services (credit check, payment processing, etc.), mainframes and SAP or other ERPs. You can virtualize only a fraction of an entire system's available functionality. As you naturally exercise the application under test, Parasoft captures interactions with dependencies and converts this behavior into flexible “virtual assets” with easily-configurable response parameters (e.g., performance, test data and response logic). Sophisticated virtual assets can be created and provisioned for role-based access in a matter of minutes.

With Parasoft Virtualize, you and your team can:

- Start testing whenever you're ready.
- Rapidly configure the environment conditions critical to your test plan.
- Complete the desired breadth and volume of tests.
- Confidently promote the application under test to the next level.

End-to-end Functional Testing

Rapidly constructs test scenarios that continuously validate all critical aspects of complex transactions—which may extend through web interfaces, backend services, ESBs, databases, and everything in between.For more details, see End-to-End Test Scenarios, Web Functional Tests, REST and SOA Functional Tests.

Security Testing

Prevents security vulnerabilities through penetration testing and execution of complex authentication, encryption, and access control test scenarios.For more details, see Security Testing.

Change Management

Alert appropriate team members about changes that impact their testing, enabling fast, intelligent updating of assets through a centralized change template.

Load/Performance Testing

Verifies application performance and functionality under heavy load. Existing end-to-end functional tests are leveraged for load testing, removing the barrier to comprehensive and continuous performance monitoring. Support is also provided for load testing non-Parasoft components such as JUnits or lightweight socket-based components, and for detecting concurrency issues.

For more details, see Load Test.

Message/Protocol Testing

Automates the testing of multiple messaging and transport protocols– including HTTP, SOAP/REST, PoX, WCF, JMS, TIBCO, MQ, EJB, JDBC, RMI, and so on.

For more details, see Testing Through Different Protocols.

Event Monitoring

During test execution, you can visualize and trace the intra-process events triggered by tests, facilitating rapid diagnosis of problems directly from the test environment. You can also continuously validate whether critical events continue to satisfy functional expectations as the system evolves.

For more details, see Event Monitoring - ESBs, Java Apps, Databases, and other Systems.