

# Introduction

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

## Getting Started with SOAtest

Tips for getting started with SOAtest:

- New SOAtest Users: We recommend starting at [SOAtest Tutorial](#).
- Users migrating from SOAtest 5.x or WebKing: We recommend starting at [Migrating from SOAtest and WebKing](#) then proceeding to [SOAtest Tutorial](#).

## 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](#).