



CTP

## Establish a Platform for Continuous Testing

- » **Customize your test environment.** Provision real and virtual services, inject realistic test data, configure performance characteristics, and execute the right test cases.
- » **Orchestrate dependencies.** Switch between real and multiple virtual states within a single test environment.
- » **Collaborate on test execution.** Ensure the end-to-end test flow has the correct data sources and endpoints to enable seamless cross-environment execution.
- » **Coordinate microservices for test coverage.** Collect end-to-end code coverage across a large number of microservices associated with functional tests.

"All the configuration and deployment of virtual assets for the various environments is automated as part of our DevOps infrastructure. Environments automatically switch between virtual assets and actual assets according to the business rules we've defined."

- Frank Jennings, Director



# Parasoft Continuous Testing Platform (CTP)

## Collaborate on Continuous Quality Across Application Test Scenarios

### Improve Quality With Continuous Testing

Today's agile, "continuous everything," security, and DevOps initiatives require the ability to test software efficiently and effectively to assess the risks associated with market delivery.

Parasoft's continuous testing solution helps organizations rapidly and precisely validate that their applications satisfy their market and business requirements around functionality, reliability, performance, and security. Both technical and non-technical team members can configure and provision a complete test environment, and immediately execute tests against the appropriate test environment configurations.

### Parasoft CTP Combines Many Capabilities Into One Collaborative Interface

#### Environment Health Checks

Access a holistic view of the functional testing process at the system level: from environments and components all the way to service endpoints. Parasoft CTP alerts users about issues when processes malfunction, so they can drill down and find the root cause.

#### Environment Diagrams

Clearly diagram interdependencies between real and virtual components in the entire test environment. Outline relationships between the application under test, backend services, and third-party APIs for clearer understanding of environment topology.

#### Virtual Test Data

Capture, manage, share, and repurpose realistic test data, reducing data friction and enabling better testing. View data as a visual model that is

Create test cases, virtual services, and test data right from your browser. These test artifacts can then be combined into powerful environment templates, which can be deployed on-demand using the Environment Manager. These activities all happen in concert with your existing CI/CD infrastructure, allowing you to seamlessly integrate continuous testing into your pipeline.

Collect coverage across interconnected microservices as functional tests execute in order to report the combined coverage of the entire system. Test impact analysis identifies which end-to-end tests need to be rerun when code changes in one of the microservices.

easy to understand, manipulate, mask, and extend through data generation, simplifying the process of creating safe test data.

#### Test Environment Replication

On-demand provisioning of test environments, made up of live and virtual services, enables automated tests to run continuously. Clone and automatically reconfigure all virtual assets connected to a test environment. Parasoft CTP easily replicates your environment into an independent test sandbox.

#### Test Report Archive

Save test execution job reports in the Continuous Testing Platform database so you can easily go back and review pass/fail history with full error messages. Thorough documentation helps with compliance audits.



Access a holistic view of the functional testing process at the system level from environments and components all the way to service endpoints.

## Make Testing More Manageable

### Collaborate and Reuse Testing

Initial API tests can be quickly built, then extended into complex end-to-end functional test scenarios that can also be leveraged for early-stage load and performance testing and security testing.

### Reuse Data Across Tests and Tools

Test data can be captured into a centralized portal and visually represented as a model, allowing users to access, generate, mask, and repurpose it for their test cases and virtual services. This reduces the overall time spent waiting for test data.

### Eliminate Testing Roadblocks

Simulate unavailable or difficult-to-configure dependent systems. Service virtualization assets can be constructed from the same artifacts used for test construction.

### Reproduce Defects Faster

If a test fails, recreate the correlated test environment with the click of a button to reproduce the failing test case within the exact same environment configuration, accelerating the process of defect remediation and eliminating the time-consuming, frustrating plight of development being unable to reproduce problems reported by QA.

### Code Coverage for Microservices

Release with confidence using Parasoft CTP's advanced code coverage capabilities. Gather coverage metrics on your distributed Java or .NET-based microservices during automated functional test runs of Parasoft and non-Parasoft test cases to identify coverage gaps and ensure that untested code doesn't slip into production. Scale code coverage workflows from one monolithic application all the way to hundreds of microservices. Identify end-to-end functional test scenarios that need to

be rerun when individual microservices change using test impact analysis. Parasoft CTP coordinates the coverage collection and reporting process for the system as a whole so that test framework integration does not need to be aware of every component. Easily collect code coverage from concurrent test executions in the same test environment.

### Understand the Impact of Change

Gain insights into the effects of change. Use Parasoft CTP to proactively scan your virtual services and test cases and, when a service changes, to notify you and provide a change template to help you update your test artifacts. Additionally, you can run change impact analysis to understand how a potential change will affect your test artifact repository. Finally, you can run this analysis in the context of your environment to address environmental changes early.

## Monitor and Log Events

Enable monitoring on components in your environment to log traffic events. CTP's Event Viewer provides advanced filtering and the ability to create virtual assets from received messages.

In addition to controlling the monitoring of individual components, you can start and stop all event monitoring on a specific Parasoft Virtualize server. View events such as request messages sent, response messages received, and more.

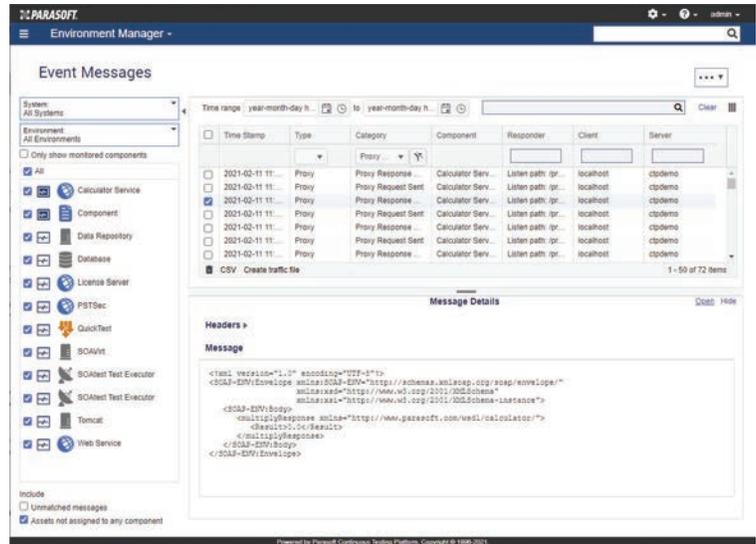
## Coordinate Test Cases

**Artifact repository.** Manage a library of test cases, virtual services, test data, and technology extensions.

**Sequence responders.** Configure pre-defined responses for repeated requests to simplify complex service behavior.

**Test execution.** Execute preconfigured jobs automatically as a part of your CI/CD pipeline.

**Environment manager.** Unite your test cases and virtual services to create an environment that can be executed continuously.



The event viewer provides centralized visibility into the entire CTP ecosystem across servers and environments. View event messages associated with a particular system, environment, or set of components.

## Streamline Testing

See how to manage your test environment with Parasoft CTP. [Request a demo.](#)

