About this Tutorial

The topics presented in this tutorial will guide you through how SOAtest addresses key areas of service testing and Web UI testing. This tutorial will demonstrate the creation of various test suites.

Requirements

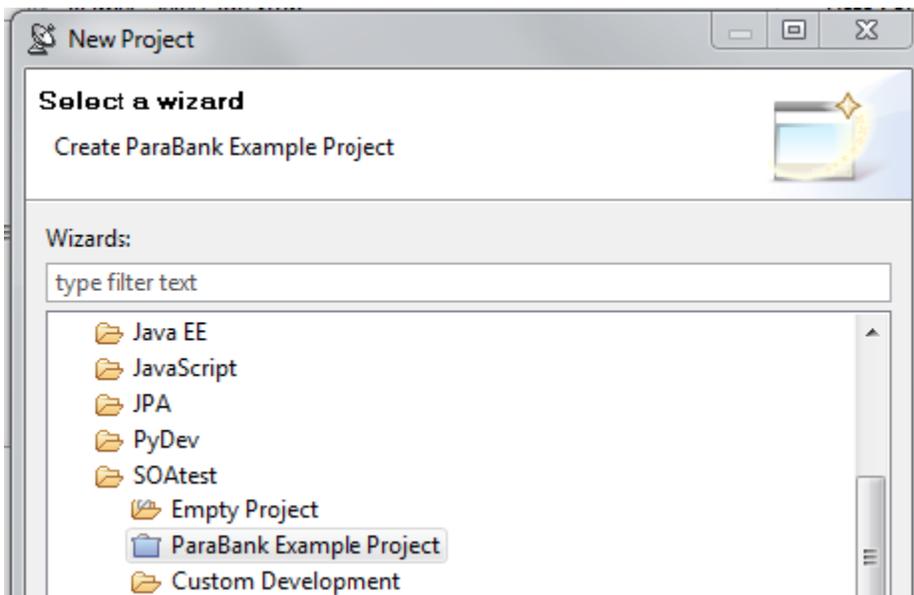Requirements for completing this tutorial include:

- Parasoft ParaBank, a sample web application included in SOAtest.

For your convenience, we've provided a sample SOAtest test suite named SOAtestTutorial.tst. This test suite is located within the SOAtest examples directory. It contains all the tests that will be created through the tutorial, and also includes specific examples referenced throughout this tutorial. Further information about each test is given in the **Requirements and Notes** tab of each test suite's configuration panel.
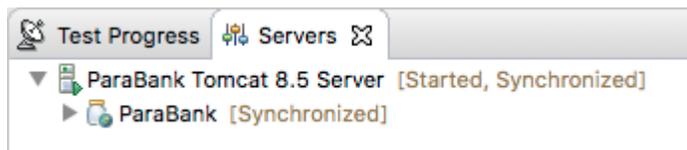
Setting Up ParaBank

To complete most of the examples in this tutorial, you will need to create and run a ParaBank project.

1. In SOAtest, select **File> New> Project**.
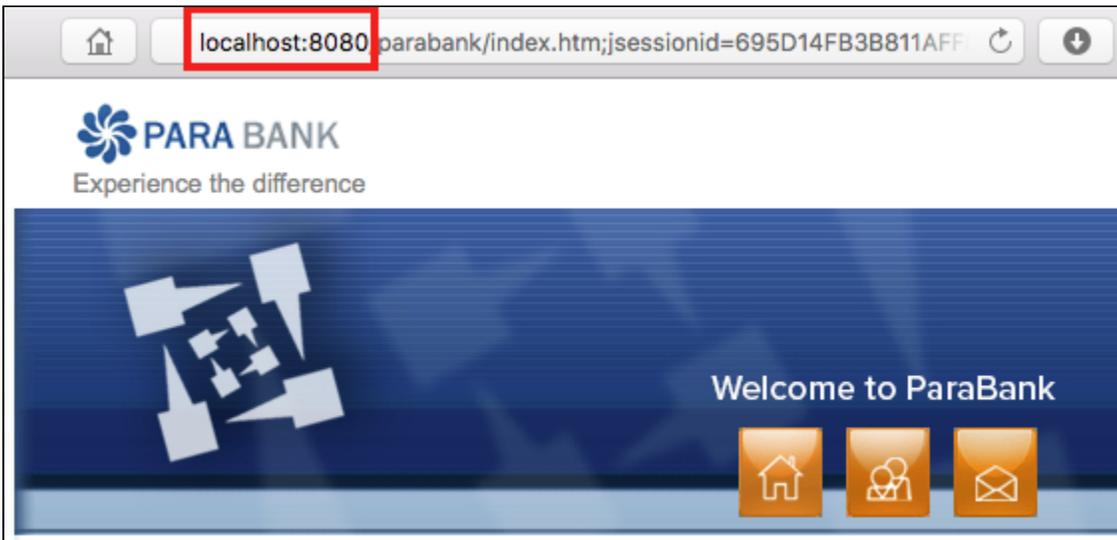2. Select **SOAtest> ParaBank Example Project**, then click **Next**.

3. Enter a project name (e.g., `ParaBank`), then click **Finish**.
4. In the **Servers** view (if it's not visible, choose **Window> Show View> Other> Server> Servers**), verify that the ParaBank Tomcat Server is started and synchronized. It might take about a minute for the server to start and synchronize.

Note the Port Being Used

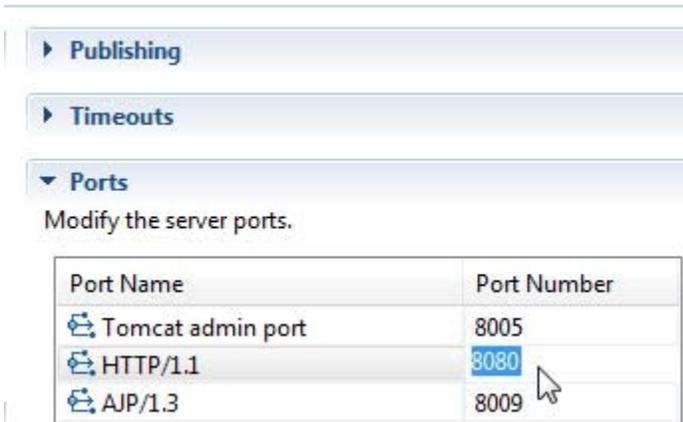When the web application launches, be sure to note what port it is using.

ParaBank uses port 8080 by default. If port 8080 is already in use, SOAtest incrementally searches for an available port, starting at 8000.

Using a Different Port

You can change the port settings as follows:

1. In the **Servers** view (if it's not visible, choose **Window> Show View> Other> Server> Servers**), double-click the **ParaBank Tomcat 5.5 Server** entry. The Overview editor opens.
2. Under **Ports**, change the **HTTP/1.1** entry to a port of your choice.
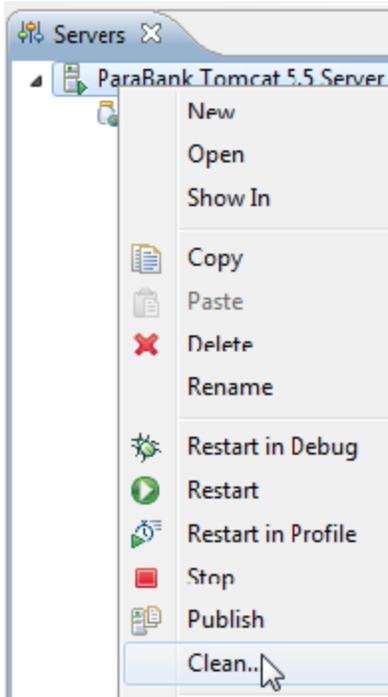

3. Save the editor by pressing **Ctrl+S**.
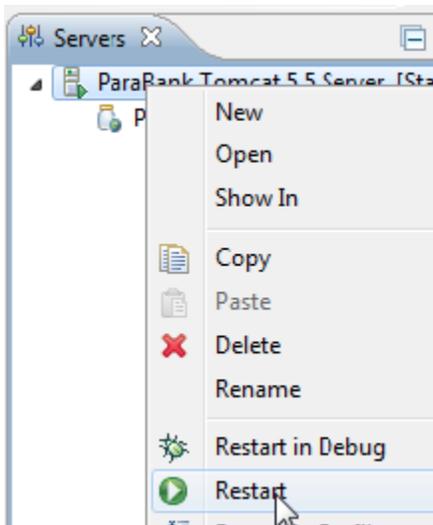4. In the **Servers** view, click the **Restart the server** button. ParaBank will restart.

Troubleshooting ParaBank

If you upgrade from a previous installation of SOAtest and the browser displays a `404 Error` message after the upgrade—even if the Java perspective reports that the server started properly—you need to clean the workspace as follows:

1. Open the Java perspective.
2. Open the Servers view (if it's not visible, choose **Window> Show View> Other> Server> Servers**).
3. In the Servers view, right-click the **ParaBank Tomcat** server, then choose **Clean**.

4. Restart the **ParaBank Tomcat** server (right-click it, then choose **Restart**).

Parasoft SOAtest Best Practices

While reading this document, you will find examples that show the recommended way of creating test cases within SOAtest. You can apply these best practices when creating tests for your own services and applications.

The following list details some core best practices for using SOAtest:

- Using the SOAtest test wizard, create a test suite of tests that are run on a nightly basis.
- Using the SOAtest test wizard, create a test suite of test clients for each operation defined within your WSDL. These test clients can then be moved into separate test suites for Functional Tests and Scenario Tests to optimize reusability and organization.
- Create positive and negative test cases for each test case to fully maximize the testing coverage of the service.
- Create regression tests for both positive and negative test cases. Regression tests alert you to any changes in service functionality as the service evolves over time.
- For each distinct testing requirement, create a separate Test (.tst file).

You will learn how to apply these and other best practices throughout the tutorial.

As you gain a basic understanding of SOAtest functionality, we strongly recommend that you read the Best Practices for SOAtest.