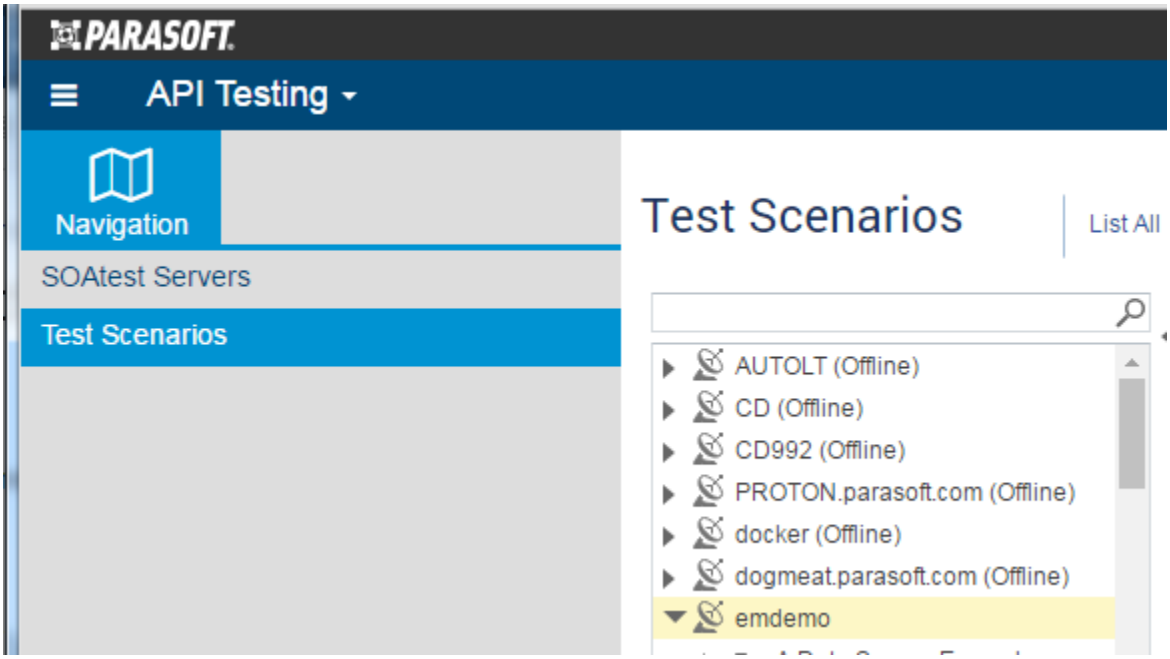
# Creating New Test Scenarios and Test Suites



You can add test scenarios and test suites on SOAtest Server directly from the Test Scenarios page, which is accessed by opening the API Testing module.

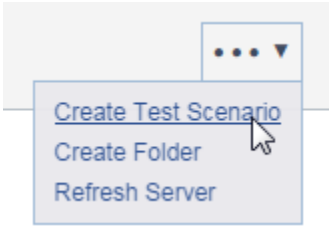



## Creating New Test Scenarios (.tst files)

A test scenario is the smallest unit that can be executed in a job or during provisioning. It is stored as a .tst file within the SOAtest workspace.

To add a new .tst from CTP:

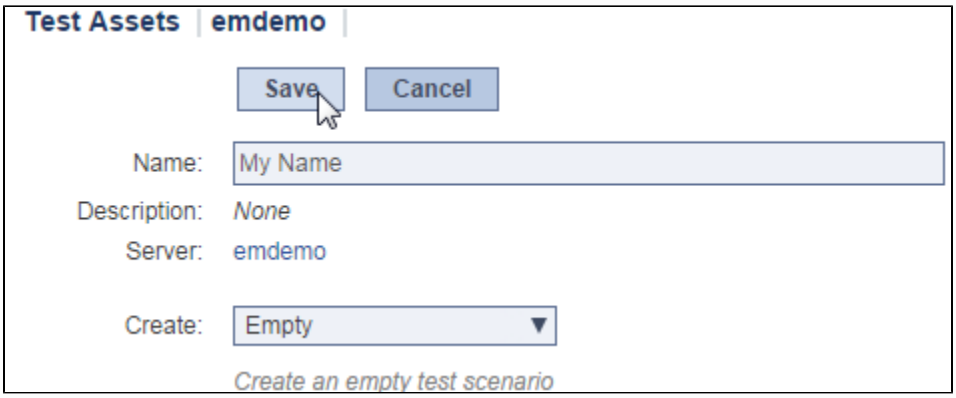
1. In the left pane, select the server or folder where you want the new test scenario added.
2. Choose **Create Test Scenario** from the page-level action menu.




3. (Optional) Modify the name of the newly-created test scenario.
4. Indicate how you want the scenario created.

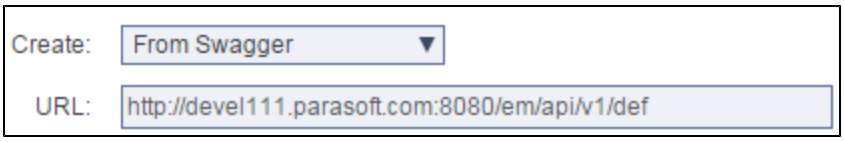
**Test Scenario Creation Options**

**Empty:** To create a skeleton (empty) test scenario, leave **Create** set to **Empty**, then click **Save**. No additional steps are needed here (you can add tests later).

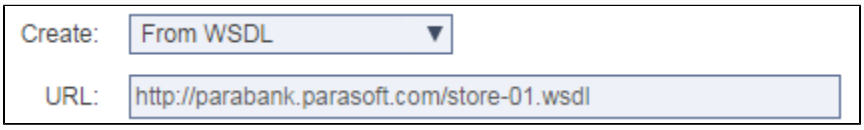



**RAML:** To automatically-generate a test scenario based on the endpoints found in a RAML description, set **Create** to **From RAM**L, enter the absolute URI at which to find the RAML definition, then click **Save**. RAML 0.8 and 1.0 are supported.

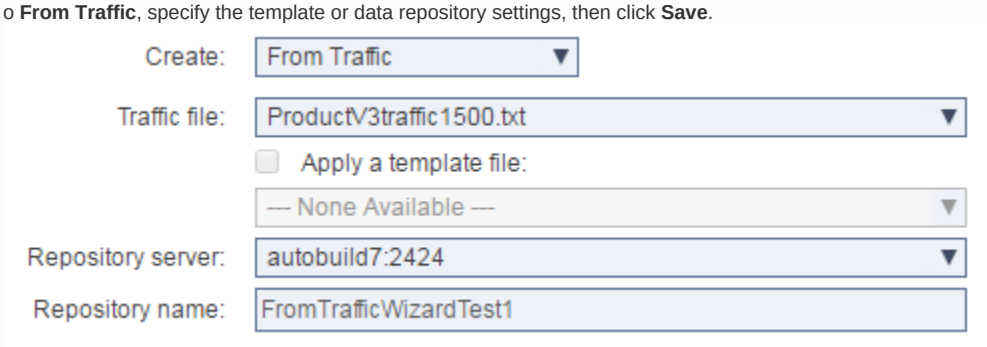

**Swagger:** To automatically-generate a test scenario based on the endpoints found in a Swagger description, set **Create** to **From Swagger**, enter the absolute URI at which to find the Swagger definition, then click **Save**. Swagger Specification 1.0 - 2.0 is supported.




**WSDL:** To automatically-generate a test scenario from a WSDL (one test for each service operation), set **Create** to **From WSDL**, specify the WSDL location, then click **Save**.




**Traffic:** To automatically-generate a test scenario from a traffic file (created from a Parasoft proxy driver recording or another utility), set **Create** t o **From Traffic**, specify the template or data repository settings, then click **Save**.

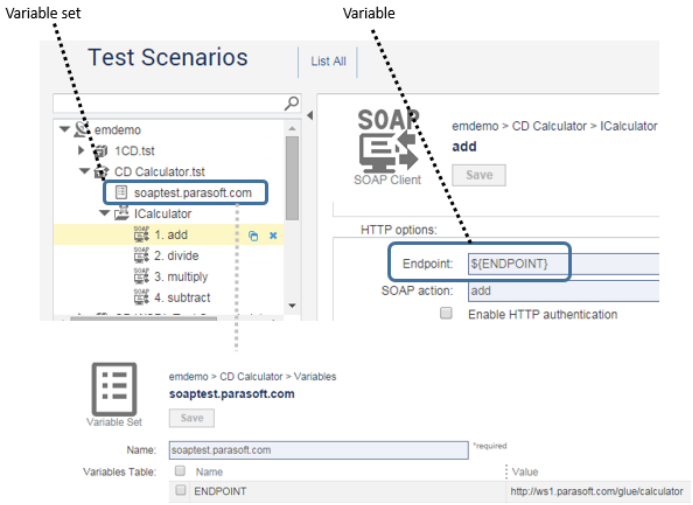



Tips for creating tests from traffic:

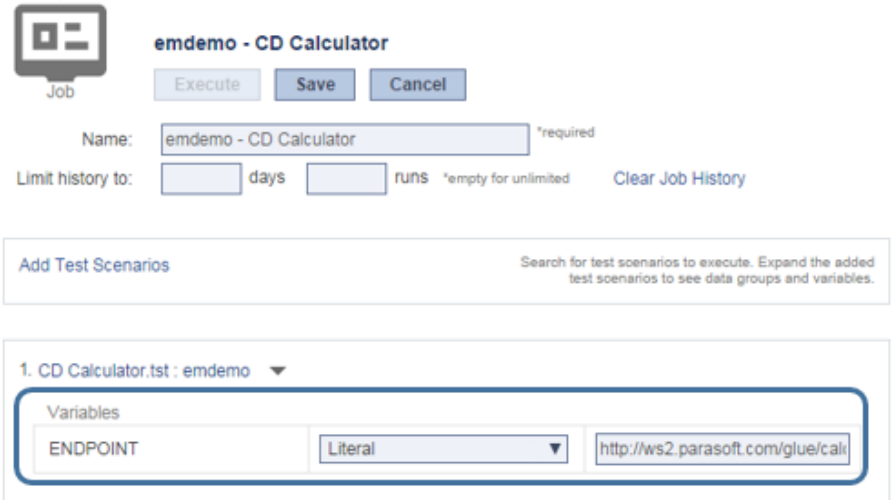
- In order to create a new test scenario from traffic, you need the following: a) Parasoft SOAtest co-installed with Parasoft Virtualize, b) Access to a running Parasoft Data Repository, and c) At least one traffic file (e.g., created by proxy recording or event message export), which must be saved within the VirtualAssets project
- (Optional) If you want to create tests using "create from traffic" configurations that were saved in a template from Parasoft SOAtest, specify which template you want to use (you can select from any templates available within the traffic_templates folder on the given SOAtest server). Once a template is specified, the related fields will be automatically set and become uneditable.
- When specifying the data repository name, you can either use an existing repository or you can specify a new one.
- The traffic's message contents must be well-formed (e.g. if XML, it must be well-formed); otherwise, auto-creation of tests from the traffic might fail. SOAP messages must have only one top-level XML element.

The .tst (included any generated SOAP Client tools) will be added to the specified SOAtest server. You can review the added SOAP Clients by selecting the associated tree node.

- **If tests were created from traffic:** The tools will be automatically configured and parameterized with values stored in the specified Parasoft data repository.
- **If tests were created from RAML or Swagger**: The test scenario will include one REST Client for each resource/method pair in the definition. Each REST Client's resource URL, HTTP method, and payload (if applicable) are configured accordingly. The service's base URL is configured as a "BASEURL" variable. Each resource URL is parameterized with the "BASEURL" variable. Query parameters are included with default or sample values (if available) as defined by the service definition. A sample message (if available) is included in the payload. If the service definition includes a JSON schema, a sample payload is constructed from that JSON schema.
- **If tests were created from a WSDL:** The tool's name, request message, endpoint, and SOAP action will be configured automatically. A variable is created for the endpoint value. With the endpoint set to a flexible variable rather than a hard-coded value, you can run the same test versus different environments without modifying the test itself. Generated variables are defined in the generated variable set.




Variables values can be adjusted at the test job level, as well as the component instance level.

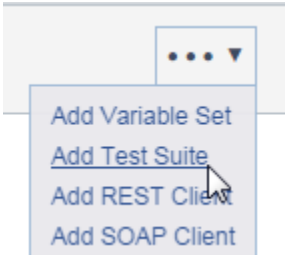



# Creating Skeleton Test Suites

To add a skeleton test suite within an existing test scenario (.tst file):

1. In the left pane, select the .tst or test suite where you want the new test suite added.

2. Choose **Add Test Suite** from the page-level action menu.




3. (Optional) Modify the name of the newly-created test suite.

The test suite will be added to the specified SOAtest server within the designated .tst file.

# Adding Tools

You can add commonly-used API Testing tools (such as REST Client, SOAP Client, Diff, and XML/JSON Assertor, and XML/JSON Data Bank tools) directly from the CTP interface. Additional tools can be added from SOAtest desktop.
For details, see:

- REST Clients
- SOAP Clients
- Assertors
- Data Banks
- Diffs

**Parameterizing Tools**

Tool values can be parameterized using data sources or extracted values. See Parameterizing with Data Source and Data Bank Values for details.