

# API Coverage - Overview

SOAtest's API Coverage provides visibility into how well your tests cover the related resources and operations. This helps you gauge the effectiveness of your existing test suite and determine where additional tests need to be added.

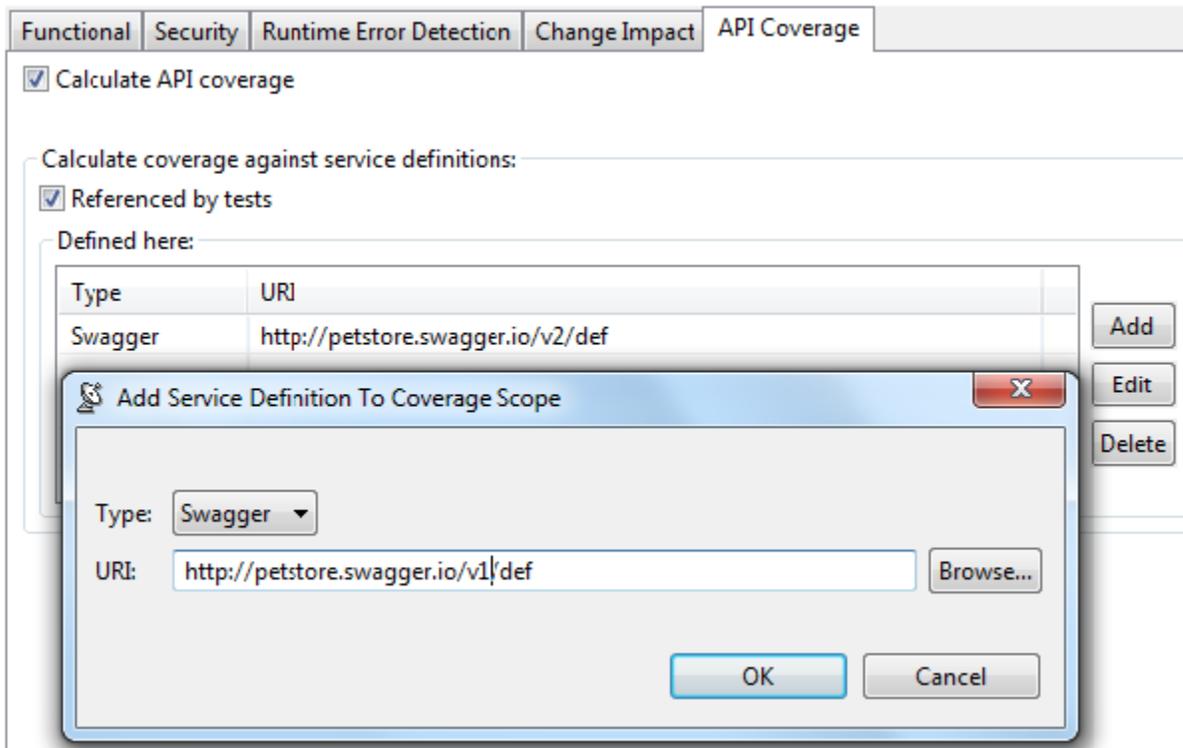
SOAtest's reports detail the level of coverage achieved for each applicable resource and operation, as well as the pass/fail status of the tests responsible for that coverage. The reports also pinpoint which .tsts and tests covered each operation (with a pass/fail breakdown at this level as well).

## Understanding "Coverable Resources"

SOAtest calculates and reports how well your tests cover the resources and operations associated with "coverable resources." If any (constrained or unconstrained) SOAP Client, REST Client, or Messaging Client tool touches a coverable resource, coverage information will be associated with the appropriate resource/operation.

If tests are executed with a Test Configuration whose **Execution > API Coverage > Referenced by tests** option is enabled, SOAtest compiles a list of coverable resources from the service definitions (OpenAPI/Swagger, RAML, WADL, WSDL) that are configured in constrained SOAP Client or REST Client tools.

If you want SOAtest to calculate coverage for a user-defined list of coverable resources in addition to—or instead of—the automatically-calculated coverable resources, you can specify that list in the Test Configuration's **Execution > API Coverage > Defined here** table.



Coverage for any services that *do not match any of the known coverable resources* is calculated and reported under "Unknown Service." Coverage information is not currently calculated for query parameters, payloads, or response codes.

## Example 1: Calculating Coverage for Service Definitions Referenced by Tests

For example, assume you have a test suite with two test clients:

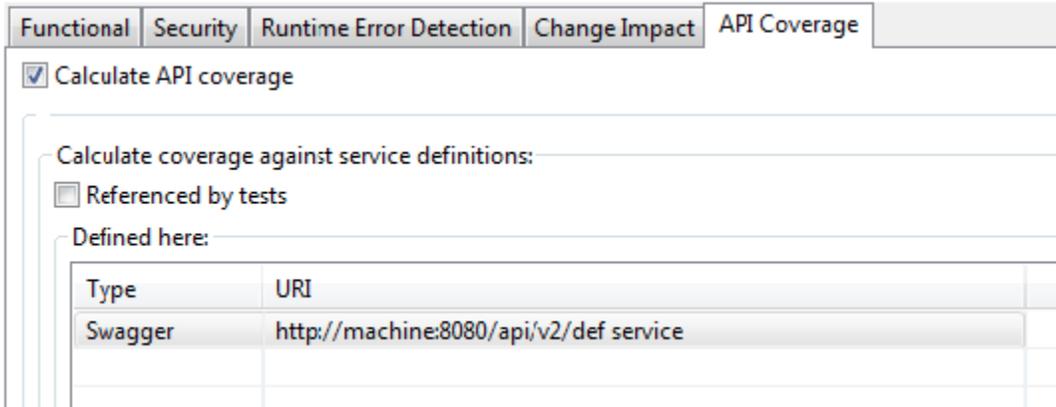
- A REST Client that is constrained to a "Petstore" service definition and exercises the associated Petstore service.
- A Messaging Client that is not constrained, but exercises that same Petstore service.

Also assume that the Test Configuration's **Execution > API Coverage > Referenced by tests** option is enabled. SOAtest will recognize the "Petstore" service definition as a "coverable resource" because it is configured in the constrained REST Client. Since the REST Client and Messaging Client both touch the Petstore service, SOAtest will report both tools' coverage under the Petstore resource and operations.

If you add a new Messaging Client that exercises a service other than Petstore (e.g., Grocerystore), this coverage will be reported under "Unknown Service".

## Example 2: Focusing on a Specific Service Definition

Now, assume that you have a test suite with hundreds of test clients (many of which are constrained to various service definitions), but you want to focus on reviewing coverage related to the API defined at <http://machine:8080/api/v2/def>. To achieve this, you would disable the Test Configuration's **Execution > API Coverage > Referenced by tests** option and set the table as follows:



The screenshot shows a configuration window with several tabs: Functional, Security, Runtime Error Detection, Change Impact, and API Coverage. The 'API Coverage' tab is active. Under this tab, there is a checked checkbox for 'Calculate API coverage'. Below this, there is a section titled 'Calculate coverage against service definitions:' with an unchecked checkbox for 'Referenced by tests'. Underneath, there is a section titled 'Defined here:' followed by a table with two columns: 'Type' and 'URI'. The table contains one row with the value 'Swagger' in the 'Type' column and 'http://machine:8080/api/v2/def service' in the 'URI' column.

Type	URI
Swagger	http://machine:8080/api/v2/def service