Indicating Code and Test Correlations

This topic explains how to associate tests/code with development artifacts.

- Introduction
- About Default Tags
- Creating Associations on the Desktop
- Correlating Load Test Scenarios
- Using Custom Defect/Issue Tracking Tags
- Linking to Correlated Artifacts in Reports

Introduction

You can associate your test/code with development artifacts to help you track the quality of the application under test. Associations are established by adding Javadoc-like tags to the test code and specifying the development artifact ID. SOAtest includes several default tags, but you can create custom tags to meet your needs.

The @req and @test are special tags in SOAtest that enable traceability from the test to the artifact in your application lifecycle or requirements management system (ALM/RMS). Parasoft DTP and the Traceability Pack extension for DTP are also required. Refer to the Working with the Requirements View chapter for details on leveraging traceability with Parasoft.

This chapter covers the basic process of how to apply the association tags.

About Default Tags

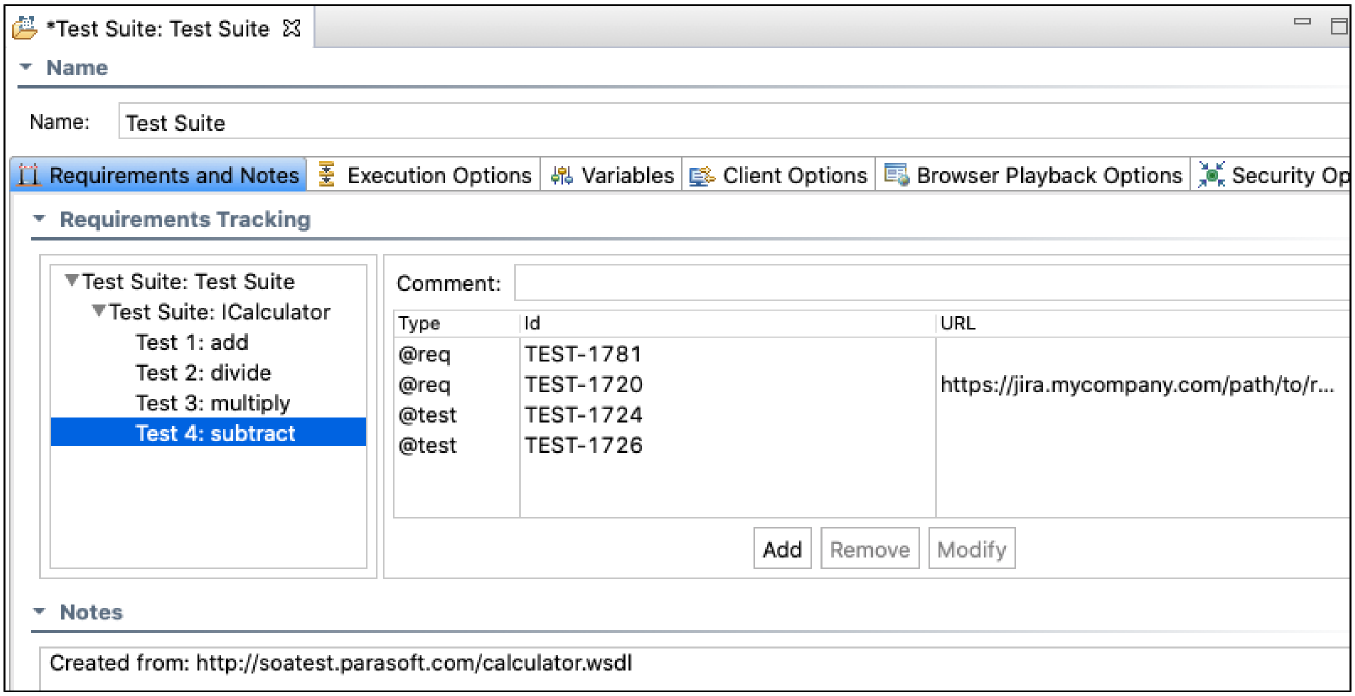
The following tags are supported by default:

- @PR
- @FR
- @TASK
- @REQ - Can be used in the requirements traceability workflow (see Working with the Requirements View)
- @TEST - Can be used in the requirements traceability workflow (see Working with the Requirements View)

See Using Custom Defect/Issue Tracking Tags for details on creating your own tags.

Creating Associations on the Desktop

The **Requirements and Notes** tab of SOAtest’s test suite configuration panel allows you to identify requirements/tasks/bugs associated with the tests in the test suite.



Name: Test Suite

- Test Suite: Test Suite
  - Test Suite: ICalculator
    - Test 1: add
    - Test 2: divide
    - Test 3: multiply
    - Test 4: subtract

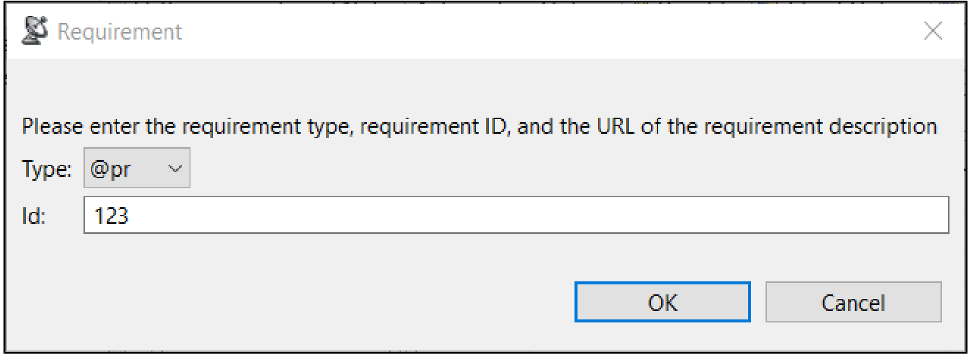
Comment:

Type	Id	URL
@req	TEST-1781	
@req	TEST-1720	https://jira.mycompany.com/path/to/r...
@test	TEST-1724	
@test	TEST-1726	

Created from: http://soatest.parasoft.com/calculator.wsdl

The requirements you define will appear in Structure reports and in DTP Report Center, allowing managers and reviewers to determine whether the specified test requirements were completed.

1. Select a node from the test suite tree within the **Requirements and Notes** tab.
2. Click the **Add** button.
3. In the **Type** box, select a requirement type. DTP will use this information to associate the test suite's test cases to the specified element type. For instance, if it is associated with a specific bug, information about the test case's status will be considered for Report Center's bugs graphs. Custom tags can be added as described in Using Custom Defect/Issue Tracking Tags. Default tags are:
   - @pr for bugs.
   - @fr for feature requests.
   - @req for requirements.
   - @task for tasks.
   - @test for tests
4. Enter a requirement **ID** and click **OK**.



Please enter the requirement type, requirement ID, and the URL of the requirement description

Type: @pr

Id: 123

The requirement you defined will display in the **Requirements** table within the **Requirements and Notes** tab and correspond to the test suite node you selected and all of its child nodes.

5. (Optional) If you want to enter notes for the test suite, enter a description in the **Notes** field. This option is useful in that it enables you to view a quick description of the test suite purpose.

Correlating Load Test Scenarios

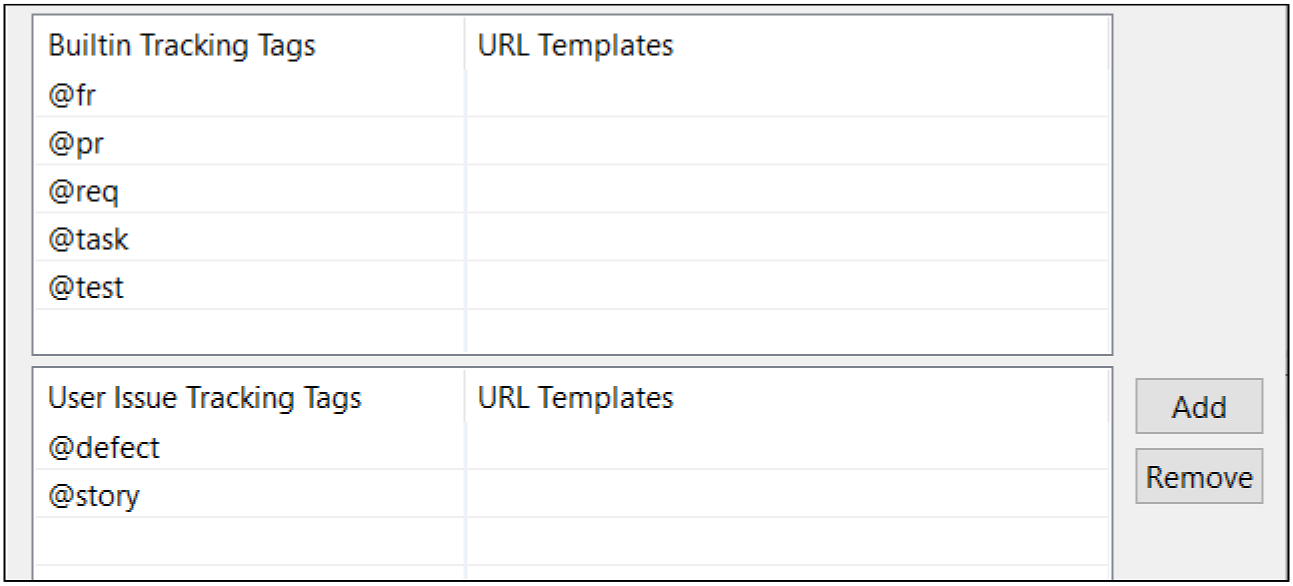
See Reviewing and Customizing Load Test Results.

Using Custom Defect/Issue Tracking Tags

You can customize the tags your team uses to associate a test case with a work item from your ALM, defects, or requirements tracking system (e.g, codeBeamer, Jira, VersionOne, etc.). This allows you to configure the tag to match the language that your organization uses to refer to defects. For instance, some organizations use the term "PR" to refer to a defect. Others use "defect" and would thus prefer to use a custom @defect tag rather than the default @pr tag.

To specify custom tags:

1. Choose **Parasoft> Preferences**.
2. Select **Parasoft> Issue Tracking Tags**.
3. Use the available controls to add, modify, or delete tags.
   - A tag is followed by one or more values that must be represented by either a single string or a number of strings separated by separator tokens.
   - Separator token values can be any permutation of comma-separated lists of strings and white space-separated lists of strings.



Builtin Tracking Tags	URL Templates
@fr	
@pr	
@req	
@task	
@test	

User Issue Tracking Tags	URL Templates
@defect	
@story	

The tags listed here will be available in SOAtest when you configure the Requirements Tracking options in the test suite editor (when you go to **Add** a new requirement correlation). See Correlating SOAtest Tests for details.

Linking to Correlated Artifacts in Reports

If you want SOAtest reports to link to correlated artifacts, you need to specify URL templates that explain how links to these artifacts are formatted. To do this:

1. Choose **Parasoft> Preferences**.
2. Select **Parasoft> Issue Tracking Tags**.
3. In the panel's top table, specify how links to each artifact type are formatted. For example:
    - Mapping @pr to http://bugzilla.company.com/show_bug.cgi?id=${id} means that the URL for PR1024 is http://bugzilla.company.com/show_bug.cgi?id=1024
    - Mapping @task to http://host.company.com:8080/grs/jsf/planning/task/edit_task.jsf?entityId=${id} means that the URL for task 1215 is http://host.company.com:8080/grs/jsf/planning/task/edit_task.jsf?entityId=1215