Using custom fields

Xray provides a set of custom fields that can be used for some issue types, as detailed in Custom Fields and Screen Configuration.

Aside from these, you can add your own custom fields to any Xray issue type: Tests, Pre-Conditions, Test Sets, Test Executions, Test Plans.

Currently, you can't add custom fields to Test Runs because they're not Jira issues.

- Xray-provided custom fields
  - Begin Date
  - Conditions
  - Cucumber Scenario
  - Cucumber Test Type
  - End Date
  - Test Repository Path
  - Generic Test Definition
  - Manual Test Steps
  - Manual Test Steps (Export)
  - Pre-Condition Type
  - Pre-Condition association with a Test
  - Requirement Status
  - Revision
  - Steps Count
  - Test Type
  - TestRunStatus
  - Test Plans associated with a Test
  - Test Sets associated with a Test
  - Tests Count
  - Tests association with a Pre-Condition
  - Tests associated with a Test Plan
  - Tests association with a Test Set
  - Tests association with a Test Execution
  - Test Environments
  - Test Execution Defects
  - Test Execution Status
  - Test Set Status
  - Test Plan Status
- Adding your own custom fields
  - Recommendations

Xray-provided custom fields

As mentioned in Custom Fields and Screen Configuration, whenever you install Xray, it will also add some custom fields that are available only for some issue types. Below are explanations of some of the most important ones:

Begin Date

The start date for the Test Execution or Test Plan.

Conditions

The Pre-Condition conditions.

Cucumber Scenario

The Cucumber Scenario clauses in Gherkin.

Cucumber Test Type

The Cucumber Scenario Type (i.e., Scenario or Scenario Outline).

End Date

The end date for the Test Execution or Test Plan.

Test Repository Path

The Test Repository complete folder path for the Test.

Folders and their respective sub-folders are delimited by "/" (e.g., "components/compA").

The "Test Repository Path" is case-insensitive and each folder is trimmed (i.e., spaces are removed from the start/end of it).

Please note

You can move a Test to an existing folder within the Test Repository by editing the custom field in the Test issue.

Whenever editing this custom field, keep in mind that it is case-insensitive. This means that "components/compA", "components /compA", and " components/COMPA" are all the same and will be mapped to the same folder within the Test Repository.

If the folder is not found, then the Test is not associated with any folder; it will be seen in the "Orphans" meta-folder in the Test Repository UI.

Generic Test Definition

The definition of a generic Test (e.g., full classname of some automated test, name of some script, goals of exploratory testing).

Manual Test Steps

The manual steps fields, in JSON format

Manual Test Steps (Export)

The manual steps fields for exporting manual tests

Pre-Condition Type

Choose the Pre-Condition Type. It must match the Test Type to which this Pre-Condition is associated.

Pre-Condition association with a Test

The Pre-Condition(s) issue(s) associated with this Test.

Requirement Status

Whenever we're speaking about the status associated with a requirement, we may be talking about different things:

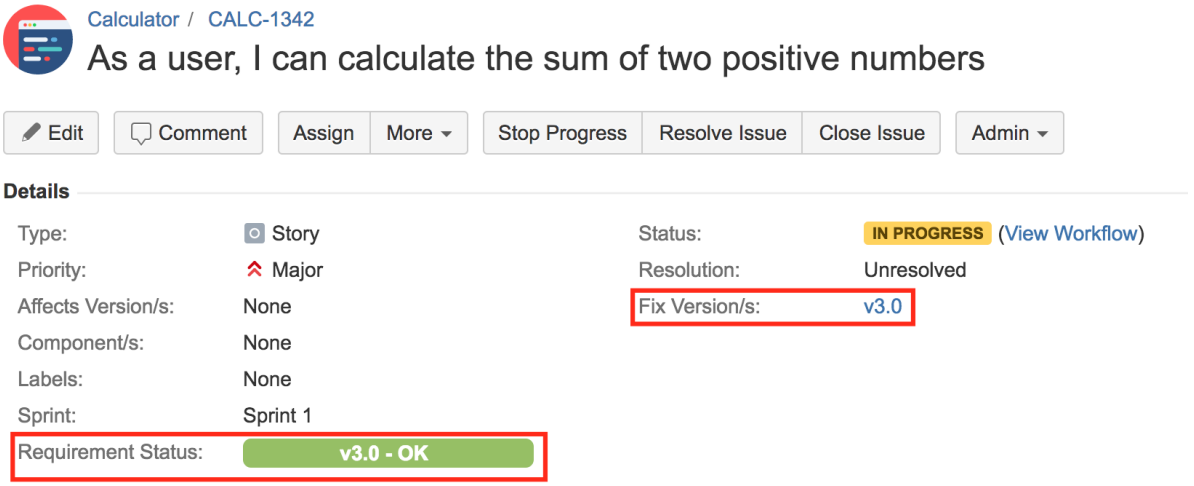
- workflow status associated to the requirement issue (e.g., "New", "In Progress", "Closed")
- (coverage) status for some version, taking into account executions made for that version of the Tests that validate the requirement
- "Requirement Status" custom field, which we will detail next.

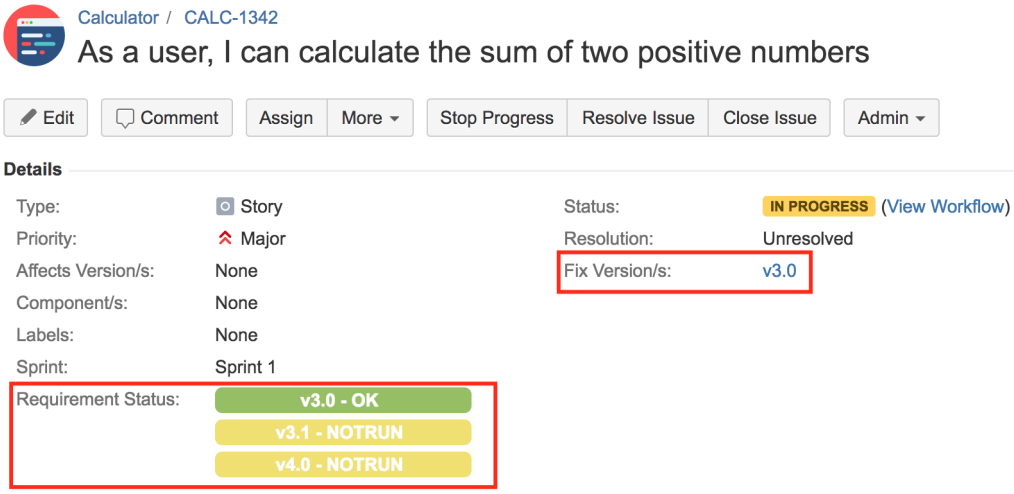
The "Requirement Status" custom field is a **calculated** field, usable in the context of requirement issues, that calculates the requirement coverage for a specific version. The version for which it calculates the coverage depends on the behavior defined in a global configuration (**see configuration details** here).

For example, you may prefer to always calculate "Requirement Status" based on the latest test run(s), regardless of which version it/they was/were executed. Or, you may prefer to calculate it for the earliest version yet to be released (i.e., the next version). In the latter case, it takes into account the order of the versions within the project settings page.

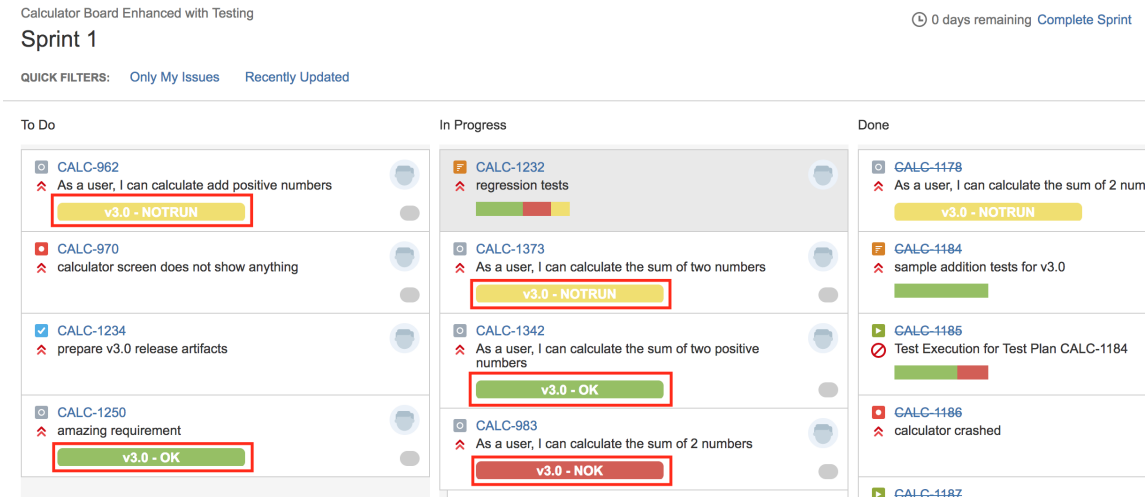
Keep in mind that a requirement is assigned to and implemented in some version. However, the requirement will still live after that version, at least, for some time. Therefore, when someone mentions a "status of a requirement", you have to ask "In which version?". The status of the requirement is therefore dependent on the executions you make for the associated Tests in some version of the system, which may by different from the version in which the requirement was initially implemented in.

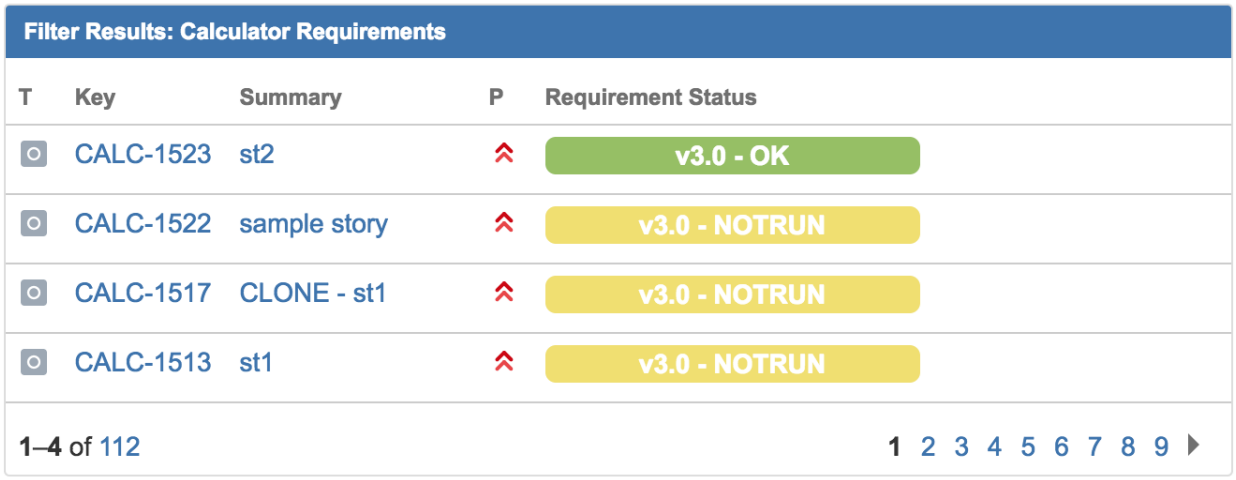
In the following example, the first image shows the calculated value for the status of requirement for the next-to-be-released version (i.e., v3.0), while the second image shows the calculated values for all the unreleased versions. This behavior was changed from "Earliest unreleased version" to "Unreleased Versions" in the Custom Fields Preferences of the Xray settings.





This field is searchable, so you can make statistics with it or use it in gadgets. However, if you wish to search for requirements in some status, we recommend you use the **requirements**() JQL function (see more information here). You can also add it to the cards of your Agile boards to have real-time feedback of its status shared across your team members.





Learn more

For more information on Requirement Status calculation, please see the Requirements Coverage Analysis.

Revision

The system revision (or code revision) being tested in the context of a given Test Execution. It is an open field that can contain the Git commit hash or the SVN revision, for example. Some users use it to put the build number/identifier.

Steps Count

Number of Steps in a Manual Test.

Test Type

The Test type (e.g., Manual, Cucumber, Generic).

TestRunStatus

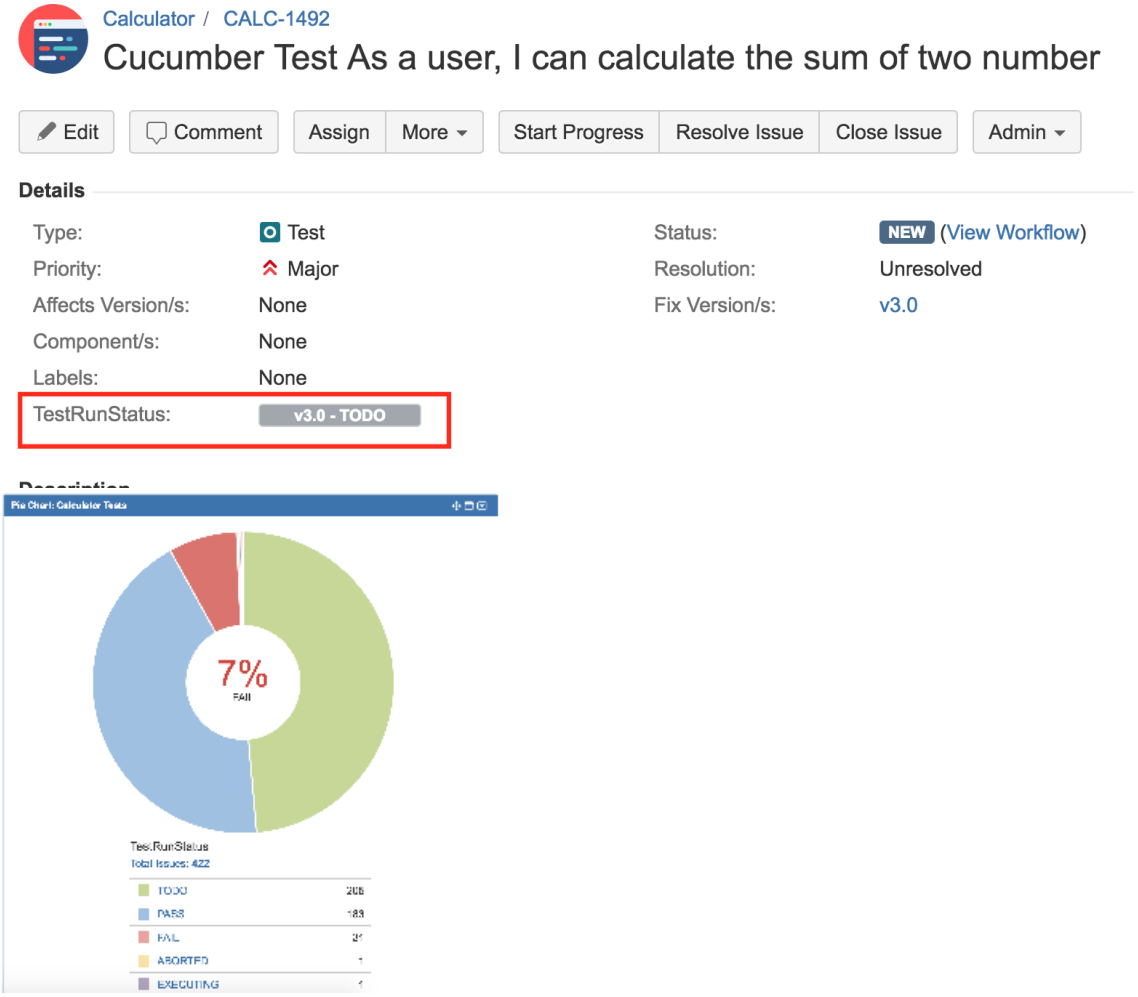
Whenever we're speaking about the status associated with a Test, we may be talking about different things:

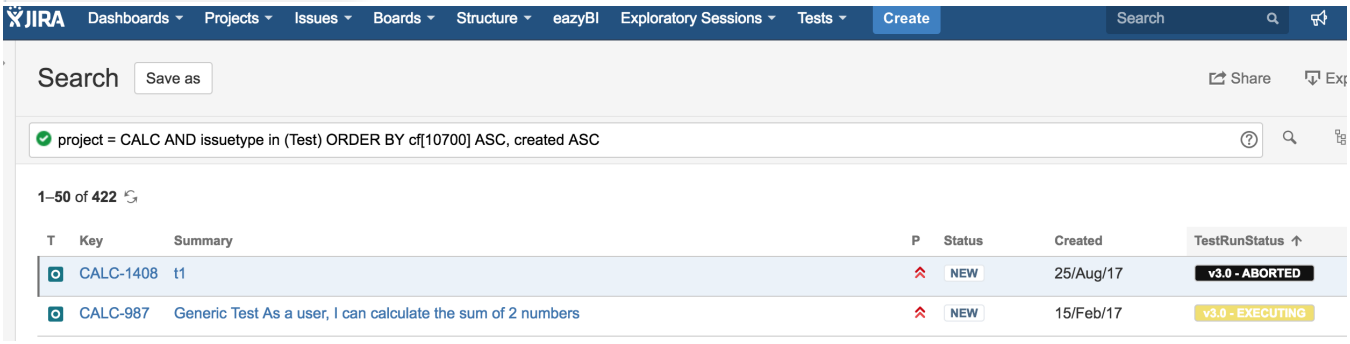
- workflow status associated to the Test issue (e.g., "New", "In Progress", "Closed")
- (coverage) status for some version, taking into account the runs made for that Test in some version of the system
- "TestRunStatus" custom field, which we will detail next

The "TestRunStatus" custom field is a **calculated** field, usable in the context of Test issues, that calculates the "status" (not the workflow status) of the Test for a specific version, considering the executions (i.e., test runs) for that version. The version for which it calculates the status depends on the behavior defined in a global configuration (**see configuration details** here).

The calculated value for the field also depends on the Test Environments, if they're being used (as mentioned here). For example, you may prefer to always show the latest test run result regardless of the version. Or, you may prefer to show the status of the latest run executed on the earliest version yet to be released (i.e., the next version). In the latter case, it takes into account the order of the versions within the project settings page.

Although a Test in Xray is essentially a test case template, i.e., not related to any execution whatsoever, you can add this field to the Test view screen as a way to display the current status for some version right on the "test template" screen (i.e., Test).





Please note

At first glance, the name of this custom field may be a bit misleading. Although it is named "TestRunStatus", don't confuse it with the status (i.e., the result) of some recorded Test Run.

As mentioned above, the TestRunStatus custom field eventually takes into account many Test Runs.

This field is searchable, so you can create statistics with it or use it in gadgets. If you want to do a search based on it, please see the proper syntax here.

Test Plans associated with a Test

Test Plan(s) issue(s) associated with this Test.

Test Sets associated with a Test

Test Set(s) issue(s) associated with this Test.

Tests Count

Number of Tests within a Test Set or a Test Plan issue.

Tests association with a Pre-Condition

Tests associated with the Pre-Condition.

Tests associated with a Test Plan

Tests associated with the Test Plan.

Tests association with a Test Set

Tests associated with the Test Set.

Tests association with a Test Execution

Tests associated with the Test Execution.

Test Environments

The Test Environment(s) associated with a given Test Execution. This is an open field, similar to a label; you should try to reuse existing Test Environments. By default, it is empty.

Test Execution Defects

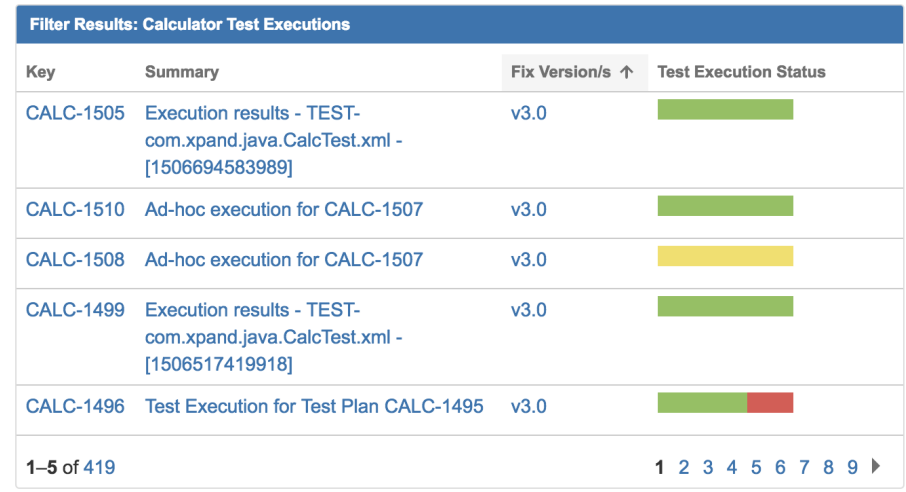
Shows the number of defect issues associated with a Test Execution issue. These can be directly linked with the issue using Jira issue links or defects that were created during the Test Runs.

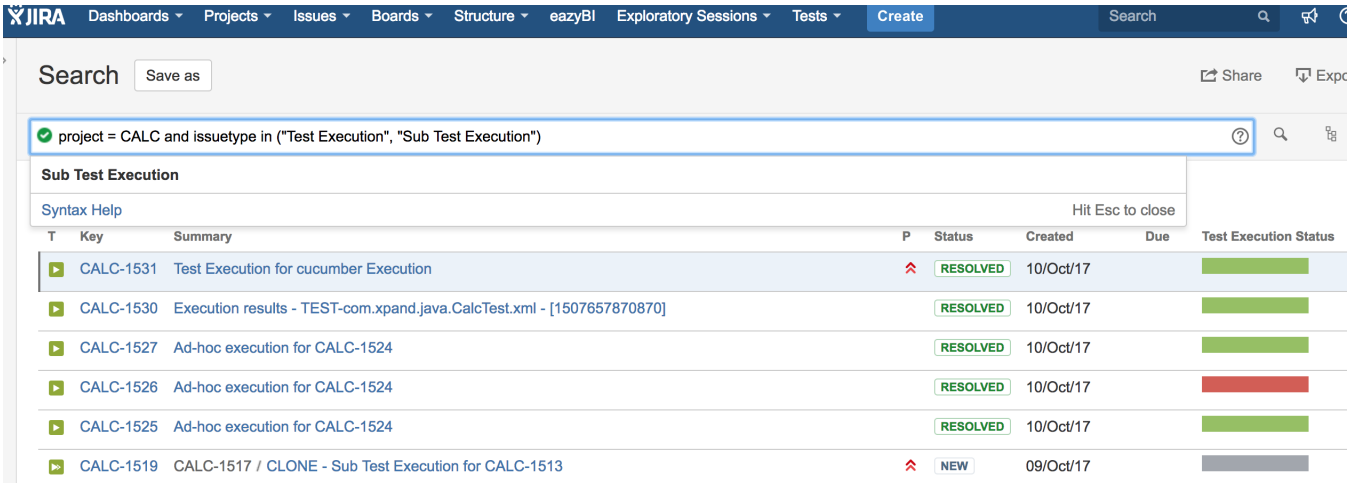
Test Execution Status

This custom field provides information about the progress (i.e., the "Overall Execution Status") of the Test Execution issue.

It can be used in gadgets, included in the cards of Agile boards, or used as a column in the Jira issues listing.

This field is non-searchable.



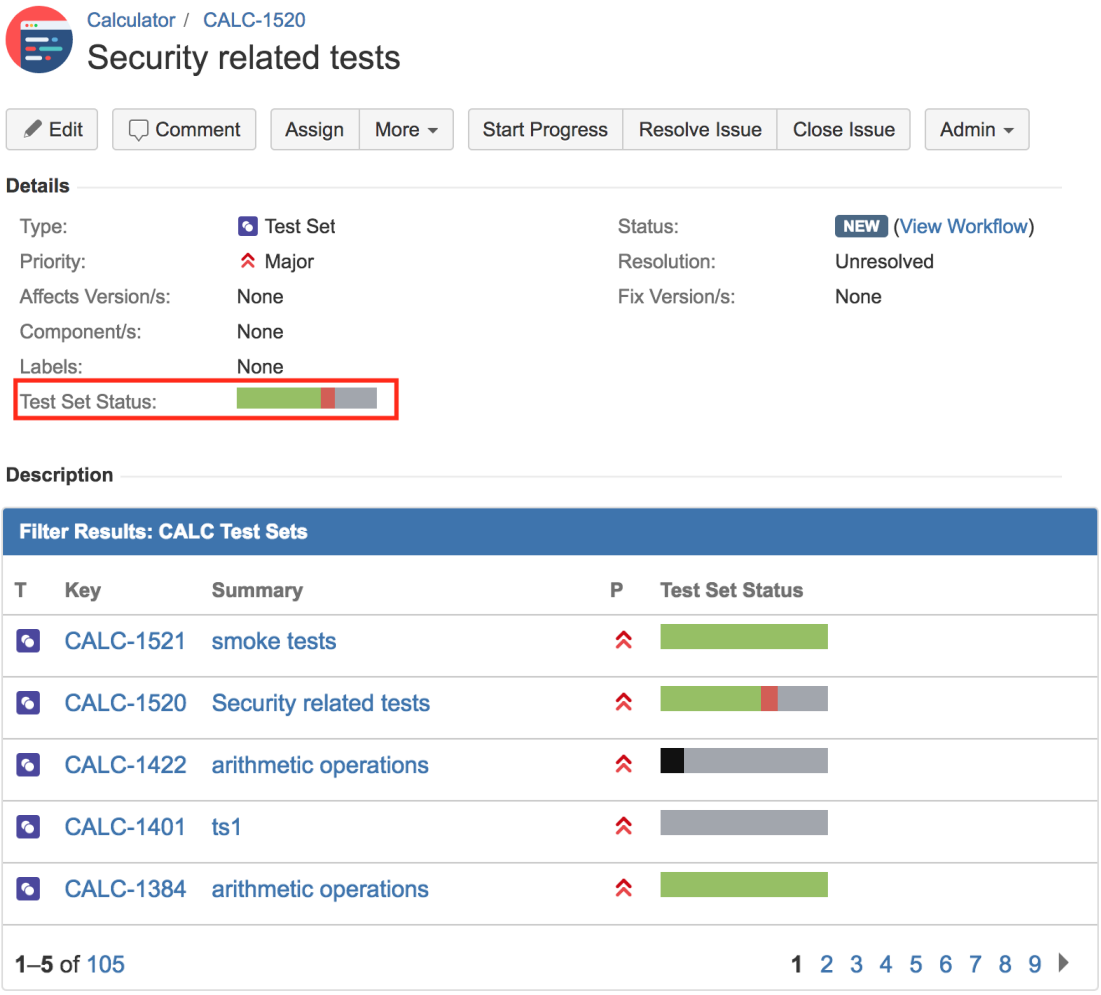


Test Set Status

The "Test Set Status" custom field is a **calculated** field, usable in the context of Test Set issues, that calculates the "status" (not the workflow status) of all the Tests within the Test Set, for a specific version, considering the executions (i.e., test runs) for that version. The version for which it calculates the status depends on the behavior for that field defined in a global configuration (**see configuration details** here).

Although a Test Set in Xray is basically a list of Tests, i.e., not related to any execution whatsoever, you can add this field to the Test Set view screen as a way to display the current status of the Tests that are part of it in some version. This can be quite useful if you have some critical Test Sets and you want to quickly have a high-level idea of the status of the Tests that belong to it.

This field is non-searchable.

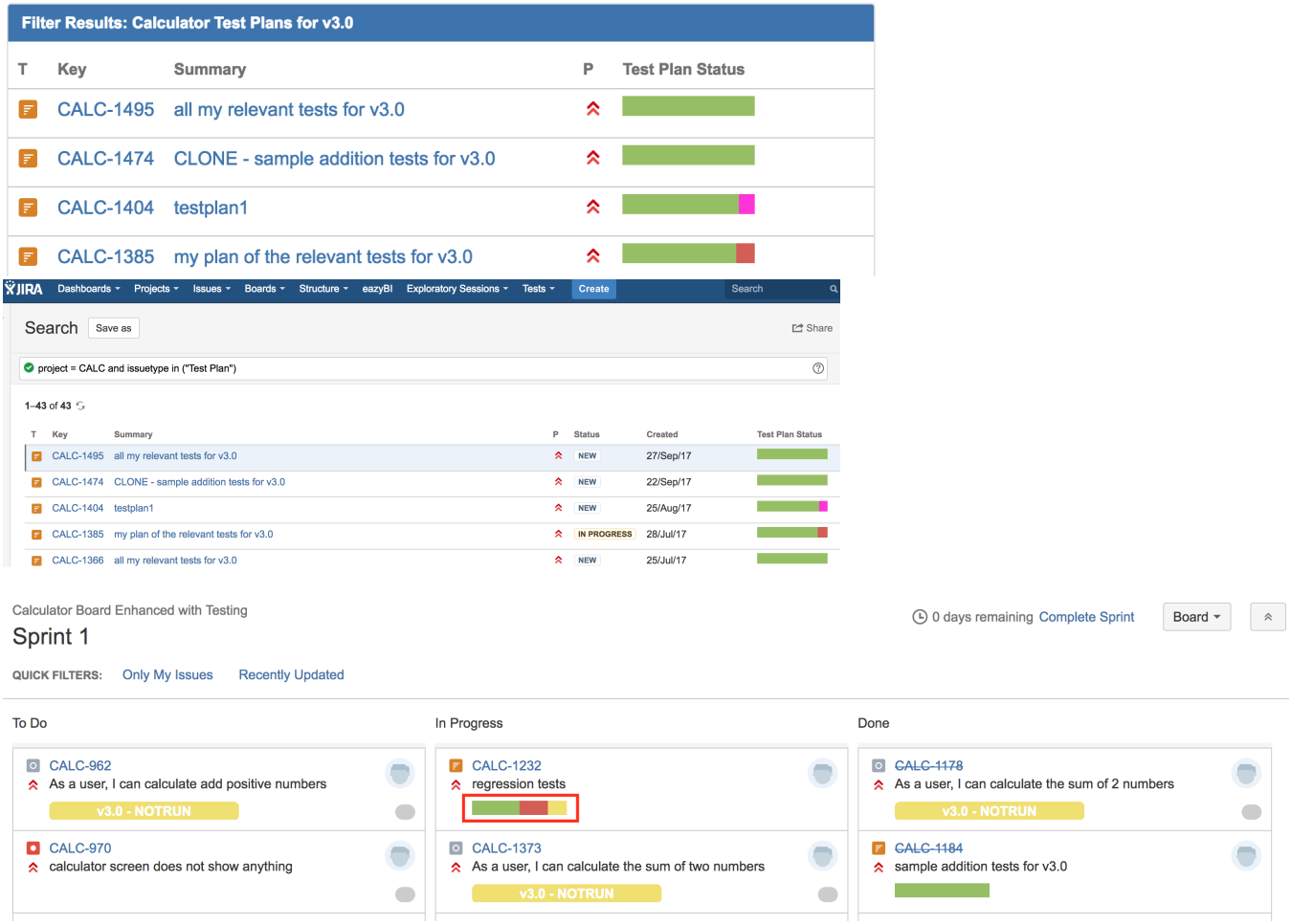


Test Plan Status

This custom field provides information about the progress (i.e., the "Overall Execution Status") of the Test Plan issue.

It can be used in gadgets, included in the cards of Agile boards, or used as a column in the Jira issues listing.

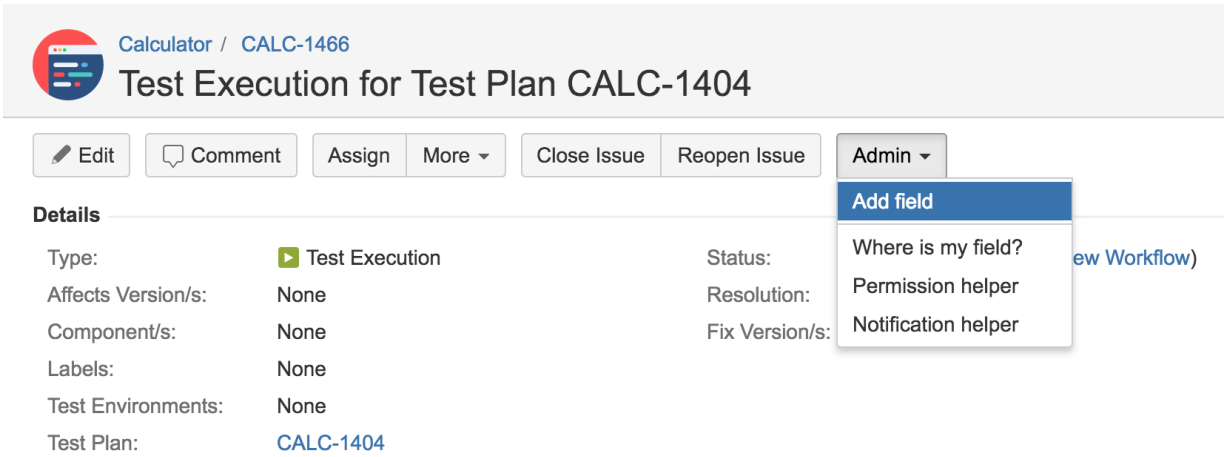
This field is non-searchable.

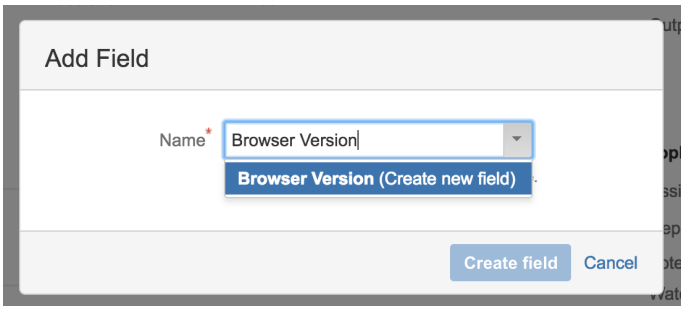


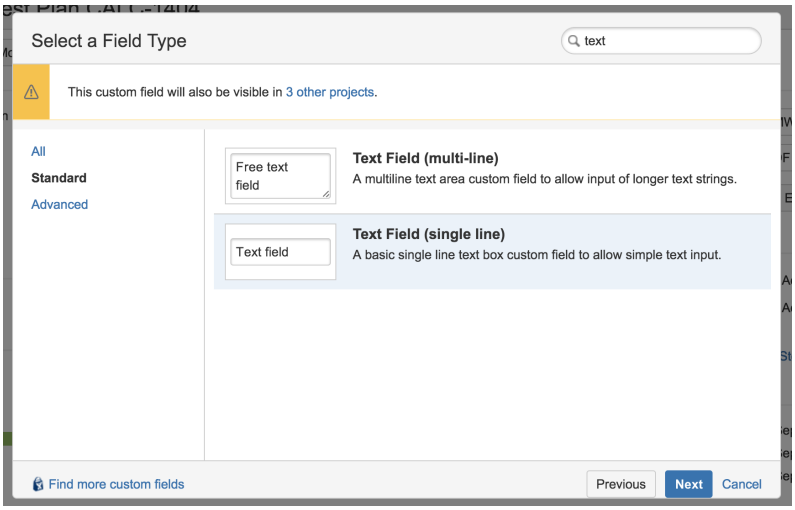
Adding your own custom fields

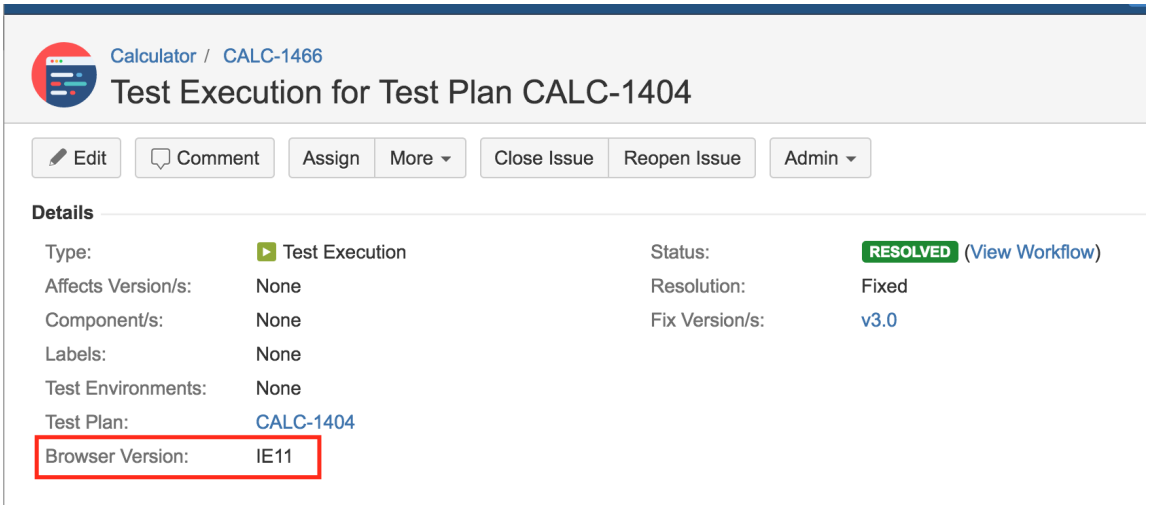
You may add CFs to Tests, however, they're not copied to the Test Run. Custom fields in Tests can be seen as complementary, non-relevant information.

Adding custom fields is not limited to the Test issue type; they can also be added to other Xray issue types, including the Test Execution and the Test Plan. For example, in the Test Execution, you could add a custom field to identify the browser version. Note that, in this example, if you wanted to analyze your Test Execution per browser, it would be preferable to use the Test Environments custom field instead; otherwise, you can add some custom fields to the Test Execution if their purpose is just to add more context to it.









Recommendations

- Don’t reinvent the wheel. Use standard fields, such as labels and components, and avoid unnecessary creation of custom fields.
- Remember that custom fields added to Tests are not copied to Test Runs, so they’re volatile/informative only; they’re not part of the specification.