

Archiving Xray Issues

Jira Datacenter supports [archiving of issues](#) since v8.1. By archiving issues, Jira will remove them from the indexes. The archived issues will still be accessible in read-only mode, however, they will not appear in search results. This enhances Jira performance because it stores less data in the indexes.

As many of the Xray entities are normal Jira issue objects, archiving its issues has the same effect of archiving any other Jira issue type. But additionally, Xray extends this feature by archiving some of its internal data.

The goal of this page is to describe the effects of archiving issues on Xray.

- [Archiving Test Executions](#)
 - [Archiving Test Runs](#)
- [Archiving Tests](#)
 - [Archiving Test Runs](#)
- [Archiving Test Plans](#)
 - [Effects in the Test Run and Requirement status scope](#)
- [Archiving Requirements](#)
 - [Archiving Sub-Test Executions](#)
- [Archiving Test Sets and Pre-Conditions](#)
- [Archiving Projects](#)
- [Impact on Xray Issue Links](#)

Archiving Test Executions



Archiving old Test Executions is a good use case for archiving Xray issues, especially if continuous integration processes are used. These can quickly create many Test Execution issues while importing new execution results into Xray on a daily basis. These issues are good candidates to be archived.

Archiving Test Execution issues will hide them from:

- Test Executions section of the Test Plan issue view
- Xray built-in [reports](#) and [gadgets](#)
- [Document generator](#) exports
- Search results from [Xray JQL functions](#)

Archiving a Test Execution will also **archive its Test Runs**.

Archiving Test Runs

Archiving a Test Execution issue automatically archives its related [Test Runs](#). Doing so will **not delete** the Test Run information from database, instead they will be:

- Read-only: You will still be able to navigate into the execution page and see the execution results but will not be able to edit
- No longer used for the calculation of the Test Run Status and [Requirement Status](#)
- Omitted from the Test Runs section in the [Test](#) and [Test Execution](#) issue view (including Overall Execution Status bar)
- Hidden from the execution inner tables in the [Tests section of the Test Plan](#)
- Excluded from Xray built-in [reports](#) and [gadgets](#)
- Hidden from [document generator](#) exports


A message will be shown in the Test and Test Execution issue view informing there are archived Test Runs:



The Test Runs in this Test Execution are archived

This Test Execution contains [2](#) archived Test Run(s). Archived Test Runs will not be considered in reports, gadgets, or the test run status and requirement status calculation.

And also in the execution page:

 This Test Run is archived and will not be considered in reports, gadgets, or the test run status and requirement status calculation.

Execution Status  PASS

Started On: 26/Oct/22 10:28 AM

Finished On: 26/Oct/22 10:28 AM

Assignee: None

Versions: -

Executed By: admin

Revision: -

Tests -
environments:

 Comment

Preview Comment

 Execution Defects (0)

...

 Execution Evidence (0)

...

Test Runs archived via Test Execution issue can be restored at any time by restoring the Test Execution issue.

Archiving Tests

Archived Test issues:

- No longer participate in the calculation of [Requirement Status](#)
- Are no longer taken in account in the [Test Coverage](#)
- No longer appear in the [Tests section in the Test Plan](#) (including Overall Execution Status bar), Test Set and Pre-Condition issue view ¹
- Are hidden from [Test Repositories](#) and [Test Plan Boards](#)
- Are excluded from Xray built-in [reports](#), [gadgets](#) and [document generator](#) exports
- No longer appear in search results from [Xray JQL functions](#)

Archiving a Test will also **archive its Test Runs**.

Archiving Test Runs

Archiving a Test issue will also archive its related [Test Runs](#) with the same effects described previously when archiving via Test Execution issue.

Test Runs archived via Test issue can be restored at any time by restoring the Test issue.

Archiving Test Plans

Archiving Test Plans will hide them from:

- Test Plans section in the Test issue view ¹
- Xray built-in [reports](#), [gadgets](#) and [document generator](#) exports
- Search results from [Xray JQL functions](#)

Archived Test Plans will also no longer be used as **scope for the Test Run Status and Requirement Status**.

Effects in the Test Run and Requirement status scope

Archived Test Plans are no longer used as [scope](#) for the Test and Requirements status custom fields, which affects the scope analysis of Requirements and Tests: After archiving a Test Plan *TP*, a given Requirement *R* that was **NOK** in the scope of *TP* will be **UNCOVERED** for the same scope *TP*.

Another effect of not being used as scope is that the Latest Statuses of Tests under a Test Plan issue will not be shown anymore. For this reason, when archiving a Test Plan, a message will be displayed informing of the situation:

▼ Details

Type:  Test Plan
Affects Version/s: None
Component/s: None
Labels: None

Status: **CLOSED**
Resolution: Done
Fix Version/s: None

▼ Tests

[Test Plan Board](#)

This Test Plan is archived

This Test Plan is archived and will no longer be used as scope for the test run status and requirement status calculation. Therefore, the below information can no longer be displayed:

- Test Plan progress bar
- Latest status for each Test in the Test Plan scope

An archived Test Plan issue can be restored anytime, doing so will retrieve the Test Plan state prior to archiving and all of the Test Plan-scoped statuses for Tests and Requirements.

Archiving Requirements

Archived Requirements will be excluded from:

- Xray built-in [reports](#) and [gadgets](#)
- [Document generator](#) exports
- Search results of [Xray JQL functions](#)

Archiving Requirements will also **archive their Sub-Test Executions**.

Archiving Sub-Test Executions

Each issue is archived along with all its subtasks.

In case a Requirement is archived, its [Sub-Test Executions](#) will be archived as well. A Sub-Test Execution has the same functionalities of a Test Execution so archiving them have the same effect as archiving a Test Execution issue, namely all its Test Runs will be archived.

Archiving Test Sets and Pre-Conditions

Archived Test Sets and Pre-Conditions will be excluded from:

- Xray [reports](#) and [gadgets](#)
- [Document generator](#) exports
- Search results of [Xray JQL functions](#)
- Test Sets and Pre-Condition sections in the Test issue view ¹


Archiving Projects

When archiving a Project all its issues are archived as well, including the Xray ones.

The effects of project archiving will be the same as archiving issues.

For example, the archived issues will no longer be present on JQL results and if Test or Test Executions are found, Xray will also archive their internal data, namely its Test Runs.

Impact on Xray Issue Links

 This section refers to the Xray custom issue links only (e.g. Tests associated with Pre-Conditions, with Test Sets, etc.) and not to the Jira native issue links.

Whenever issues are archived, those will no longer be present on JQL search results.

Due to the fact that Xray Issue Data Table components, used for presentation of Xray issue associations, are built on top of JQL searches and that archived issues no longer appear in those search results, the links will no longer be shown in some issue views. Because of that, there is no way to know if an issue is still linked to an archived one.

Whenever this happens, a warning message will be displayed at the top of the affected Xray Issue Data Table informing that some associated issues are being hidden. The below image shows an example of the Test Sets section in a Test issue view after archiving one of its related Test Sets.

▼ **Test Sets**

Add Test Sets



There are archived Test Sets not being displayed in this section



Show entries

Columns ▼

Key		Summary	Test Count	
<input type="checkbox"/>	TEST-275	TS	2	...

Showing 1 to 1 of 1 entries

[First](#) [Previous](#) [1](#) [Next](#) [Last](#)