Importing / Migrating Data from Zephyr for Jira

Table of Contents

Introduction

Since v3.3, Xray has provided a built-in importer for Zephyr Essential for Jira.

> Xray built-in importer is only available for Zephyr Essential (previously known as Zephy Squad). Zephyr Scale (previously known as TM4J) is not supported by this migration utility.
>
> references to Zephyr in this documentation only apply to Zephyr Essential.

As of Xray v3.3, the importer only performs inline data migration (i.e., Zephyr's Test Issues are moved to Xray's Test Issues).

Before Using the Zephyr Import Tool

Please check beforehand whether the below versions are compatible with the Import tool and Zephyr, whether they meet the necessary requirements, and whether they have any existing features or limitations.

Versions Compatibility

Xray Version	Supported Zephyr Essential Version
v3.3.0 - v3.6.X	v4.X.X
From v4.0.0	v4.0.0 - v5.5.X
From v4.2.0	v4.0.0 - v5.6.X
From v5.1.0	v4.0.0 - v6.2.X
From v6.5.0	v4.0.0 - v9.2.0
From v7.9.0	v4.0.0 - v9.6.2
From v7.12.3.0	v4.0.0 - v10.1.1

Requirements Before Proceeding with the Migration

> We strongly recommend you create a backup of your Jira instance before migrating the data.

- Zephyr and Xray must be installed.
- The Project, where migration is being done, must have Xray issue types (at least the Tests, Test Executions, and Test Plans): you may use the *Add Xray Issue Types* action shortcut, which is available on the Project settings screen.
- Requirement Issue Types used in Zephyr must be configured in Xray: all the different Issue Types that Zephyr Tests cover should be configured in Xray's Issue Type Mapping settings.
- Defect Issue Types used in Zephyr must be configured in Xray: all the different Issue Types that are being used as defects in Zephyr must be configured in Xray's Issue Type Mapping settings.

- Create similar Test Statuses and Test Step Statuses in Xray; this is not mandatory but may ease the process, which will always ask you to make the mapping between Zephyr statuses and Xray counterparts.
- Make sure Zephyr is using different Issue links between Test<=>Defect and Test<=>Requirement, by going into Zephyr's configuration settings.
- Do not change, create, or delete any Issue in the Project while the importation is running.
- The only mandatory fields in Xray's Test, Test Execution, and Test Plan must be the Issue Summary and Issue Reporter.
- Make sure that the Jira workflow states that are being used by the Zephyr Test Issue Type are editable.

The current process performs an inline migration, i.e. Tests and data, are migrated to Xray, and the original entities are lost. Thus, we recommend backing up your Jira instance before performing the migration.

Also, as the amount of data to migrate may be considerably large, we advise you to perform this migration during non-working hours. Please also make sure that users are not changing data on the project while the migration is being done.

Features and Limitations

Below there's a list of the supported features and current limitations.

Most information will be migrated seamlessly, but please check the following table.

Supported Features	Unsupported

Inline migration (not cloning) of:
• Test and Test Steps.
• Test and Test Step attachments.
• Links between Tests and Defects/Requirements.
• Cycle (including Cycle folders*).
• Executions:
  ◦ Defects (Global and Step level).
  ◦ Attachments (Global and Step level).
  ◦ Result (Global and step level).
  ◦ Comments (global and step level).
  ◦ Assignee.
• Custom fields in the Test Steps and on the Execution.
	• All Zephyr executions from the Ad-hoc cycle; only the last Execution is migrated.
	• Activity information.
	• Test Statuses and Test Step Statuses configurations.
	• Cycle folders as such*.

(*) Cycle folders will be migrated to Test Executions since the semantics on Xray are a bit different in terms of entities/organization.	

Operations

Within this section, you're able to find the exact mapping of entities from Zephyr to Xray.

Mapping Information

Zephyr Entity	Xray Entity	Notes
Test: • Steps. • Attachments.	Test: • Steps. • Attachments.	If empty, the *Step* column will be filled with <*undefined*>
Cycle	Test Plan (optional): • **Summary:** <version> - <cycle's name> • **fixVersion**: version assigned to Cycle Test Execution (linked to the previous Test Plan): • **Summary:** <version> - <cycle's name> Execution • **fixVersion**: version assigned to Cycle	• The Test Execution created here will contain the Executions assigned directly to the Cycle, since in Zephyr you can have Executions at that level besides on each folder. • Please note that if the Cycle contains multiple Executions for the same Test, only the last Execution will be migrated.
Cycle's folder	Test Execution, linked to the Test Plan created earlier from the Cycle	N/A

	• **Summary:** <version> - <cycle's name> - <folder name> Execution • **fixVersion**: version assigned to Cycle	
Ad-hoc Cycle	Test Plan (optional) • **Summary:** <version> - Ah Hoc • **fixVersion**: version assigned to Cycle Test Execution, linked to the previous Test Plan • **Summary:** <version> - Ad Hoc Execution • **fixVersion**: version assigned to Cycle	N/A
Execution: • Global comment. • Global defects. • Global status. • Step results. • Step comments. • Step defects. • Step status.	Test Run: • Global comment. • Global defects. • Global status. • Step results. • Step comments. • Step defects. • Step status.	N/A

All Issues will be created in the project where the migration is being performed.

Migrating the Zephyr Test Set and Execution Custom Fields

When importing a project, Xray performs a detailed check of the possibility of migrating custom fields. This verification follows the criteria and steps described below:

1. Recreation and Reuse of Fields:
    - Xray will recreate the migrated custom fields, keeping the same name, Type, and options where applicable.
    - If the custom field to be migrated already exists in Xray with the same name and Type, it will be reused, avoiding duplications.
2. Recreating Fields with the Same Name and Different Types:
    - If a custom field with the same name already exists in Xray but with a different type, Xray creates a new custom field by prefixing it with *Zphr_Xray_<field_name>*
    - This process is recursive. For example, when migrating the *cf_1* field (a toggle in Xray), and if a single-line text type *cf_1* already exists in Xray, Xray will attempt to create *Zphr_Xray_cf_1*. If *Zphr_Xray_cf_1* already exists as a number type field, the system will continue applying the prefix (*Zphr_Xray_Zphr_Xray_cf_1*) until it can create a field with the desired name and type.
3. Field Type Conversion: during migration, *Zephyr Checkbox* Type fields will be converted to *Multiselect*. All other field types are matched directly, retaining their original Type.
4. Treatment of Options in Existing Fields: when a *Zephyr* field has options and this field already exists in Xray, the system will add the missing options to the existing options list in Xray, without overwriting the current options.
5. Warning in Case of Excessive Fields (only applied to Test Set Custom fields): if the sum of existing custom fields in Xray and new Zephyr fields exceeds the maximum allowed limit (6), Xray will cancel the migration process and notify the user of the exceeded limit.

Performing the Migration

Performing the migration is easy; however, it is currently limited to Jira administrators.

The migration follows a wizard-like interface; after going through the steps, some additional tasks are required to ensure the consistency of data.

Suppose you have a project that you wish to migrate from Zephyr for Jira to Xray and that the requirements mentioned above are met.

To start the importation process:

1

On your Jira Data Center instance, click the gear icon (Figure 1 - 1) and then select *System* (Figure 1 - 2).

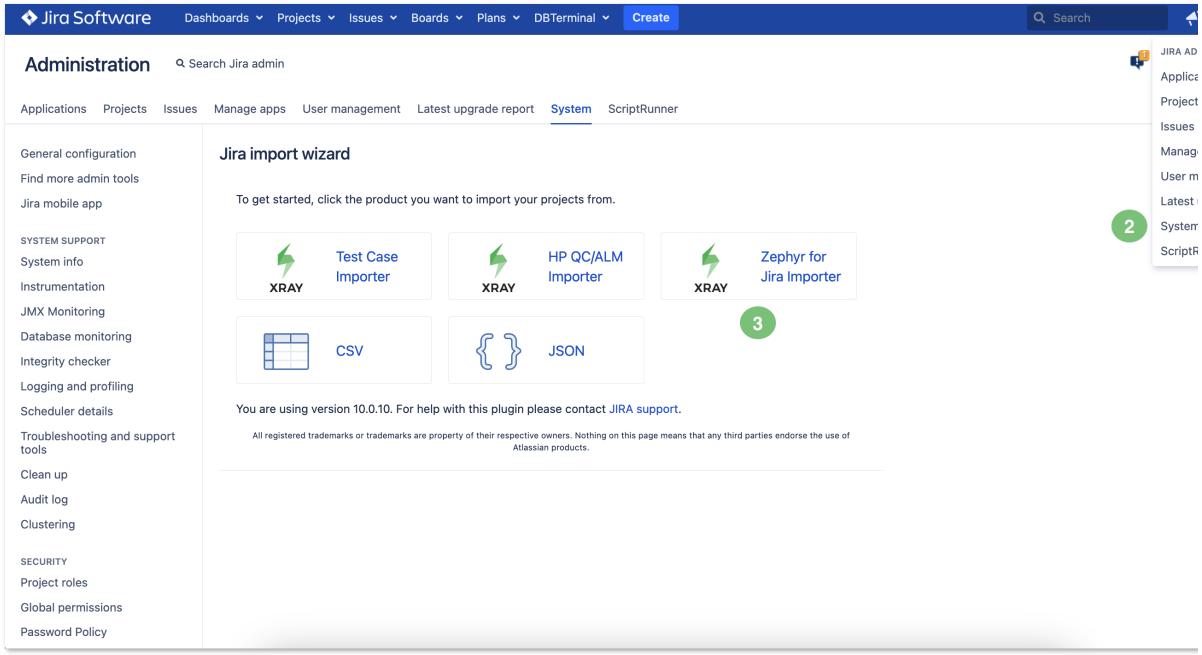

Figure 1 - Access

2

A menu will appear on the left side of the screen. There, click *External System Import* (Figure 2 - 1). On the Jira import wizard screen, select the *Zephyr for Jira Importer* option (Figure 1 - 3).

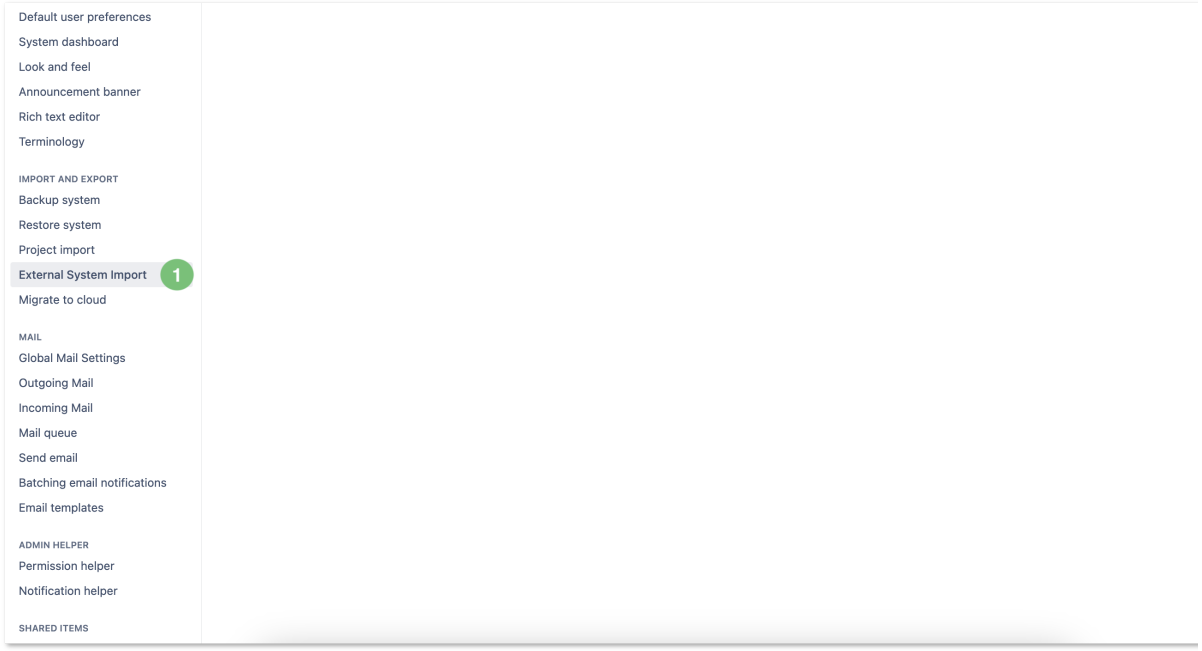

Figure 2 - Access

3

If you see the following error message (Figure 3), it's because Zephyr is using the same Issue Link Types between *Test<=>Defect* and *Requirement<=>Test*.

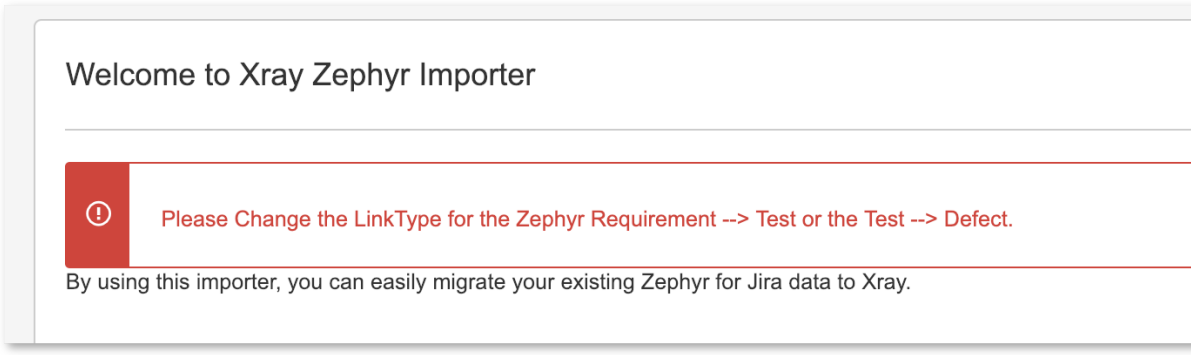

Figure 3 - Message

4

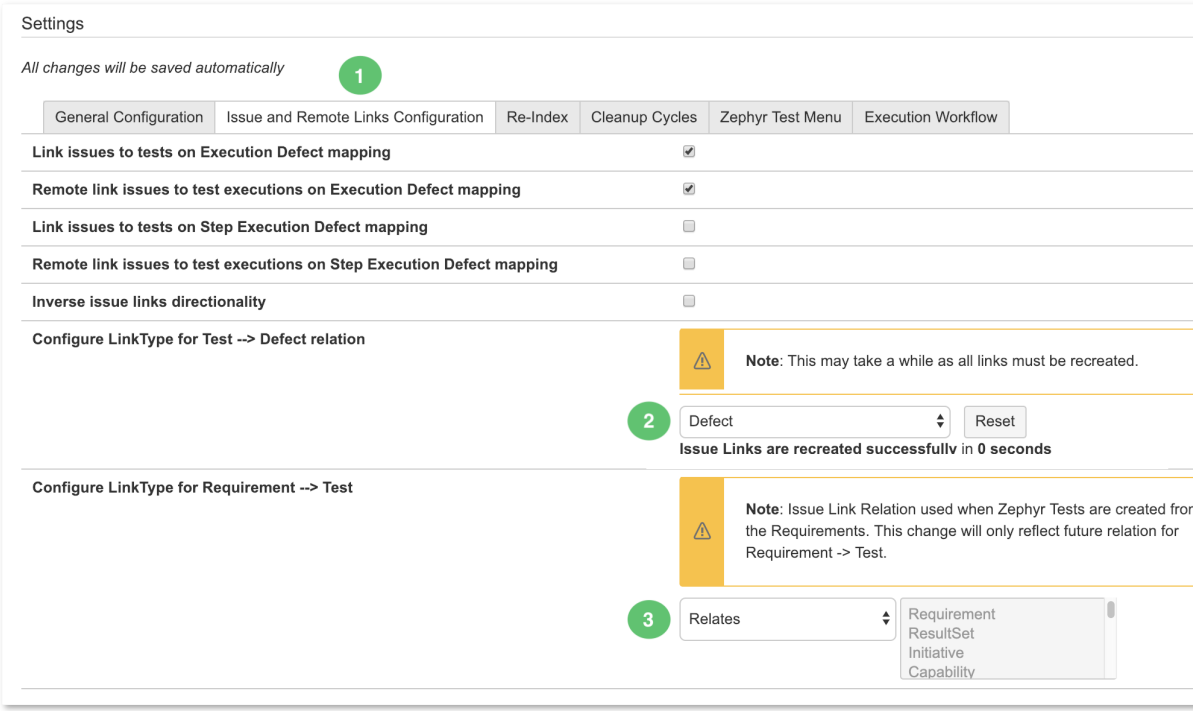

Figure 4 - Settings

Xray needs to have different relations so that it can understand the different cases. Thus, you need to change the *Linktype for Test Defect relation* to something different from the value *LinkType for Requirement Test*. For example, you can leave *LinkType for Requirement Test* with *Relates* (Figure 4 - 3) and change the *Linktype for Test Relation* to *Defect* (Figure 4 - 2).

These relations are configurable in *Add-ons > Zephyr for Jira > General Configuration > Issue and Remote Links Configuration* (Figure 4 - 1).

5

After ensuring that the Link Types are configured with distinct values, you may proceed once again with the migration process.

The importer modal (Figure 5) will show you some brief information that we advise you to read carefully.

Once you're ready, click *Being Import* (Figure 5 - 1).

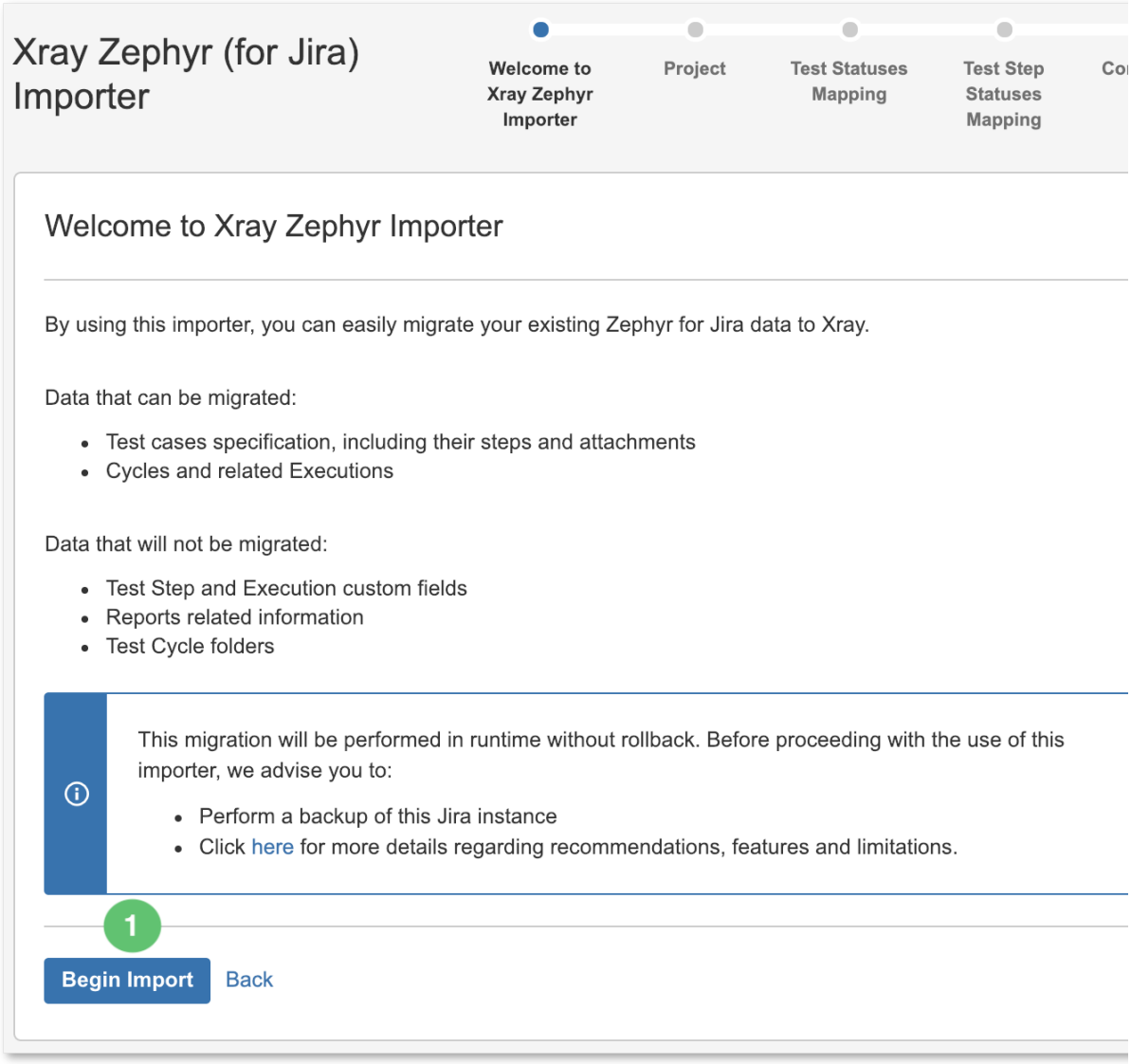

Figure 5 - Import

6

Choose the project where to perform the migration (Figure 6 - 1). This will be the project where the Xray entities will be created.

You may refine the process by (un)checking some boxes:

- *Links between requirements and tests to respective Xray Issue Link Type used for requirement coverage* (Figure 6 - 2): creates the link Xray uses for tracking coverage between Tests and requirements; by default, Xray uses the Tests Issue Link Type.
- *Zephyr Cycles to Xray Test Plans* (Figure 6 - 3): creates Test Plans based on Zephyr Cycles.
- *Zephyr Ad Hoc Cycles to Xray Test Plans* (Figure 6 - 4): creates Test Plans based on Zephyr Ad Hoc Cycles.

If the *Zephyr Cycles to Xray Test Plans* (Figure 6 - 3) and *Zephyr Ad Hoc Cycles to Xray Test Plans* (Figure 6 - 4) options are unchecked then no Test Plans will be created; nevertheless, Test Executions will always be created if Zephyr Executions exist.

Once you're finished, click *Next* (Figure 6 - 5).

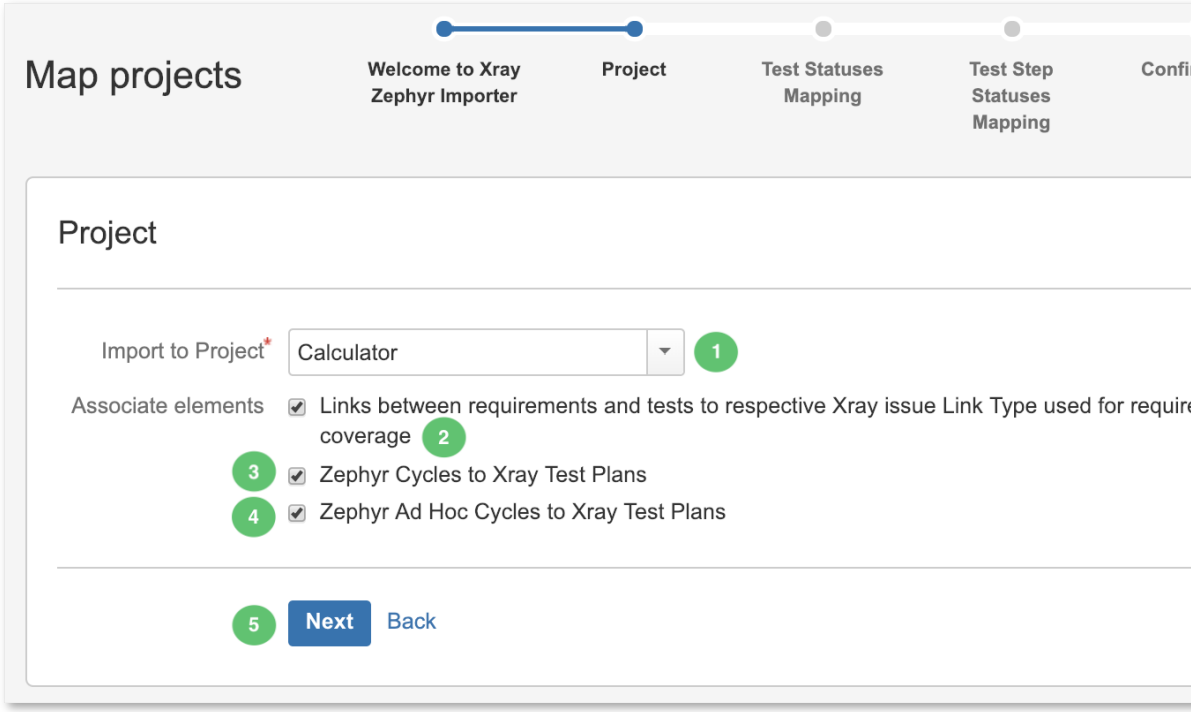

Figure 6 - Project

7

Map Zephyr's Test Statuses to Xray counterparts using the dropdown menus (Figure 7 - 1).

Once you're finished, click *Next* (Figure 7 - 2).

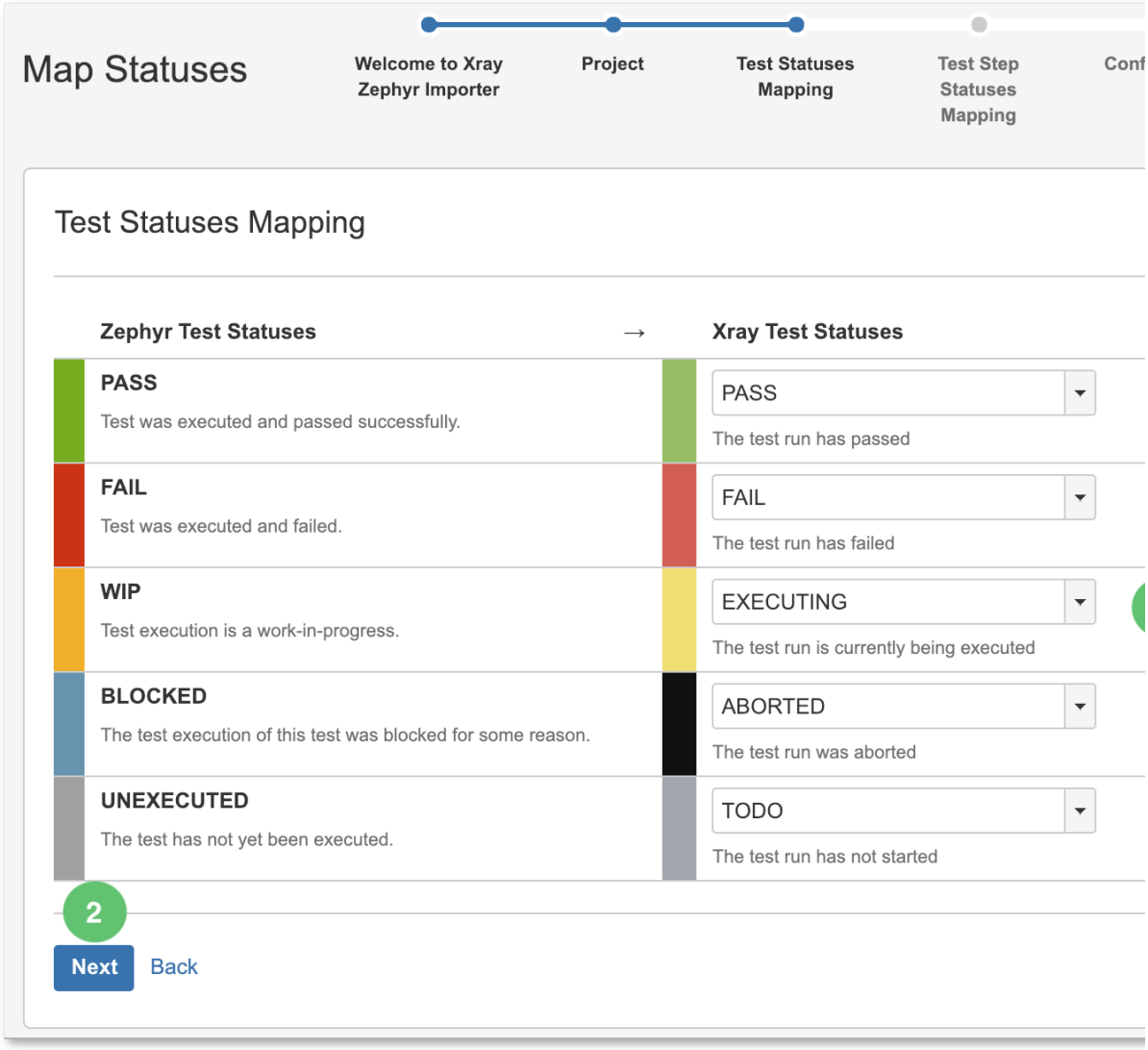

Figure 7 - Mapping

8

Map Zephyr's Test Step Statuses using the dropdown menus (Figure 8 - 1).

Once you're finished, click *Next* (Figure 8 - 2).

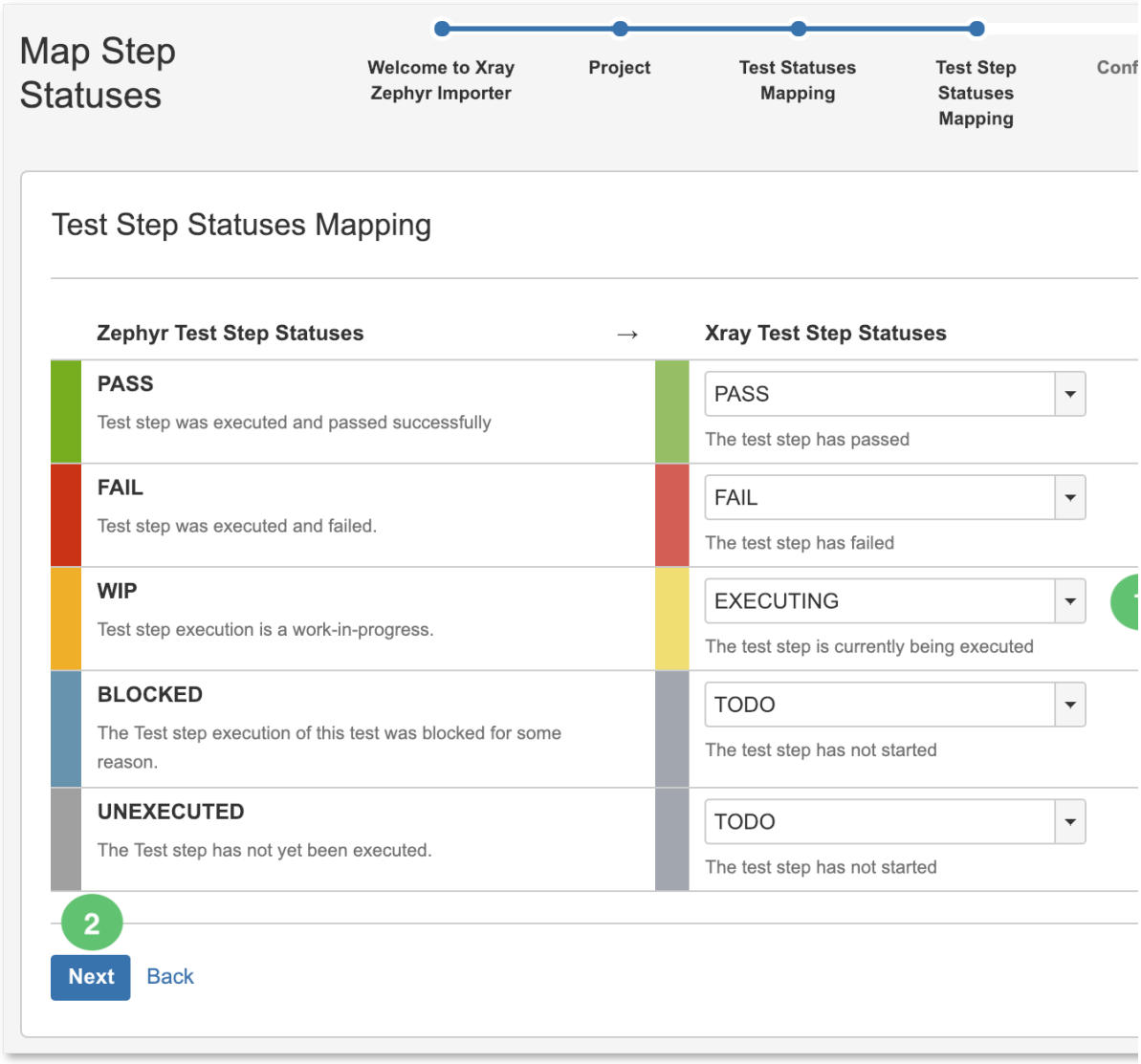

Figure 8 - Mapping

9

A final confirmation dialog presents information about the total number of Xray entities that will be created (Figure 9 - 1).

Once you're ready, click *Begin Import* (Figure 9 - 2).

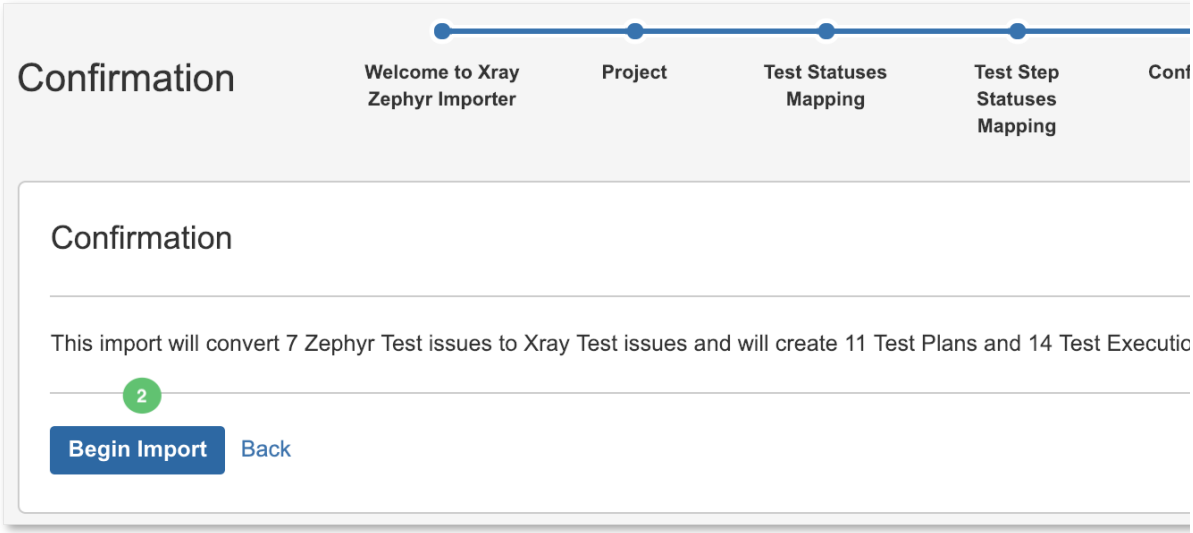

Figure 9 - Confirmation

10

Once the migration process ends, a brief summary is shown mentioning the total number of Xray Issues created and any warnings that occurred during the process (Figure 10 - 1).

You can immediately start a new importing process by clicking *Import another project* (Figure 10 - 2).

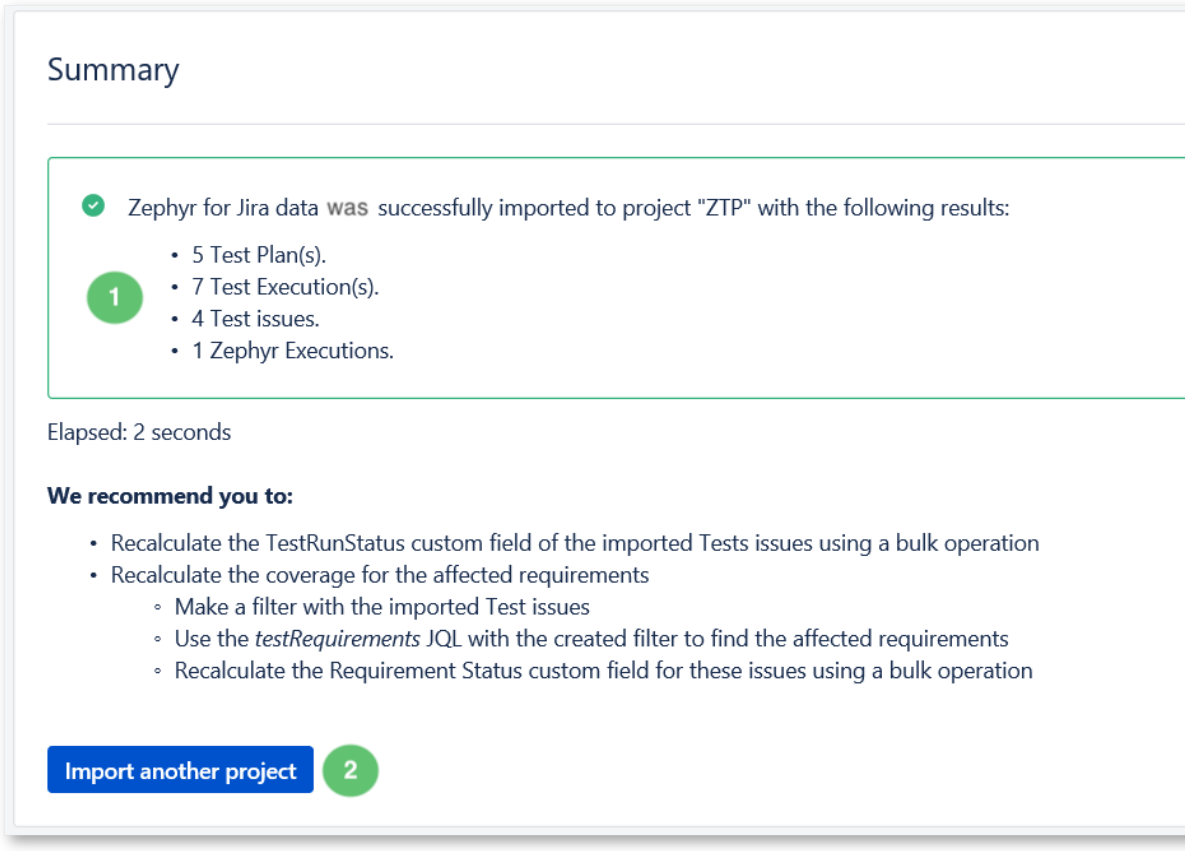

Figure 10 - Concluded

Please have a look at the following section for some additional steps before concluding the migration process.

After the Migration Concludes

After migrating data from Zephyr to Xray, you will need to perform some additional operations to recalculate the status of Tests and the coverage of the related requirements.

1

On the top menu of your Jira Data Center instance, click *Issues* (Figure 11 - 1) and then select *Search for issues* (Figure 11 - 2).

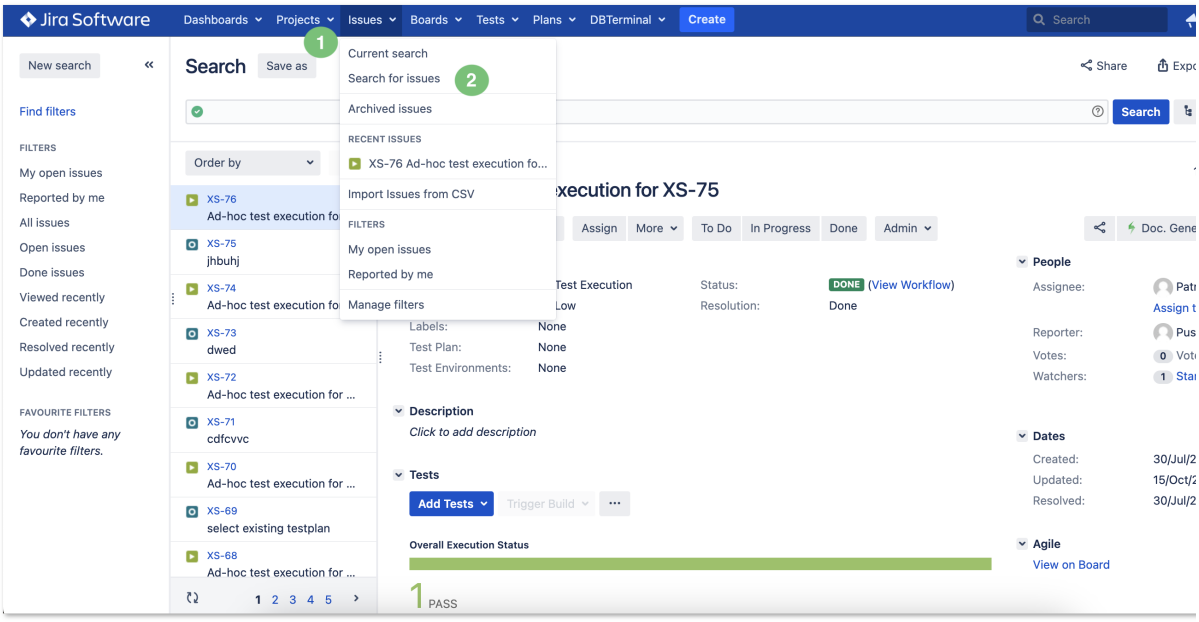

Figure 11 - Search

2

Search for the importation using the search bar (Figure 12 - 1).

Reset the *TestRunStatus* custom field of the migrated Tests (Figure 12 - 1). You can use the link provided on the final screen mentioned earlier to quickly obtain the created Tests.

You will be redirected to the Issues search page.

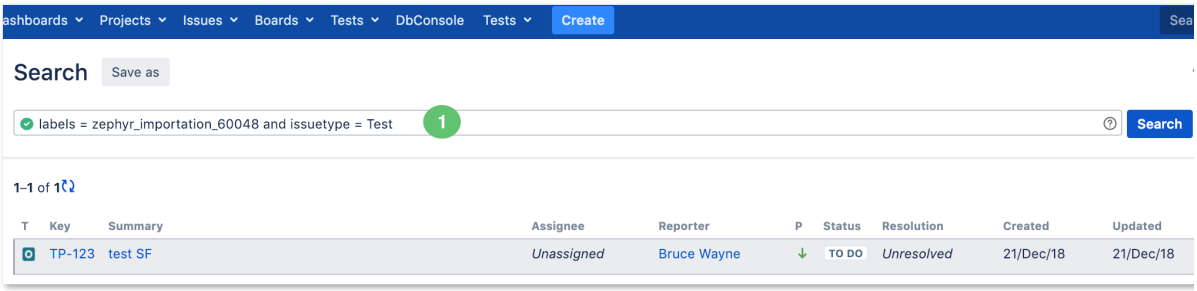

Figure 12 - Reset

3

Click the *Save as* button (Figure 13 - 1) to save this search as a filter (you will need it afterwards).

A modal opens (Figure 13). Enter a filter name (Figure 13 - 2) and click *Save* when you're finished (Figure 13 - 3).

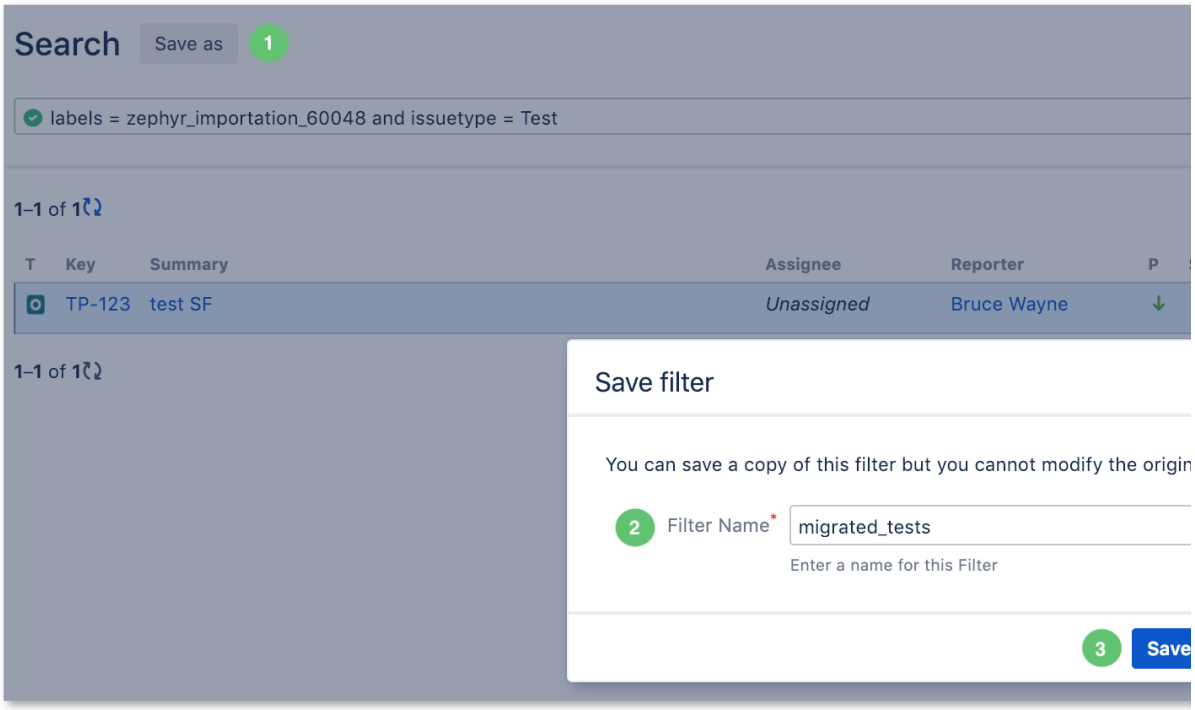

Figure 13 - Modal

4

Click *Tools* (Figure 14 - 1) and *select all 1 issue(s)* (Figure 14 - 2) to perform a bulk change operation on the Test Issues.

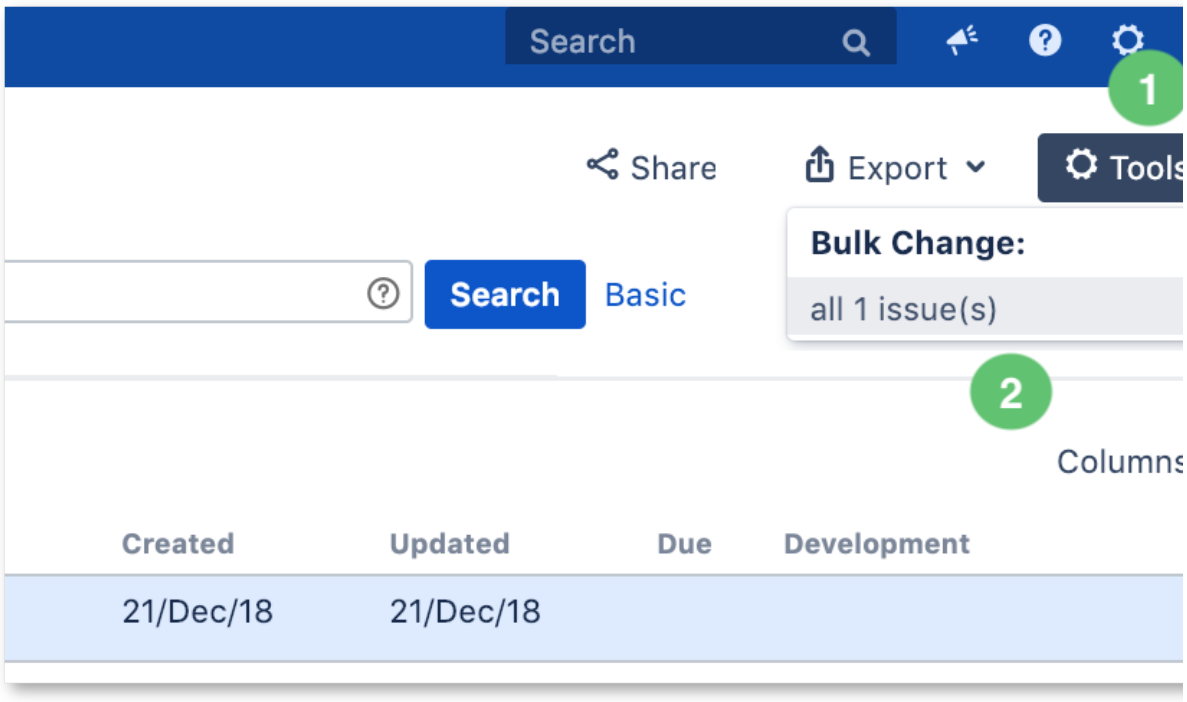

Figure 14 - Bulk

5

The Bulk Operation screen will open (Figure 15). Select the Issue(s) and then Reset the *Test Run Status Custom fields* option (Figure 15 - 1).

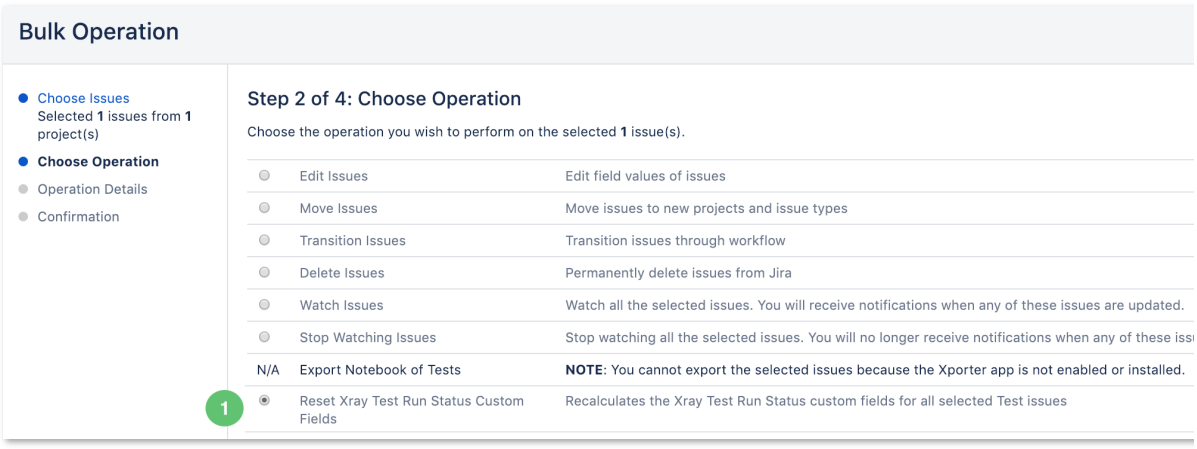

Figure 15 - Reset

6

Reset the *Requirement Status* custom field of the requirements linked to the migrated Tests: use the testRequirements JQL function using the name of the previously saved filter as an argument (Figure 16 - 1).

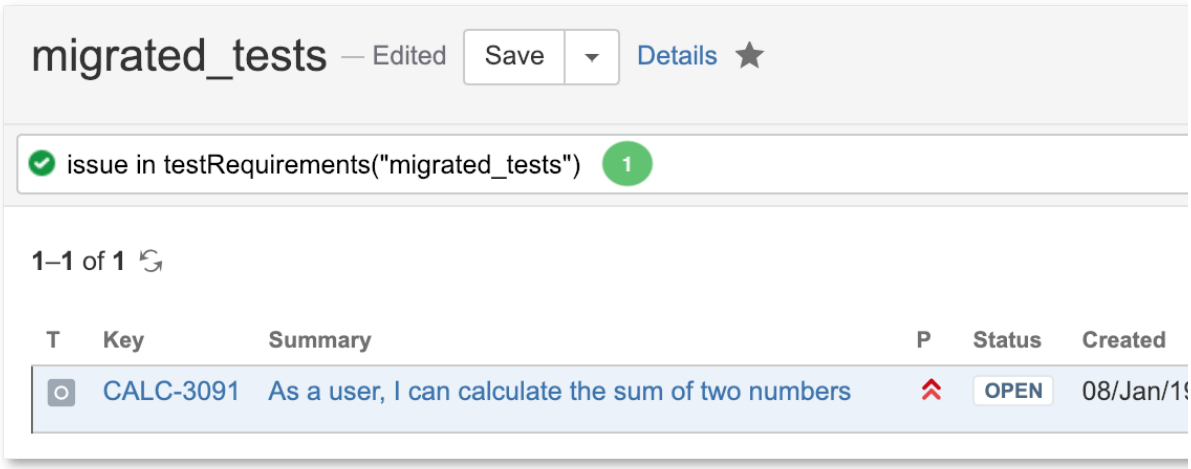

Figure 16 - Saved

7

Go back to your Data Center Jira instance screen click *Tools* (Figure 17 - 1), and *select all 1 issue(s)* (Figure 17 - 2) to perform a bulk change operation on the Requirement Issues.

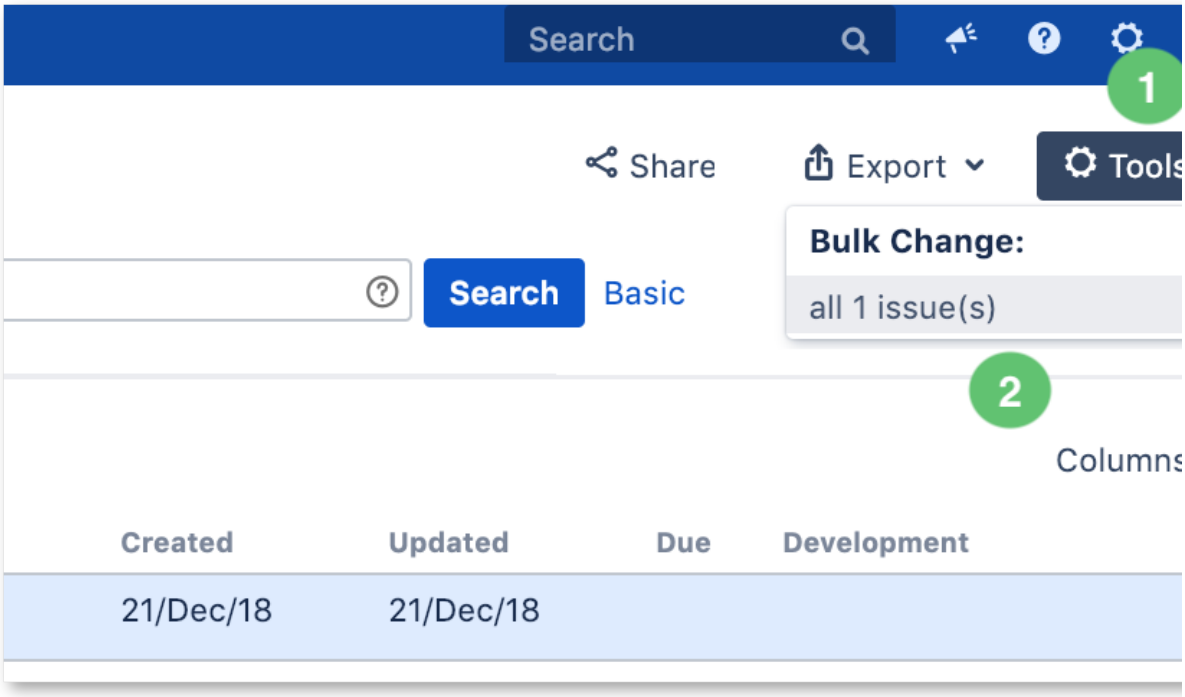

Figure 17 - Bulk

8

The Bulk Operation screen will open (Figure 18). Select the Issue(s) and then Reset the *Requirement Status Custom fields* option (Figure 18 - 1).

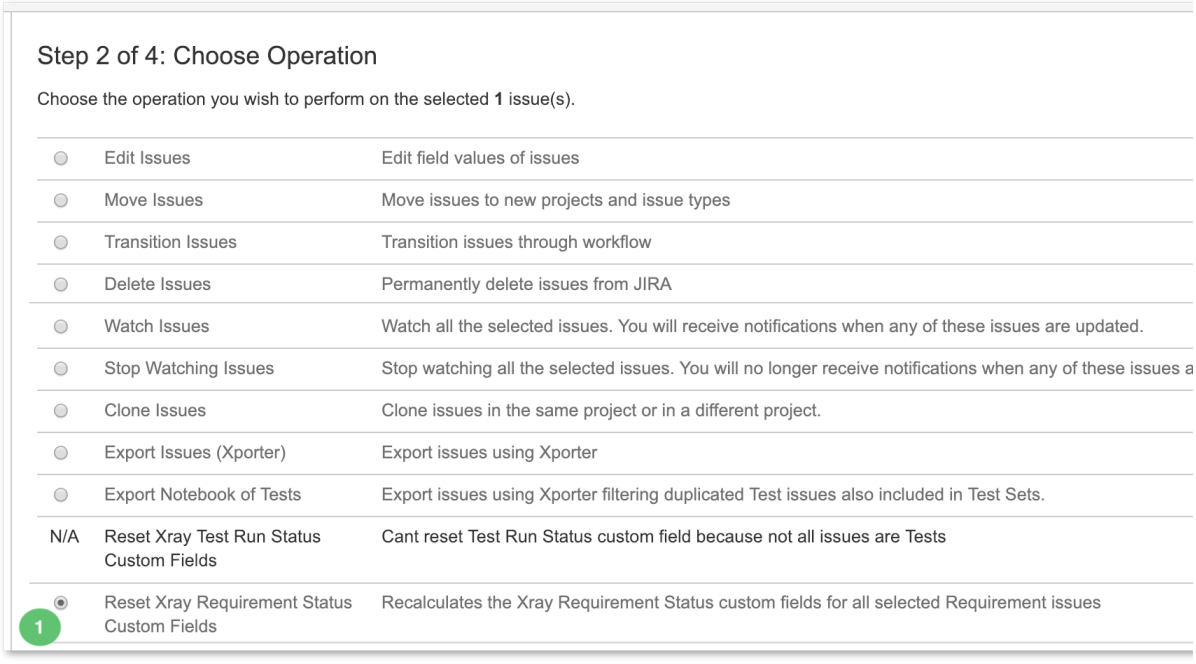

Figure 18 - Reset

If you have questions or technical issues, please contact the Support team via the Customer Portal (Jira service management) or send us a message using the in-app chat.