Importing / Migrating data from Zephyr for Jira

Xray v3.3 provides a built-in importer for Zephyr for Jira.

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

- Before using the Zephyr Import tool
  - Requirements before proceeding
  - Features and Limitations
- How it works
  - Mapping of information
- How to use
  - Performing the migration
  - After doing the migration

Before using the Zephyr Import tool

Before using the tool, please check the necessary requirements and also the existing features and its limitations.

Requirements before proceeding

- Zephyr version 4.x
- Zephyr and Xray should be both installed
- 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 available in the project settings page
- 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 are being used as defects in Zephyr should 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 field in Xray's Test, Test Execution and Test Plan should be the issue Summary and issue Reporter
- Make sure that the Jira workflow states, that are being used by "Zephyr Test" issue type, are editable. Click here for the official Jira documentation on this subject.
- **We strongly recommend you to make a backup of your Jira instance before doing the importation**

> **Please note**
>
> The current process performs an inline migration, i.e. Tests and data is migrated to Xray and the original entities are "lost".
>
> Thus, we recommend to backup your Jira instance before performing the migration.
>
> Also, as the amount of data to migrate may be considerable large, we advise you to perform this on 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, please find a list of the support features and current limitations.

Most information will be migrated seamlessly but please check the following table in more detail.

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
	• All Zephyr executions from the Ad-hoc cycle; only the last Execution is migrated
	• Custom fields in Test steps and on the Execution
	• 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.	

How it works

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

Mapping of 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 • **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	
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	

Please note

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

How to use

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

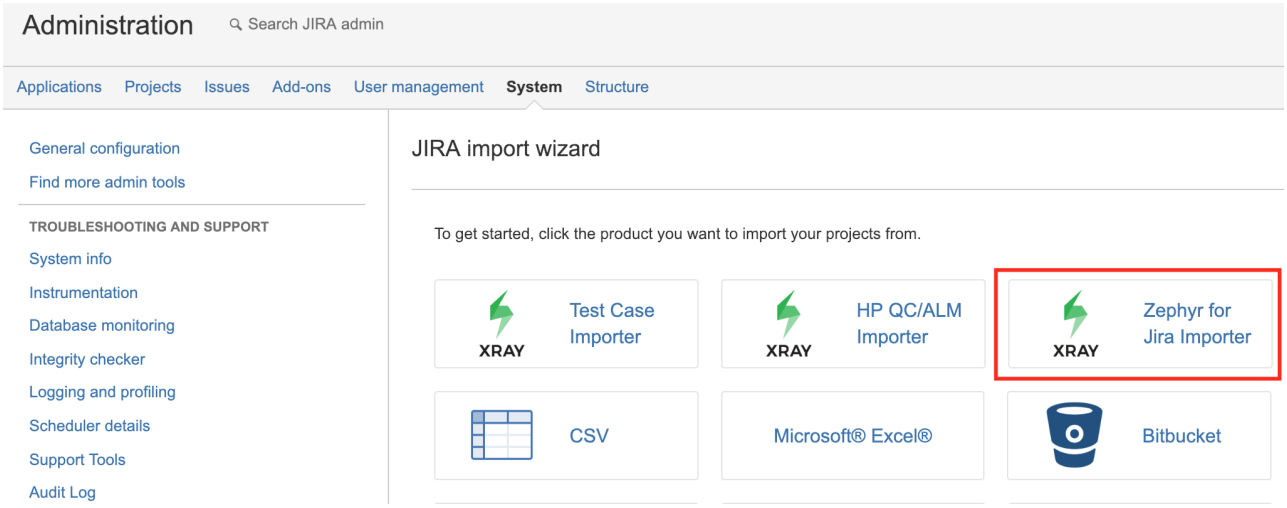
The migration follows a wizard like interface; after going through the steps, some additional tasks required to ensure consistency of data.

Performing the migration

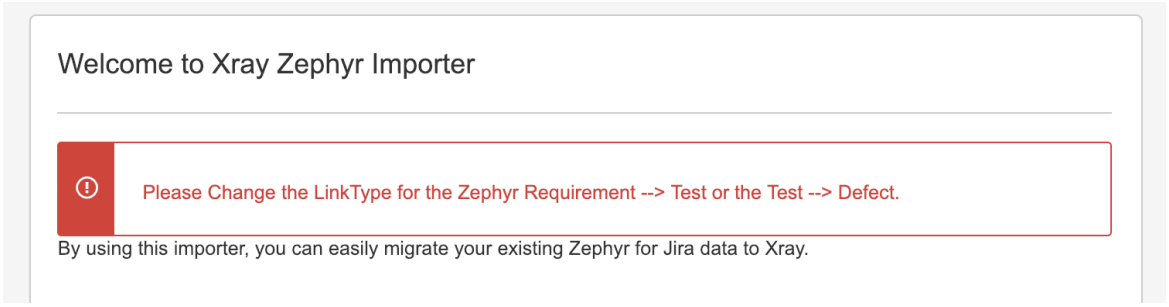
Suppose you have a project that you wish to migrate from Zephyr for Jira to Xray and that the requirements mentioned above are met.

To start the importation process:

1. Log in with an administrator account;
2. Go to System > Import and Export > External System Import;
3. Select Xray's **Zephyr for Jira Importer.**

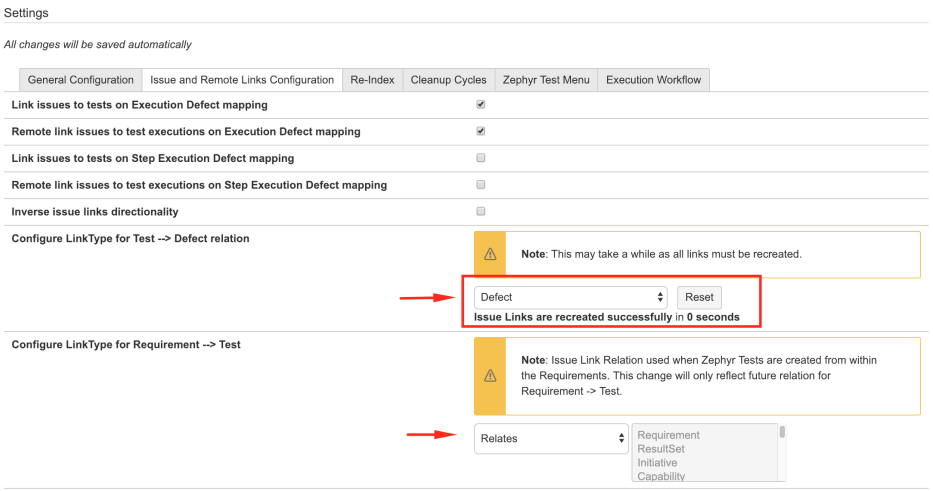


If you see the following error message, then it's because Zephyr is using the same issue link types between Test<=>Defect and Requirement<=>Test.



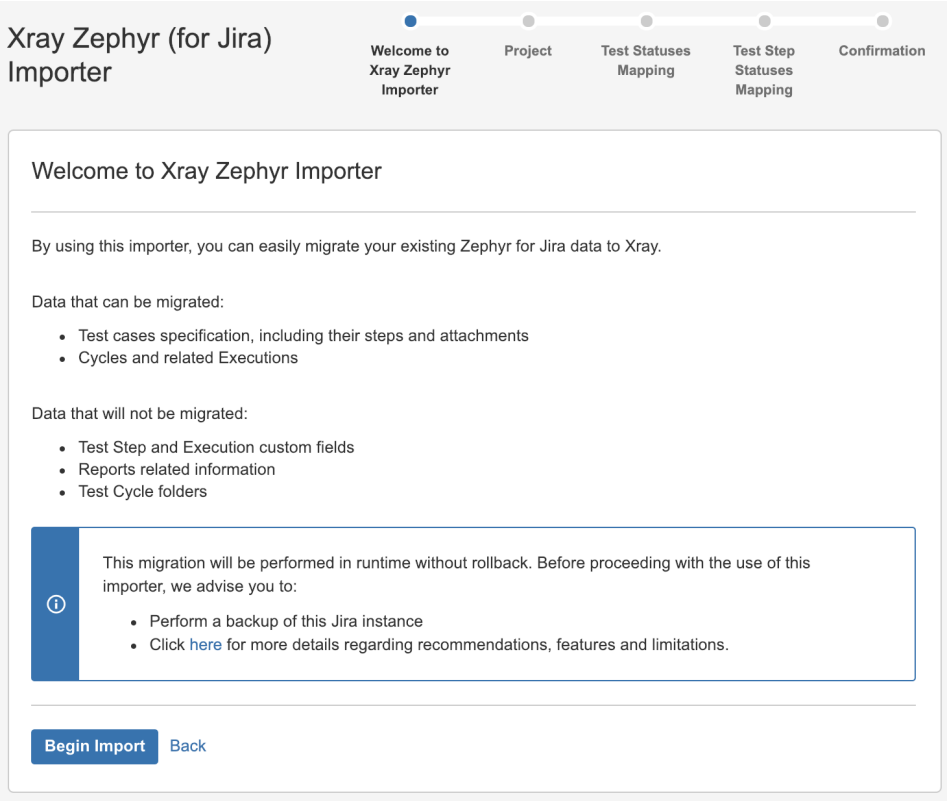
Xray needs have different relations, so that is able to understand the different cases. Thus, you need to change the "Linktype for Test Relation" to something different from the value "LinkType for Requirement Test". For example, you can leave "LinkType for Requirement Test" with "Relates" and change the "Linktype for Test Relation" to "Defect".

These relations are configurable in Add-ons > Zephyr for Jira > General Configuration > Issue and Remote Links Configuration.



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

The importer will show you some brief information, that we advise you to read carefully.

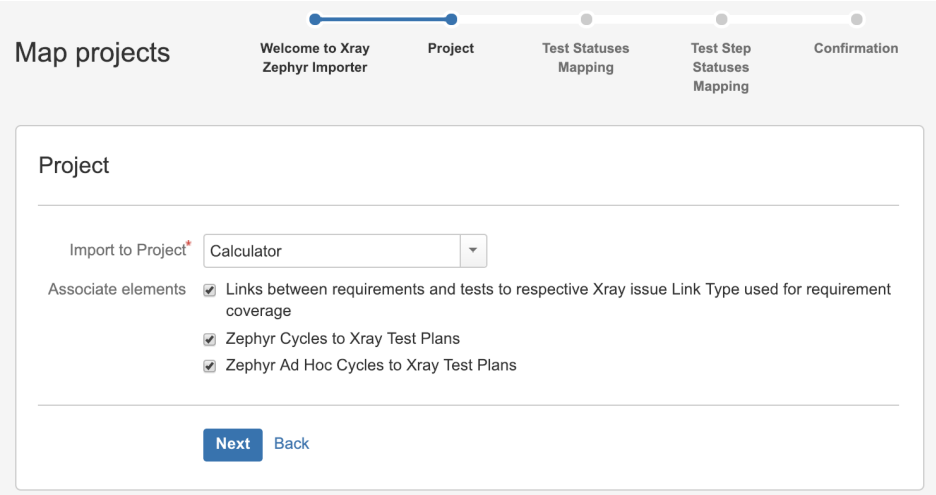


Then, you need to choose the project where to perform the migration; this will be same where the Xray entities will be created in.

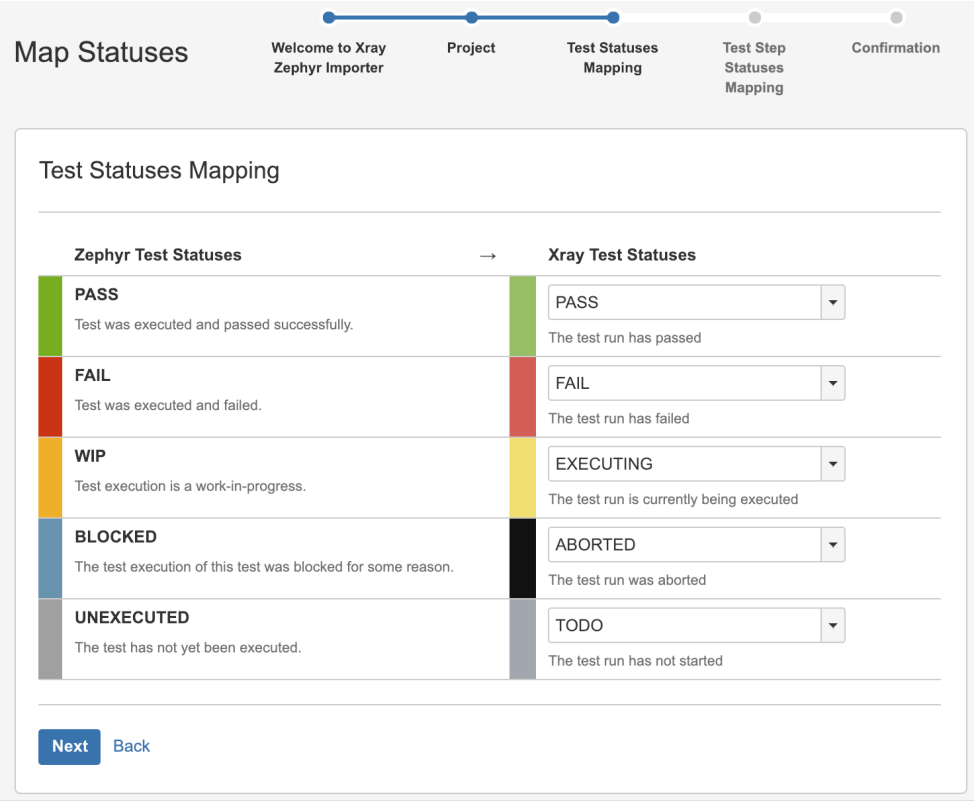
You may fine-tune the process by (un)checking some flags:

- **Links between requirements and tests to respective Xray Issue Link Type used for requirement coverage**: creates the link Xray uses for tracking coverage between Tests and requirements; by default Xray used the "tests" issue link type
- **Zephyr Cycles to Xray Test Plans**: creates Test Plans based on Zephyr Cycles
- **Zephyr Ad Hoc Cycles to Xray Test Plans**: creates Test Plans based on Zephyr Ad Hoc Cycles

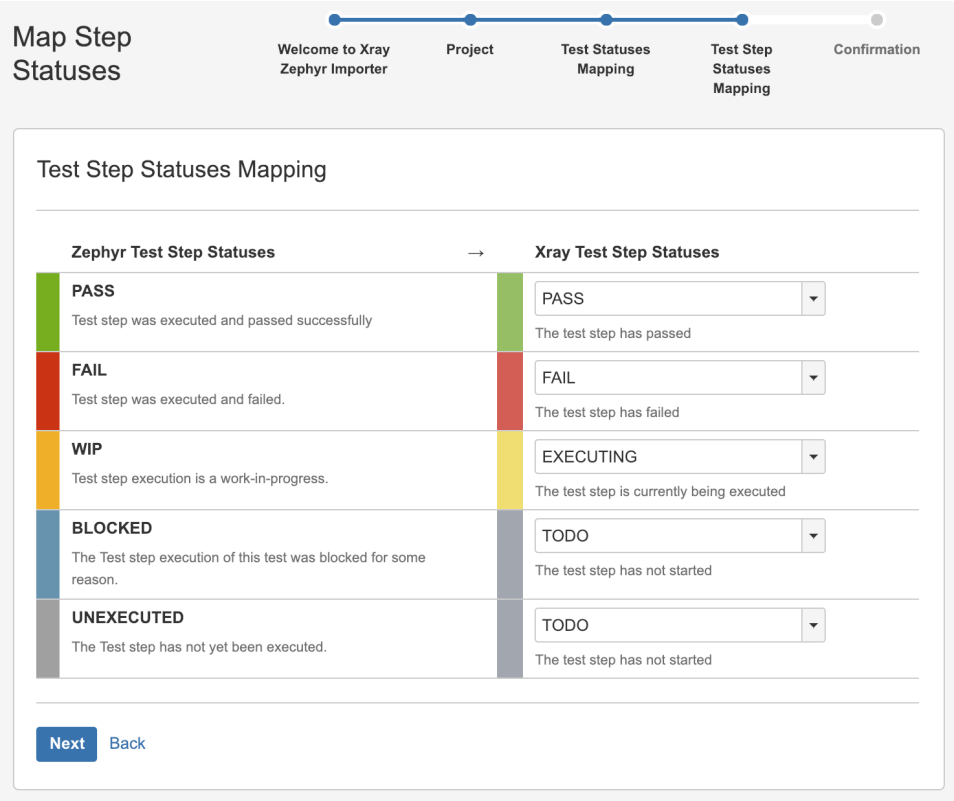
If **Zephyr Cycles to Xray Test Plans** and **Zephyr Ad Hoc Cycles to Xray Test Plans** are unchecked then no Test Plans will be created; nevertheless, Test Executions will always be created if Zephyr Executions exist.



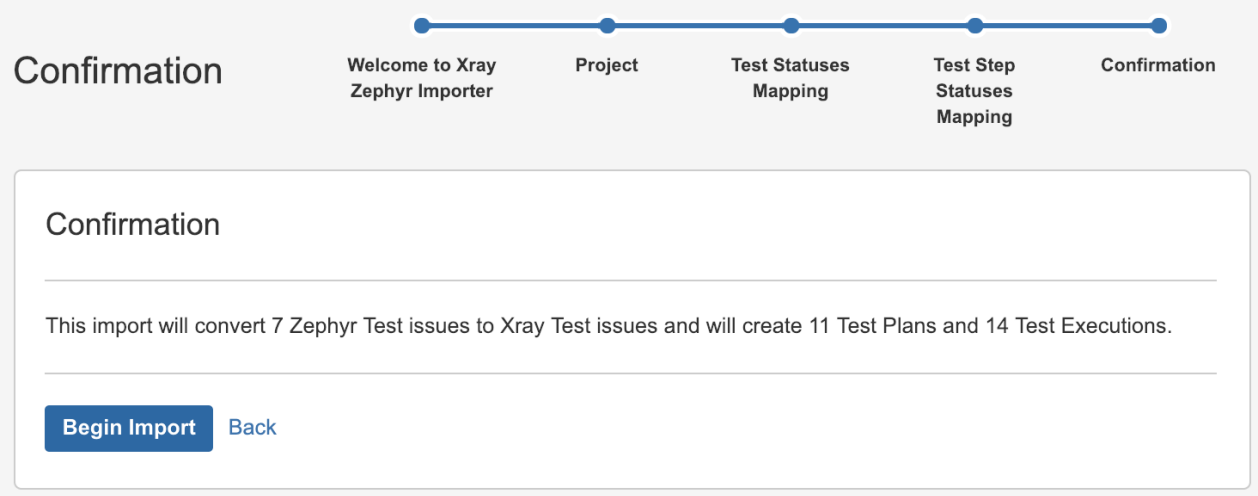
You need to map Zephyr's Test Statuses to Xray counterparts.



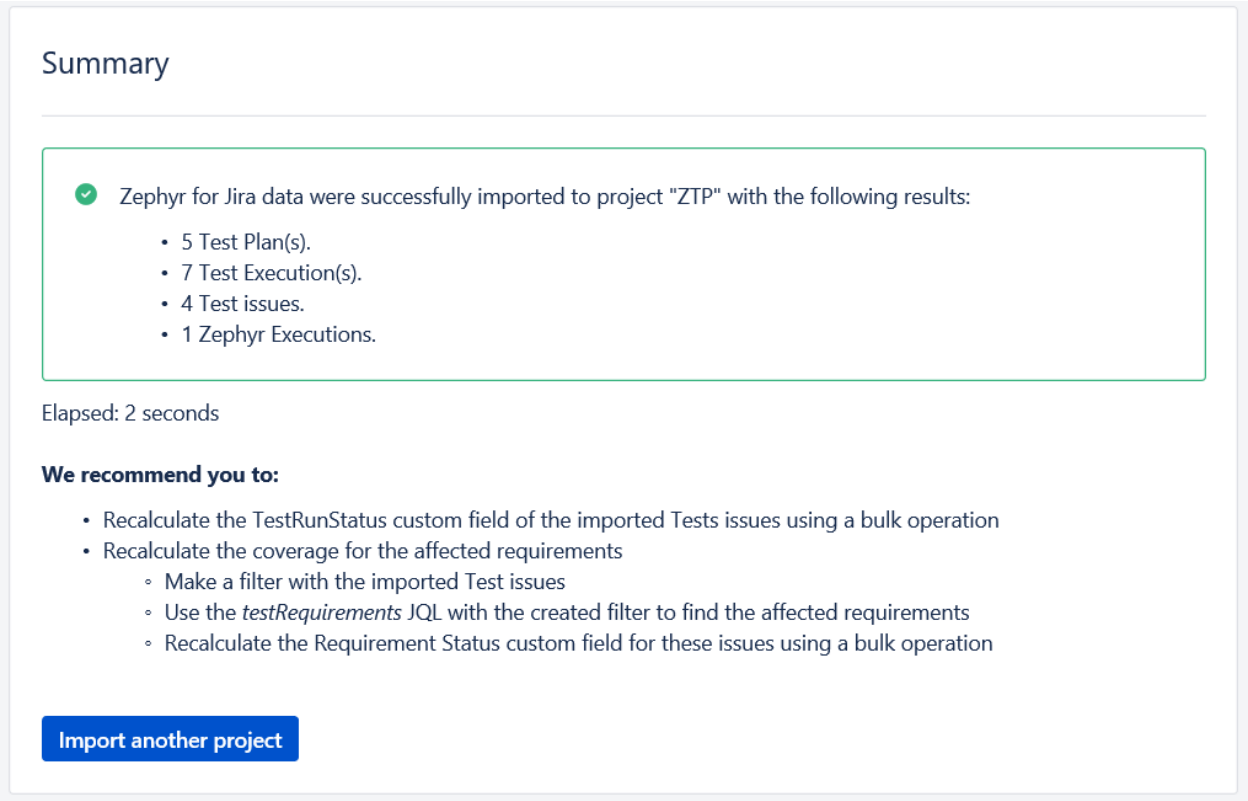
Likewise, you also need to map Zephyr's Test Step Statuses.



A final confirmation dialog presents information about the total number of Xray entities that will be created.



At the end, a brief summary is shown mentioning the total number of Xray issues created and any warnings that occurred during the process.

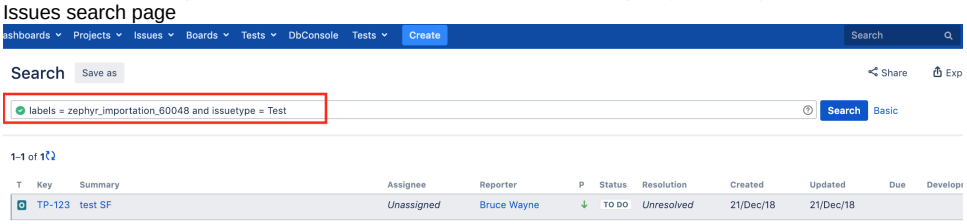


Please have a look at the following section for some additional steps before concluding the migration process.

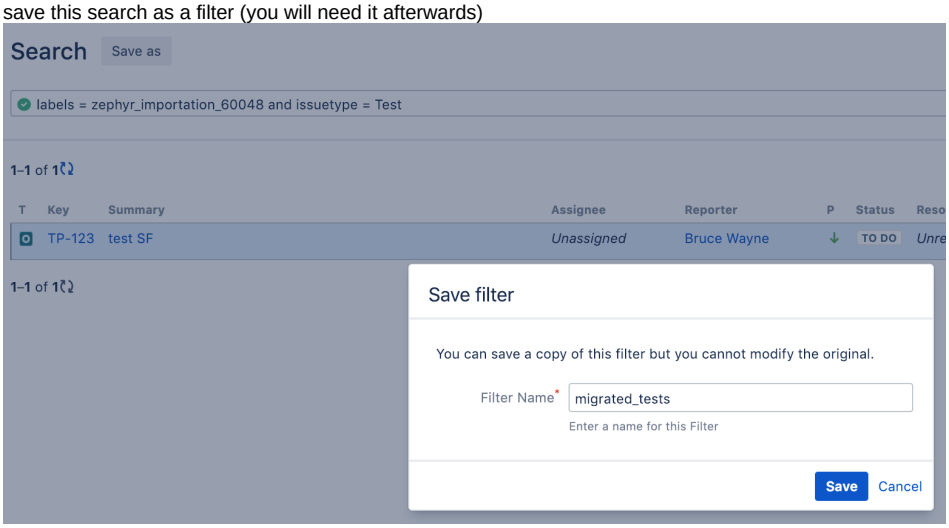
After doing the migration

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.

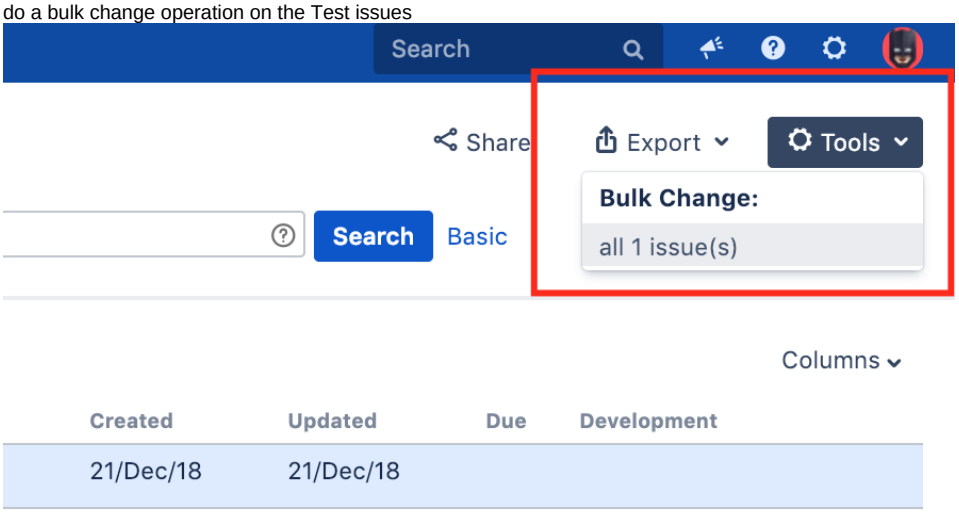
1. reset the "TestRunStatus" custom field of the migrated Tests
    a. you can use the link provided in the final screen mentioned earlier for quickly obtaining the created Tests; you will be redirected to the Issues search page



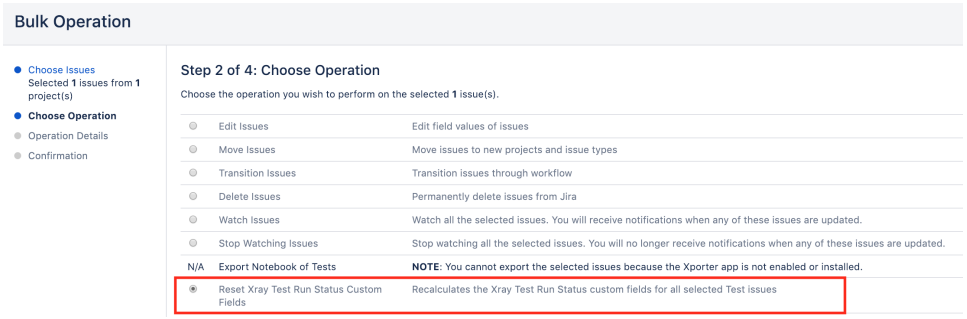
b. save this search as a filter (you will need it afterwards)



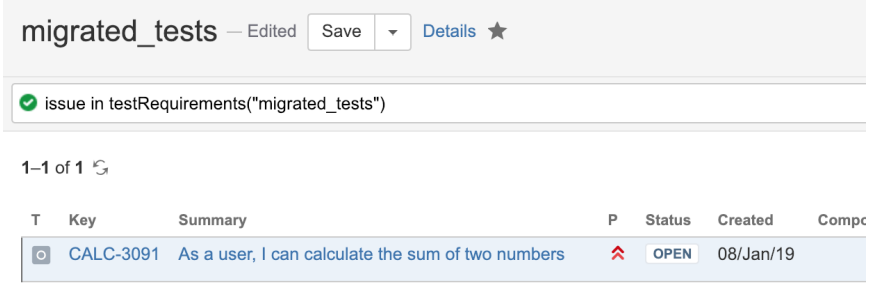
c. do a bulk change operation on the Test issues



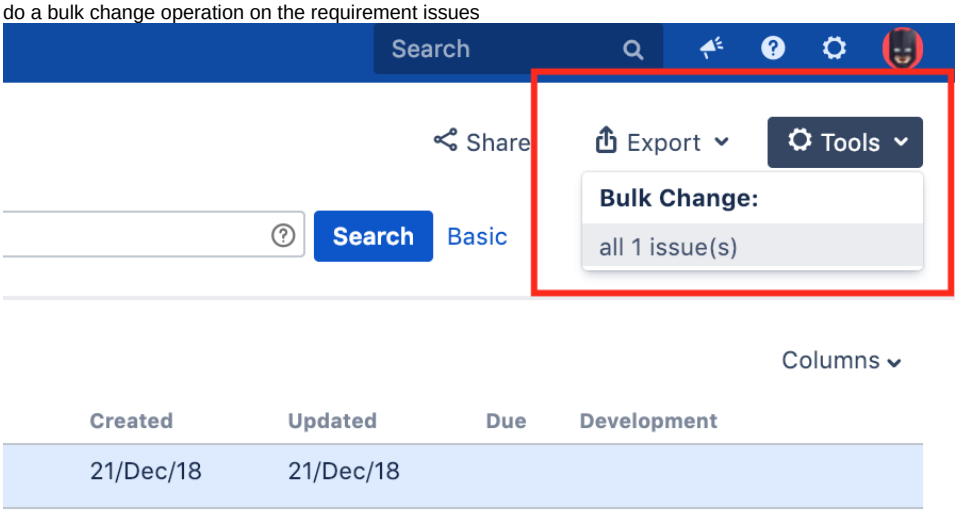
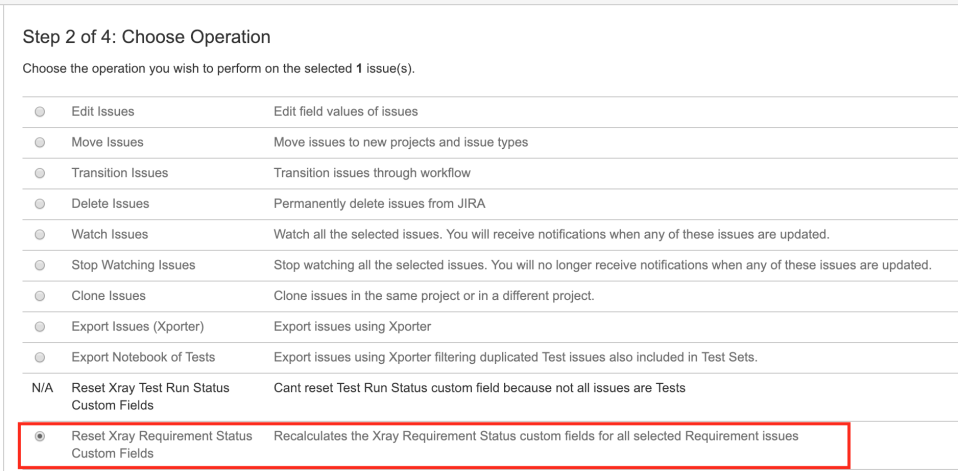
d. reset the TestRunStatus custom field



2. reset the "Requirement Status" custom field of the requirements linked to the migrated Tests
   a. use the testRequirements JQL function using use the name of the previously saved filter as argument



b. do a bulk change operation on the requirement issues



c. reset the Requirement Status custom field