

Xporter 6.9.0 Release Notes

12 November 2021

The Xporter for Jira team proudly announces the release of **Xporter 6.9.0**. This point release contains several new features and bug fixes.



If installing manually from the Atlassian Marketplace, please make sure you install the correct asset for your Jira version (i.e. 7.x, 8.x, datacenter) from the [releases page](#).

- [Highlights for this release](#)
 - [Parameterized Tests](#)
 - [Xray Custom Fields](#)
 - [Test Step Custom Fields](#)
 - [Test Run Custom Fields](#)
 - [Export Sub Test Execution](#)
- [New Features and Bug Fixes](#)

[Download the latest version](#)

Highlights for this release

Parameterized Tests

Test parameterization is a powerful practice that allows the same test to be executed multiple times with different parameters.

Parameters are similar to input values (variables) that can change with each execution.

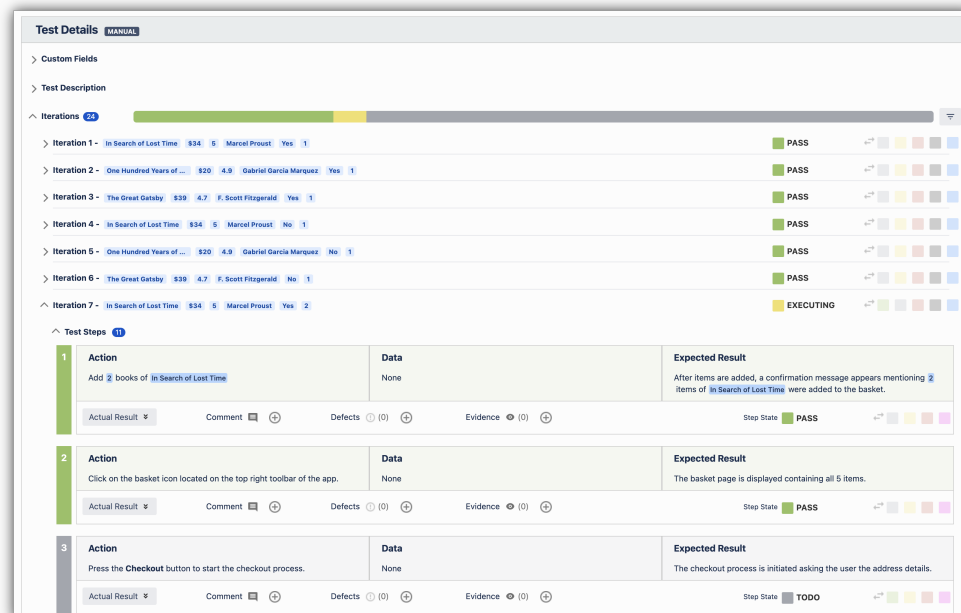
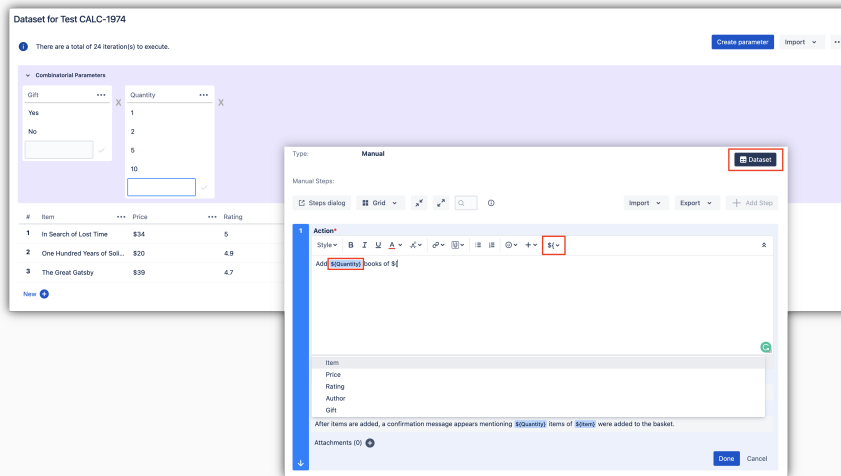
Parameterized tests in Xray are defined just like any other test with the addition of some parameter names within the specification using the following notation: **`${PARAMETER_NAME}`**.

All the iterations for a given test are executed within the context of the same test run. The step parameters will be replaced by the corresponding iteration values.

Xporter allows exporting Test Runs parameters and iterations directly from:

- Tests
- Test Executions

Please, check the Xray integration documentation [here](#)



Xray Custom Fields

Test Step Custom Fields

Xray allows configuring additional fields/columns for Test Step fields. This way you can fully customize the columns for your manual Test Steps hiding native fields, and providing new fields.
With this new version, Xporter allows exporting test steps custom fields using the same approach, which means, the placeholder with the custom field name.

```
Iterating over Test Steps Custom Fields:
#{for teststeps}
    ${TestSteps[n].Action}
    ${TestSteps[n].Data}
    ${TestSteps[n].ExpectedResult}
    ${TestSteps[n].DateCustomField}
#{end}

#Iterating over Test Runs
#{for testruns}
    #Iterating over Manual Test Steps from Test Run
    #{for r=TestRuns[n].TestStepsCount}
        DateCustomField: ${TestRuns[n].TestSteps[r].DateCustomField}
    #{end}
#{end}
```

Calculator / Test Execution: CALC-1974 / Test: CALC-869

Manual Test As a user, I can calculate the sum of 2 numbers

Dataset

Return to Test Execution

Custom Fields

TextRunCustomField

André Rodrigues

Test Description

Test Issue Links

Pre-Conditions

Test Steps

1	Action	Data	Expected Result
	press 1 button	Data from Step 1	the screen should present "1"
	DateCustomField		
	06/Oct/21		
	Actual Result: #	Comment	Defects
			Evidence
			Step Status: PASS

2	Action	Data	Expected Result
	press button identified in the data column	+	the screen should present "+"
	DateCustomField		
	25/Nov/21		
	Actual Result: #	Comment	Defects
			Evidence
			Step Status: PASS

3	Action	Data	Expected Result
	press the "equal" (=) button	Value returned = 3	the result should be: 3
	DateCustomField		
	01/Nov/21		

Test Run Custom Fields

It is now possible to export Test Run custom fields. These fields can be useful to add extra information to Test Runs, usually only available during or after executing Tests.

```
#Iterating over Test Runs
#{for testruns}
    ${TestRuns[n].TextRunCustomField}
#{end}
```



If your custom field type is a Number, Data, or Date Time you can use [formatting functions](#).



2. Test Run Details

TextRunCustomField: André Rodrigues

This Manual Test has 3 Steps

Step	Data	Expected Result	DateCustomField	Comment	Defects	Evidences	Status
1	press 1 button	Data from Step 1	the screen should present "1"	2021-10-06			PASS
2	press button identified in the data column	+	the screen should present "+"	2021-11-25			PASS
3	press the "equal" (=) button	Value returned = 3	the result should be: 3	2021-11-01			PASS

Export Sub Test Execution

Xporter with this new version allows users to export all the information from a Sub-Test Execution.

A Sub-Test Execution has exactly the same functionalities as the Test Execution issue type.

Xporter for Jira allows you to get the following data from the Xray Test Execution:

- Overall Execution Status
- Iterating Test Runs associated with a Test Execution
- Exporting Test Runs custom fields
- Iterating Pre-Conditions associated with a Test Run
- Iterating Attachments associated with a Test Run
- Iterating Evidences associated with a Test Run
- Iterating Defects associated with a Test Run
- Iterating Automated Test details associated with a Test Run
- Iterating Manual Test Step details associated with a Test Run
- Iterating Manual Test Step Attachments associated with a Test Run
- Iterating Manual Test Step Defects associated with a Test Run
- Iterating Manual Test Step Evidences associated with a Test Run
- Manual Test Step Custom Fields associated with a Test Run
- Exporting Test Run Activity
- Exporting Parameterized Tests from a Test Run

Please, check the Xray integration documentation [here](#)

New Features and Bug Fixes

Key	T	Summary	Resolution
XPORTER-3642		As a user, I should be able to use the "XrayTestExecAdvancedWithCoverPage.docx" template with Sub Test Executions	Done
XPORTER-3640		Export the the new test steps/Test Runs custom fields	Done
XPORTER-3368		Xray Templates for Test Plans with Post Functions will result in badly exported Templates	Done
XPORTER-3529		It's not possible to export the Actual Results field	Done
XPORTER-3576		The fullname:TestRuns[n].Assignee function does not obtain the correct information	Done
XPORTER-3552		Xray's Execution Evidences are not being rendered in Xporter Templates	Done

[6 issues](#)