Transition issues

This function has been **renamed** with the **JWT 3.0** release.

Find the new documentation at:

Transition issue

On this page

Purpose

Post-function "**Transition issues**" is used for ordering the **execution of a transition** in one or more issues. This feature is an alternative to **virtual fields** "**Issue status**", "**Issue status (delayed writing)**", "**Execute transition**" and "**Execute transition (delayed execution)**" which is the classic method for executing transitions in **Jira Workflow Toolbox**.

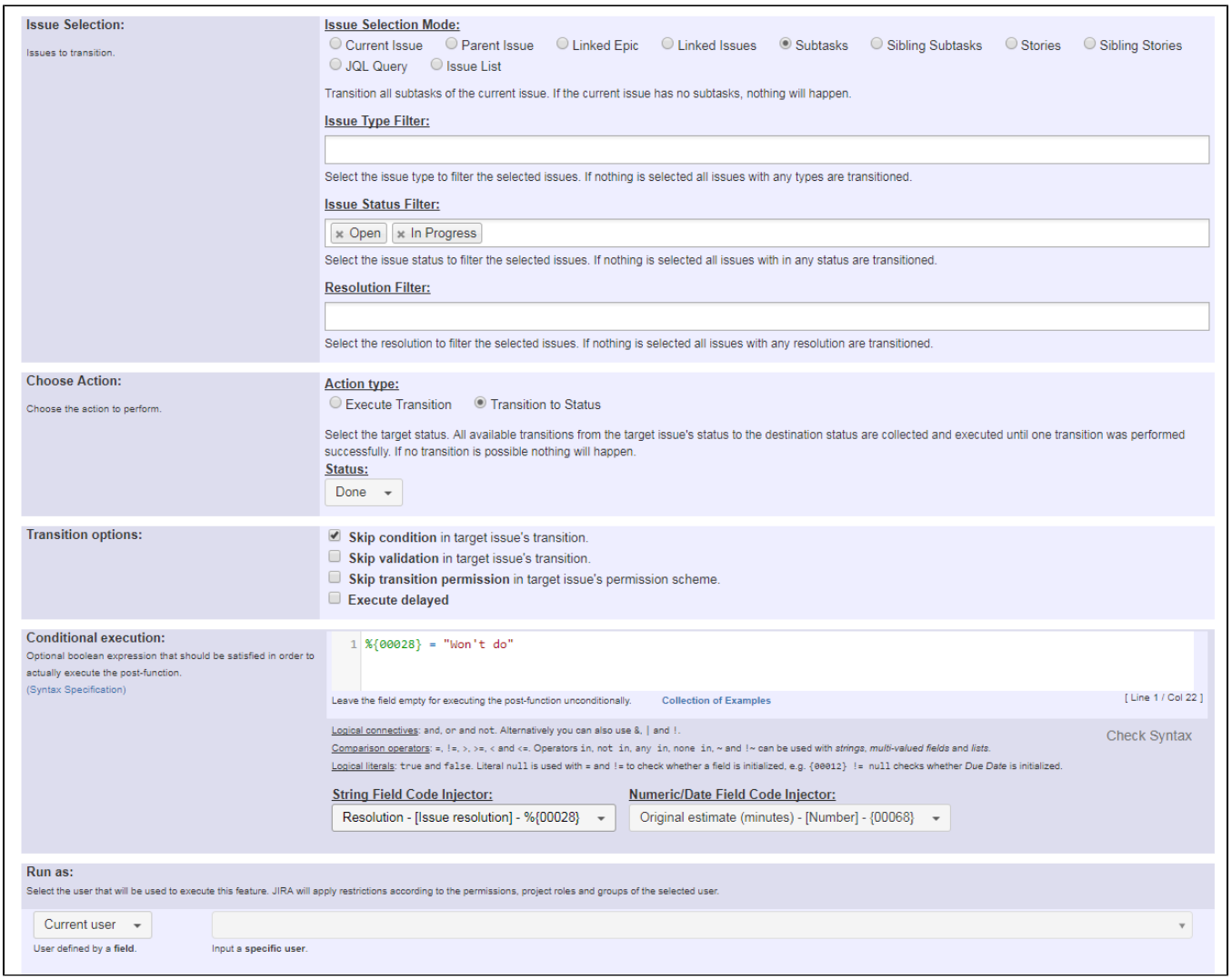
Post-function "**Transition issues**" uses the same transition manager as **virtual fields** above, but with the difference that instead of using the **names of statuses or transitions**, it uses their **internal ID**. This has the advantage that the renaming of transitions or statuses will not break your workflow, besides it's more intuitive than virtual field in most cases.

Anyway, **virtual fields** "**Issue status**", "**Issue status (delayed writing)**", "**Execute transition**" and "**Execute transition (delayed execution)**" keep working as always, and will keep being irreplaceable for some uses like automatic transitioning of issues newly created by post-function **Create issues and sub-tasks**.

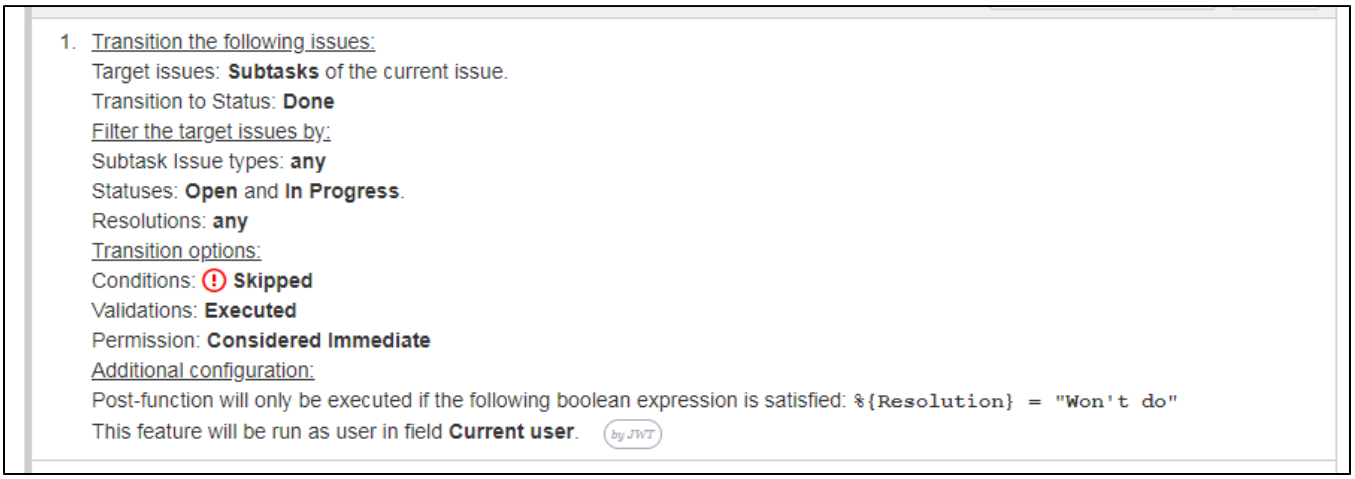
Available since **Jira Workflow Toolbox 2.3.0**.

Example: Transition open sub-tasks if the selected resolution of the current issue is set to "Won't do"

Select the **transition issue** post-function. The following screen shows the configuration page.



Once configured, the transition will look like this:



Configuration Parameters

Target Issue Selection Modes

There are 9 different modes for selecting the issues whose field values are going to be set: **Current issue**, **Parent Issue**, **Linked Epic**, **Linked Issues**, **Subtasks**, **Sibling Subtasks**, **Issues under Epic**, **Sibling issues under Epic**, **JQL query** and **Issue List** expression.

Current issue

If the current issue should be transitioned, a delay will be forced if nothing is specified in the Transition Options. The function must be placed before the Fire Event post function.

Parent Issue

The parent issue of the current issue. If the current issue is not a subtask, nothing will happen.

Linked Epic

Epic linked to current issue. If the current issue has no linked epic, nothing will happen.

Linked Issues

Issues linked to current issue. If current issue has no linked issues, nothing will happen.

The target issues can be filtered by:

- **Issue link type**: only for linked issues.
- **Issue types**: if no issue type is selected, then no filter by issue type is applied.
- **Statuses**: if no status is selected, then no filter by status is applied.
- **Resolutions**: if no resolution is selected, then no filter by resolution is applied.

Sub-tasks

Sub-tasks of current issue. If current issue has no sub-tasks, nothing will happen.

The target issues can be filtered by:

- **Sub-task Issue types**: if no sub-task issue type is selected, then no filter by sub-task issue type is applied.
- **Statuses**: if no status is selected, then no filter by status is applied.
- **Resolutions**: if no resolution is selected, then no filter by resolution is applied.

Sibling Sub-tasks

All sub-tasks sharing with same parent as current issue. If current issue isn't a sub-task, or the parent has no other sub-tasks, nothing will happen.

The target issues can be filtered by:

- **Sub-task Issue types**: if no sub-task issue type is selected, then no filter by sub-task issue type is applied.
- **Statuses**: if no status is selected, then no filter by status is applied.
- **Resolutions**: if no resolution is selected, then no filter by resolution is applied.

Issues under Epic

Issues under current issue, which is assumed to be an epic. If the current issue is not an epic issue, nothing will happen.

The target issues can be filtered by:

- **Issue types**: if no issue type is selected, then no filter by issue type is applied.
- **Statuses**: if no status is selected, then no filter by status is applied.
- **Resolutions**: if no resolution is selected, then no filter by resolution is applied.

Sibling issues under Epic

Issues under same epic as the current issue. If the current issue has no linked epic, nothing will happen.

The target issues can be filtered by:

- **Issue types**: if no issue type is selected, then no filter by issue type is applied.
- **Statuses**: if no status is selected, then no filter by status is applied.
- **Resolutions**: if no resolution is selected, then no filter by resolution is applied.

JQL Query

In this issue selection mode we use JQL, which is a language provided by Jira for doing **advanced issue searching**.

You can insert field codes with format **%{nnnnn}** in your JQL query. These field codes will be replaced with the values of the corresponding fields in current issue at execution time, and the resulting JQL query will be processed by Jira JQL Parser. This way you can write dynamic JQL queries that depend on values of fields of current issue. Example: **`issuetype = "%{00014}" AND project = "%{00018}"`** will return issues in same project and with same issue type as current issue.

When you write your JQL for selecting the issues, take into account the following advice:

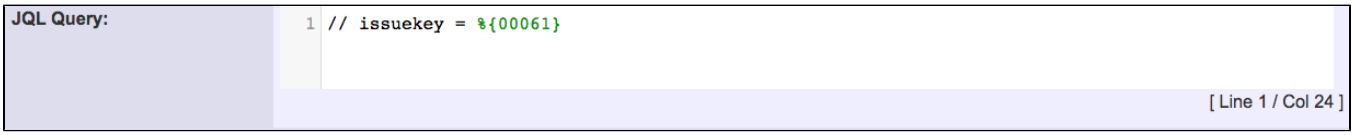
- If field values are expected to have white spaces or JQL reserved words or characters, you should write field code between quotes (double or simple). Example: **`summary ~ "%{00021}"`** will return issues with current user's full name. As full name can contain spaces, we have written the field code between double quotes.
- In general we will write field codes between quotation marks, since in most cases it doesn't hurt and it's useful for coping with field values containing white spaces or reserved JQL words. Anyway, there is an exception to this general rule: when our field contains a comma separated list of values, and we want to use it with JQL operator **IN**. In those cases we will not write the field code between quotes, since we want the content of the field to be processed as a list of values, not as a single string value.

  Example: Let's assume that "**Ephemeral string 1**" (field code **%{00061}**) contains a comma separate list of issue keys like "**CRM-1, HR-2, HR-3**". JQL Query: **`issuekey in ("%{00061}")`** will be rendered in run-time like **issuekey in ("CRM-1, HR-2, HR-3")**, which is syntactically incorrect. On the other hand, JQL Query: **`issuekey in (%{00061})`** will be rendered in run-time like **issuekey in (CRM-1, HR-2, HR-3)**, which is correct.

Disabling JQL Syntax Pre-Checking

When we enter our JQL query, a syntax pre-checking is carried out in order to verify that it's correctly written. But when we insert field codes in our JQL query, the definitive form of the query that will be executed is unknown, since it depends on the actual values of the fields in runtime. In these cases the syntax pre-checking is done with speculative values given to the fields, and it might happen that fake syntax errors are reported. In order to inhibit the JQL syntax pre-checking you should enter `//` at the beginning of the line. Those characters will be removed in the actual JQL query that will be executed.

Example:

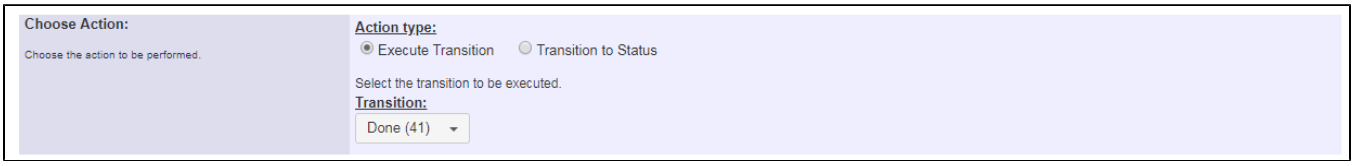


Issue List expression

In this issue selection mode we use an issue list expression according to the **syntax of the Expression Parser**. Here you can find **Examples of Issue List expressions**.

Action to be performed

Example:



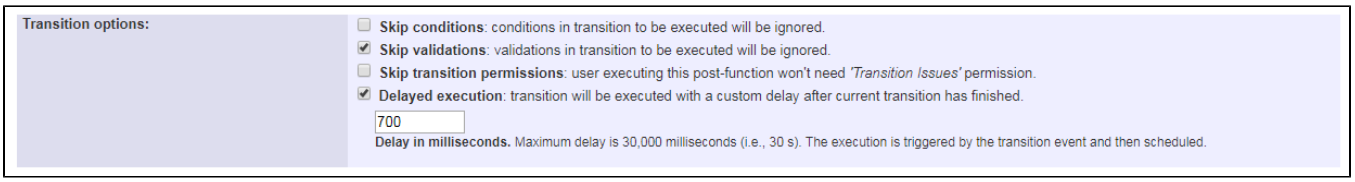
Execute transition

This will execute the specified transition in the target issues.

Transition to Status

Select a status the target issues should reach. Available transitions from the current status to selected destination status are retrieved, and sequentially tried out until one of them is executed successfully. If no transition to selected destination status is available, nothing will happen.

Transition Options



Skip Conditions

The conditions in the transitions to be executed will be ignored

Skip Validations

The validations in the transitions to be executed will be ignored

Skip transition permissions

The user executing this post-function won't need *'Transition Issues'* permission in the target issues project.

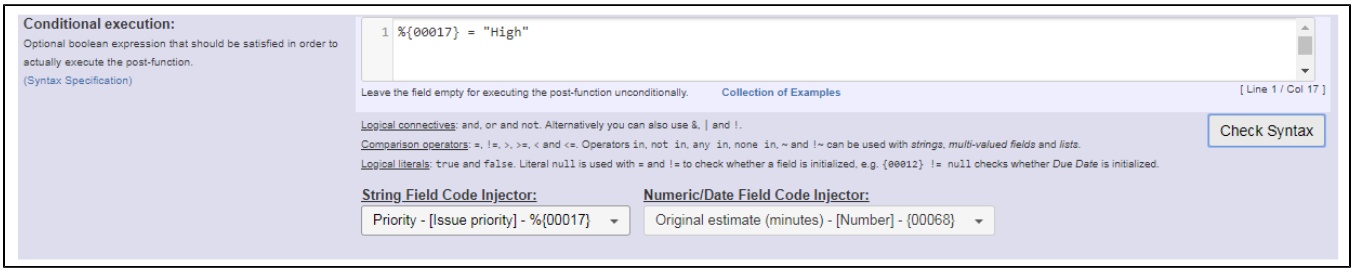
Delayed execution

The transition will be executed with a custom delay after current transition has finished.

The post-function must be placed before the 'Fire Event' post-function if a custom delay is specified.

Conditional Execution

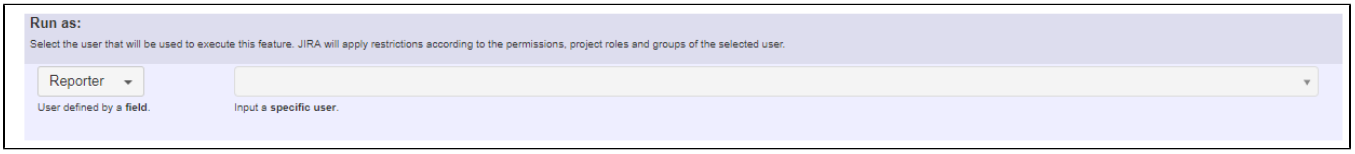
Example:



The post-function will be executed only when the boolean expression entered in this parameter is true, otherwise nothing will happen. You can make your boolean expression depend on the values of one or more fields, issue links, sub-tasks, etc. Use the syntax defined by the **Expression Parser**.

Run as

Example:



The post-function is going to be executed as the specified user. This parameter can be set to a **fixed user** (e.g. "john.nash"), or to a **user field** (e.g. "Reporter", "Assignee", etc). This parameter is important when we have permission or security restrictions that might prevent fields from being read or written.

Usage Examples

Page: Automatically close resolved sub-tasks when parent issue is closed
Page: Automatically close resolved sub-tasks when parent issue is closed (Transition issues)

Related Features

Page: Change parent's status depending on sub-task's summary
Page: Change parent's status depending on sub-task's summary (Transition issues)
Page: Moving story to "In Progress" when one of its sub-tasks is moved to "In Progress"
Page: Moving story to "In Progress" when one of its sub-tasks is moved to "In Progress" (Transition issues)
Page: Moving story to "Ready for QA" once all its sub-tasks are in "Ready for QA" status
Page: Moving story to "Ready for QA" once all its sub-tasks are in "Ready for QA" status (Transition issues)
Page: Moving sub-tasks to "Open" status when parent issue moves to "In Progress"
Page: Moving sub-tasks to "Open" status when parent issue moves to "In Progress" (Transition issues)
Page: Transition only a sub-task among several ones
Page: Transition only a sub-task among several ones (Transition issues)
Page: Transition sub-tasks when parent is transitioned
Page: Transition sub-tasks when parent is transitioned (Transition issues)
Page: Triage Jira Service Desk email requests (Move issues)