

Transition issue

The **Transition issue** post function automatically executes a **specific transition** on **one ore multiple issues** or transitions the issues **to the selected status**. Both, transition and target status, have to be available from the issue's current status.



Configuration

Target issue

The issues to be transitioned. If a list of issue keys is passed, the transition will be performed for every issue in the list. The following options are available:

- Current issue
- Parent issue
- Parser expression

When the option **parser expression** is selected, an expression parser input field is displayed. The input has to be an **issue key** or an **issue list**. For every issue in the given issue list, a transtion will be executed.

Mode

Select the transition to be executed or the status to be transitioned to. You can choose between to options:

- Execute transition:

Pick either a transition which is shown by name and grouped by workflows or provide a parser expression. In case of a parser expression a **transition ID** is expected.

- Transition to status:

Pick either a status that is shown by name or provide a parser expression. In case of a parser expression a **status ID** is expected.

If more than one transition is available to reach the configured status, the first found transition will be executed.

Delayed execution

Executing the post function will be delayed by the specified value in milliseconds. Please note that you are not able to define a value that is greater than 120 seconds.

Run as

Select the user that will be used to execute the post function. By default, it is set to the current user that executes the transition.

The following options are available:

Option	Description
Selected user	Select a specific Jira user.
User in field	Select the field containing the user that will be used to execute the post function.

The configured user must have all necessary permissions to transition the target issue.

Conditional execution

You can **optionally** specify a [logical expression](#) or a [Jira expression](#) depending on the chosen [Parsing mode](#) to define the circumstances (or conditions) under which the post function should be executed.

The result of the logical expression must return a boolean value of either:

- **true** the post function will be executed
- **false** the post function will **not** be executed

Using the **conditional operator**, even complex or multi-layered conditions can be constructed.

Make sure to learn more about defining logical expressions and browse through the various **examples** here: [Logical mode](#)



Use cases and examples

Use case	JWT feature	Workflow function	Parser functions	Use case description	Complexity
Auto-transition when related issues are in a specific status		Transition issue	linkedIssues()	Automatically transition issue , if all linked issues are in specific status .	INTERMEDIATE
Escalate an issue if it is being raised with a "Blocker" priority		Transition issue		When an issue is raised with a " Blocker " priority, a transition escalating the issue will be executed immediately after its creation .	BEGINNER
Fast-track transition issues assigned to the project lead		Transition issue		Automatically execute the " Start progress " transition if the issue's assignee is the project lead .	BEGINNER
Start progress on an issue immediately after creation		Transition issue		The transition to start progress on the current issue will be executed directly after its creation.	BEGINNER
Start progress on the parent issue		Transition issue		When a sub-task is transitioned to the " In Progress " status the parent issue will be transitioned to the " In Progress " status as well if it is still in the " To Do " status.	BEGINNER
Transition parent issue to another status		Transition issue		Transition the parent issue to the " Open " status.	BEGINNER

If you still have questions, feel free to refer to our [support](#) team.