

# Move issue

This post function allows you to **move** a **standard issue type** to other projects, statuses or issue types.

This post function must be the **last post function** of a transition.



## Configuration

### Project

Select the project which the issue will be moved to by selecting one of the following options:

Option	Description
Retain the project	The project will not be updated. This option makes sense, if you want to move an issue to an issue to other projects, statuses or issue types.
Selected project	Select the target project from the list of available projects.
Set project manually (parser expression)	Specify the target project by an expression using the <a href="#">Advanced text mode</a> .  The expression must return a <b>project key</b> or <b>project name</b> .

### Issue type

Select the issue type which the issue will be moved to by selecting one of the following options:

Option	Description
Retain the issue type	The issue type will not be updated.
Selected issue type	Select the target issue type from the list of available issue types.
Set issue type manually (parser expression)	Specify the target issue type by an expression using the <a href="#">Advanced text mode</a> .  The expression must return an <b>issue type name</b> .

### Status

Select the status which the issue will be moved to by selecting one of the following options:

Option	Description
Retain the status	The status will not be updated.
Selected status	Select the target status from the list of available statuses.

Set status manually (parser expression)	Specify the target status by an expression using the <a href="#">Advanced text mode</a> .  The expression must return <b>status name</b> or field code reference like <a href="#">Issue status</a> .
---	--

## Fields

Optionally define **fields** and **target values** that should be updated as part of the move operation.

Upon selecting a field that **should be updated** from the dropdown you will be presented with these options:

Option	Description
Automatic	This option is <b>only available</b> for the field <b>Assignee</b> .
Selected value	Select a specific value from the dropdown list.
Copy field value from current issue	Copy the value from a field in the current issue. Select the field you want to copy the value from.
Copy field value from parent of new issue	This option is <b>only available</b> if you create a sub-task under a new parent. <b>Only available</b> if the issue to be created is a sub-task.
Set field value manually (parser expression)	Select the field to copy from using an expression.  Learn more about the possibilities of the <a href="#">JWT expression editor</a> .
Clear field value	Clear the field value.

## Write protection

Check the **write protection** option if you want to prevent a field to be updated, if it already has a value. If checked, only empty fields will be set.

By default, all other field values will be retained.

## Additional options

You have the following two options:

Option	Description
Block transition and show an error message when the move operation fails	In case the move operation fails, e.g. because the specified issue type is <b>not available in the target project</b> , the entire transition will be <b>aborted</b> and an error message is shown to the user.
Delayed execution	The execution of this post function will be delayed by the value specified in <b>milliseconds</b> .  This parameter is useful, when multiple operations have to be performed in a single transition. The delayed execution ensures that all <b>previous operations have been completed</b> , e.g. field values that are referenced in this post function are present.  The maximum delay is 60,000 milliseconds (60 seconds).

## Conditional execution

You can **optionally** specify a [logical expression](#) 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](#)

## Run as

Select which **user** will be used to execute this post function. By default this parameter is set to the **current user**. You can also use field codes to run the function as a dynamic user (e.g. current assignee).

Make sure that the user running the post function has all the **relevant permissions** to perform the actions defined in the configuration (e.g. "Update Issues")!

If you want to keep track the actions being performed automatically, we suggest to create a **dedicated JWT account**, granted all relevant **permissions**, and use it in the Run as parameter to identify which changes have been made with JWT.

In case the configuration is **incorrect**, e.g. the issue type is not available in the target project or the target status is not available for the new issue type, the post function will be **skipped**.

After the execution of the post function you might need to **refresh your browser** in order to view the the issue fully updated to the new project and issue type.

Moving individual **sub-tasks** is **currently not supported**. Sub-tasks can only be moved in sync with their parent issue.

When moving a parent issue, make sure that valid configurations (issue type and workflow statuses) for all sub-tasks issues exist in the target project.



## Use cases and examples

Use case	JWT feature	Workflow function	Parser functions	Label
<a href="#">Move an issue to another issue type</a>		<a href="#">Move issue</a>		
<a href="#">Move an issue to another project</a>		<a href="#">Move issue</a>		

---

Triage issues created by email



Move issue

toUpperCase()  
toString()  
findPattern()  
isInGroup()

STAFF PICK

---

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