

Update linked issue or sub-task

This post function is used to update **one** or **multiple issue fields** in **linked issues** or **sub-tasks**.

By default, **only linked issues** will be updated.

If you want to also update **(sibling) sub-tasks**, the specific **checkbox** in the "Additional options" section has to be **checked**.



Configuration

Fields

Every supported field can either be set or cleared.

After selecting a field and clicking the **Add** button, you can select in the **popup** how to set the value. There are three or - when dealing with selectable fields - four options:

- Copy field from current issue
- Clear field value
- Set field value manually - Read more about the [JWT expression editor](#)
- The option **Selected value** is available for the following fields types:
 - Jira Software related fields like **Sprint** or **Epic**
 - User related fields like **Assignee** or **Reporter**
 - Version related fields like **Affects-** or **Fix version/s**
 - Component/s
 - Labels
 - Priority
 - Resolution
 - Security level
 - and all option based custom fields that are supported

Filter

Use any of the following filters to **refine** which issues will be updated.

Filter by project

Filter your project by choosing from one of the following options:

Option	Description
Any project	This is the default. If selected, the projects the linked issues belong to do not matter .
Current project	If selected, only linked issues which belong to the same project as the current issue will be updated.
Any but current project	If selected, only linked issues which do not belong to the same project as the current issue will be updated.

Filter by link type

Only issues linked to the current issue by the selected **link types** will be updated. This field is **required**.

Filter by issue type

Optionally limit the issues to be updated by their **issue type**.

Filter by status

Optionally limit the issues to be updated by their **status**.

Filter by field value

Optionally limit the issues to be updated by their **field values**. To achieve this you must enter an expression in [Logical mode](#). The expression must return `true` or `false`.

Examples:

Expression	Description
<pre>%{seed.issue.assignee} != null</pre>	This expression ensures that unassigned issues will not be updated . Values in linked issues or sub-tasks have to be referenced by their seed field codes.
<pre>%{issue.dueDate} <= %{seed.issue.dueDate}</pre>	This expression ensures that the due date of the linked issues is equal or later than current issue's due date. Values in linked issues or sub-tasks have to be referenced by their seed field codes.

The single filters are **AND concatenated**. In order for the operation to be successful and update the issues **all filter criteria must be met**.

Additional options

The following options are available:

Option	Description
Update sub-tasks matching the specified filter criteria	Check this option if you also want to update sub-tasks matching the specified filter criteria. This option only makes sense when the current issue itself is not a sub-task .
Update sibling sub-tasks matching the specified filter criteria	Check this option if you also want to update sibling sub-tasks matching the specified filter criteria. This option only makes sense when the current issue itself is a sub-task .

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.

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	Use case description	Complexity
Update the sub-task's assignees by the current issue's value		Update linked issue or sub-task	Update the sub-task's assignee from its parent	BEGINNER
Set the fix version from the current sub-task for all sibling sub-tasks		Update linked issue or sub-task	Update the fix version of all sibling sub-tasks from the current issue .	BEGINNER
Copy the due date from the current issue to blocking issues		Update linked issue or sub-task	Copy the due date value from the current issue to blocking issues .	BEGINNER

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