Evaluate custom fields

On this page

Use case

Evaluate custom fields of different types with Jira expressions.

1 Add a **Jira expression validator** or a **Jira expression condition** to the transition of your choice

2 **Fields**

Follow the instructions to evaluate any type of custom field depending on your preference.

For every example, the expression editor must be set to **Jira expression** and the **field code IDs** must be replaced.

Date Time Picker

1 **Jira expression***

```
new Date() < new Date(issue?.customfield_nnnnn)
```

The selected date and time is posterior to the current date and time.

Date Picker

1 **Jira expression***

```
new Date().toCalendarDate() < new CalendarDate(issue?.customfield_nnnnn)
```

The selected date must be posterior to the current date.

Select List (single choice)

1

Jira expression*

```
issue?.customfield_nnnnn?.value != null
```

The field must not be empty.

2

Jira expression*

```
issue?.customfield_nnnnn?.value == "Option 01"
```

The selected option must be a specific value.

3

Jira expression*

```
["Option 01", "Option 02"].includes(issue?.customfield_nnnnn?.value)
```

The field must contain one of two specific values.

Select List (multiple choices) and Checkbox

1

Jira expression*

```
issue?.customfield_nnnnn?.map(c => c.value).includes("Option 01")
```

One of the options values must be a specific value.

2

Jira expression*

```
issue?.customfield_nnnnn?.length > 2
```

More than two values must be selected.

3

Jira expression*

```
issue?.customfield_nnnnn?.every(c => c.value.match('Option 01'))
```

Only a specific value must be selected.

4

Jira expression*

```
issue?.customfield_nnnnn != null
```

The field must not be empty.

Select List (cascading)

1

Jira expression*

```
issue?.customfield_nnnnn?.value == "Option 01"
```

The parent value must be a specific value.

2

Jira expression*

```
issue?.customfield_nnnnn.child?.value == "One"
```

The child value must be a specific value.

3

Jira expression*

```
issue?.customfield_nnnnn?.value == null && issue?.customfield_nnnnn.child?.
value != null
```

The parent value must be empty and the child value must not be empty.

Short text and Paragraph

1

Jira expression*

```
issue?.customfield_nnnnn == "This is a text"
```

The field value must match exactly a text.

Jira expression *

```
issue?.customfield_nnnnn.includes("is a text")
```

The field value must contain a text.

Related examples

Use case	JWT feature	Workflow function	Use case description	Complexity
Block a transition based on issue links		Jira expression condition Jira expression validator	Evaluate **issue links** and hide transitions based on the outcome. This use case is valid for both conditions and validators . The only difference is that you can specify an additional **error message** when using a validator.	ADVANCED
Block a transition based on sprint information		Jira expression condition Jira expression validator	Make sure that an issue is **not** in an **active sprint**. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Block a transition based on the day of the week		Jira expression condition Jira expression validator	Block transitions on weekends or any other day of the week. This use case is valid for both conditions and validators . The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Block a transition if a predefined field value has not been changed		Jira expression condition Jira expression validator Update fields	Evaluate **a Date Picker field and block the transition if it has not been updated**. This use case is valid for both conditions and validators . The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Block a transition if some issues under an epic are not in a certain status		Jira expression condition Jira expression validator	Check **whether an epic** has **all issues under it in a certain status**. This is particularly important if you want to **block an epic** as long as work is still being done on related sub-tasks. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	INTERMEDIATE

Check current issue status		Jira expression condition Jira expression validator	Check **whether the current issue** is in a **particular status**. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Check for unresolved sub-tasks		Jira expression condition Jira expression validator	Check **whether the current issue** has any **unresolved sub-tasks**. This is particularly important if you want to **block a parent issue** as long as work is still being done on related sub-tasks. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	INTERMEDIATE
Check if an attachment was added recently		Jira expression condition Jira expression validator	Make sure that the current user has **uploaded a attachment** during a definite period of time. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Check parent issue type		Jira expression condition Jira expression validator	Check whether the **parent of the current issue** is of a **certain issue type**. This is particularly important if you want to **reuse** a workflow for multiple sub-task issue types but only want a transition to be available if the sub-task belongs to a certain **user story** or a **bug**. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	INTERMEDIATE
Check the number of times that a field has changed		Jira expression condition Jira expression validator	Check the **number of times that a field has changed**. This use case is valid for both conditions and validators. The only difference is that you can specify an additional **error message** when using a validator.	INTERMEDIATE
Evaluate the Parent Link field		Jira expression condition Jira expression validator	Evaluate different values of the issue in the Parent Link field of the transitioned issue. This use case is valid for both conditions and validators . The only difference is that you can specify an additional **error message** when using a validator.	INTERMEDIATE
Evaluate worklogs in sub-tasks		Jira expression condition Jira expression validator	Evaluate if **work has been logged in a sub-task** to prevent transitioning the parent issue when no work has been logged. This use case is valid for both conditions and validators . The only difference is that you can specify an additional **error message** when using a validator.	BEGINNER
Validate an issue only if a comment is written during the transition		Jira expression validator	Evaluate the **comments** and hide transitions based on the outcome. This use case is only valid for validators as it involves making changes during a transition. An additional **error message** can be added.	BEGINNER

| Validate worklogs | Jira expression condition | Evaluate if **a user has logged more than a certain amount of time** in the latest worklog. |
| | Jira expression validator | |

If you still have questions, feel free to refer to our support team.