Evaluate worklogs in sub-tasks

On this page

Use case

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.

Configuration

(1)

Jira expression*

```
issue?.subtasks.some(s => s?.timeSpent != null)
```

Any of the sub-tasks must have work logged for the issue to be transitioned.

(2)

Variations

(1)

Jira expression*

```
issue.timeSpent != null || issue?.subtasks.some(s => s?.timeSpent != null)
```

Either the parent issue or any of the sub-tasks must have work logged for the issue to be transitioned.

(2)

Jira expression*

```
issue?.subtasks.every(s => s?.timeSpent != null)
```

All of the sub-tasks must have work logged for the parent issue to be transitioned.

📁 Related use cases

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](#) [Jira expression validator](#)	Evaluate if **a user has logged more than a certain amount of time** in the latest worklog.	

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