

# Block a transition until all sub-tasks have certain fields populated

## On this page

- [Features used to implement the example](#)
- [Example: Block a transition until all subtasks have certain fields populated](#)
- [Other examples of that function](#)
- [Related Usage Examples](#)

## Features used to implement the example

- [Boolean validator with math, date-time or text-string terms](#)

## Example: Block a transition until all subtasks have certain fields populated

I have a parent issue (**issue type A**) and this issue could have sub-tasks (**issue type B**). The sub-tasks have two custom fields (one custom field is a user picker custom field, "**user**", and the other custom field is a date picker custom field, "**date**").

I want to allow certain workflow transition in "**issue A**" only if all their sub-tasks have a value on custom fields "**user**" and "**date**". Basically, I want to know if sub-tasks' custom fields are not empty as a condition of issue workflow transition for "**issue A**".

Using [Jira Workflow Toolbox](#) a **condition** or a **validation** can be implemented, so that you can prevent a transition from being executed while there are sub-tasks that do not have certain fields populated. Let us see an example:

To implement this example we are going to use [Boolean validator with math, date-time or text-string terms](#) with the following configuration:

**Boolean expression to be evaluated:**Syntax Specification

count(filterByFieldValue(filterByFieldValue(subtasks(), %{}00070), !=, ""), %{}00012), !=, "") = count(subtasks())

Logical connectives: **or**, **and** and **not**. Alternatively you can also use **|**, **&** and **!**.

Comparison operators: **=**, **!=**, **>**, **>=**, **<** and **<=**. Operators **~**, **!~**, **in**, **not in**, **any in** and **none in** can be used with **strings**, **multi-valued fields** and **lists**.

Logical literals: **true** and **false**. Literal **null** is used with **"=**" and **"!="** to check whether a field is initialized, e.g. **{00012} != null** checks whether **Due Date** is initialized.

**NUMERICAL AND DATE-TIME TERMS**

Numeric and Date-Time field values: insert field codes with format **{nnnnn}**.

Original estimate (minutes) – [Number] – {00068}

Insert Numeric Value

Valid date-time literal formats: **yyyy/MM/dd [hh:mm]** or **yyyy-MM-dd [hh:mm]**. Time literals use format: **hh:mm**.

There is a set of [mathematical functions](#) and [time macros and functions](#) available to be used in your expression.

**TEXT-STRING TERMS**

Text-String field values: insert field codes with format **%{nnnnn}** or **%{nnnnn.i}** for referencing levels in cascading select fields (*i* = 0 for base level).

Summary – [Text] – %{00000}

Insert String Value

String literals: written in **double quotes**, e.g., **"This is a string literal."**

String concatenation: use operator **+** to concatenate string values, e.g., **"The summary of issue with key " + %{00015} + " is \"" + %{00000} + "\"."**

Escape character: character **\** is used with characters **"**, **\**, **n**, **r**, **t**, **f** and **b** to invoke an alternative interpretation.

There is a set of [string functions](#) available to be used in your expression.

**Message to show when validation fails:**

All subtasks must have fields "Environment" and "Due date" populated.

Boolean expression used is:

```
count(filterByFieldValue(filterByFieldValue(subtasks(), %{00070}, !=, ""), %{00012}, !=, "")) = count(subtasks())
```

Note that:

- `%{00070}` is code for string value of "Environment"
- `%{00012}` is code for string value of "Due date"

This example, can be extended to any number of fields and custom fields, and can also be restricted to certain types of subtasks using function `filterByIssueType`.

---

## Other examples of that function

Page: [Block a transition until all sub-tasks have certain fields populated](#)  
Page: [Block an epic's transition depending on linked issues status and due date](#)  
Page: [Block or hide a transition for an issue depending on its issue links](#)  
Page: [Block or unblock a transition after an issue rested a specific time in a status](#)  
Page: [Block transition until all sub-tasks are in a specific status category](#)  
Page: [Close parent issue when all sub-tasks are closed](#)  
Page: [Enforce a field \(Select List\) to be set when another field \(Radio Button\) has a certain value \(works with any kind of field type\)](#)  
Page: [Ensure that all issues linked with a certain issue link type have "Due Date" field set](#)  
Page: [If field A is populated then, field B must also be populated](#)  
Page: [Limit issue creation per role and issue type](#)  
Page: [Limit the number of hours a user can log per day](#)  
Page: [Limit valid dates for work logs](#)  
Page: [Make "Time Spent" field required when there is no time logged in the issue](#)  
Page: [Make a custom field mandatory when priority is "Critical" or "Blocker" and issue type is "Incident"](#)  
Page: [Make attachment mandatory depending on the value of certain custom field](#)  
Page: [Make different fields mandatory depending on the value of a Select List custom field](#)  
Page: [Make linked issues, sub-tasks and JQL selected issues progress through its workflows](#)  
Page: [Make parent issue progress through its workflow](#)  
Page: [Prevent issue creation if another issue with same field value already exists](#)  
Page: [Reject duplicated file names in attachments](#)  
Page: [Require at least one sub-task in status "Resolved" or "Closed" when "Testing required" is selected in Check-Box custom field](#)  
Page: [Require issue link when resolving as duplicate](#)  
Page: [Restrict parent issue from closing if it has sub-tasks that were created during a given parent issue status](#)  
Page: [Restrict sub-task type creation depending on parent issue status](#)  
Page: [Restrict sub-task type creation depending on parent issue type](#)  
Page: [Set a condition in a global transition which only applies in a certain status](#)  
Page: [Validate a custom field "Story Points" has been given a value in Fibonacci sequence](#)  
Page: [Validate compatible values selection among dependent custom fields](#)  
Page: [Validate only issue links created in transition screen](#)  
Page: [Validate that multi-user picker custom field A does not contain any user in multi-user picker custom field B](#)  
Page: [Validation and condition based on time expressions](#)  
Page: [Validation based on the value of a date type project property](#)  
Page: [Validation on issue attachments](#)  
Page: [Validation on MIME types of issue attachments](#)  
Page: [Validation on sibling sub-tasks depending on issue type and status](#)  
Page: [Validation on the value of a Cascading Select field](#)

## Related Usage Examples

- [Validation on sibling sub-tasks depending on issue type and status](#)
  - [example](#)
  - [validator](#)
  - [sub-task](#)
  - [transition](#)
- [Block a transition until all sub-tasks have certain fields populated](#)
  - [example](#)
  - [condition](#)
  - [validator](#)
  - [sub-task](#)
  - [transition](#)
- [Transition sub-tasks when parent is transitioned](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Transition only a sub-task among several ones](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Moving sub-tasks to "Open" status when parent issue moves to "In Progress"](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Moving story to "Ready for QA" once all its sub-tasks are in "Ready for QA" status](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Moving story to "In Progress" when one of its sub-tasks is moved to "In Progress"](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Automatically close resolved sub-tasks when parent issue is closed](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Change parent's status depending on sub-task's summary](#)
  - [example](#)
  - [post-function](#)
  - [sub-task](#)
  - [transition](#)
  - [outdated](#)
- [Close parent issue when all sub-tasks are closed](#)

- example
  - condition
  - validator
  - post-function
  - sub-task
  - transition
- Proceed with a task only when all sub-tasks are completed
  - example
  - condition
  - validator
  - sub-task
  - transition
- Prevent transitioning when there is a blocking issue
  - example
  - validator
  - issue-links
  - sub-task
  - transition
- Transition parent issue only when certain issue sub-task types are done
  - example
  - validator
  - sub-task
  - transition
- Enforce certain type of sub-tasks to be "Resolved" to allow executing a transition
  - example
  - validator
  - sub-task
  - transition
- Restrict parent issue from closing if it has sub-tasks that were created during a given parent issue status
  - example
  - validator
  - post-function
  - sub-task
  - transition