

# Block or unblock a transition after an issue rested a specific time in a status

## On this page

- [Features used to implement the examples](#)
- [Example 1: Prevent a closed issue from being reopened after resting 1 week closed](#)
- [Example 2: Ensure that issues rest in certain status at least 24 hours](#)
- [Example 3: Enroute issues to different statuses depending on the time they rested in current status](#)
- [Other examples of that function](#)
- [Related Usage Examples](#)

## Features used to implement the examples

- [Boolean condition and validator with math, date-time or text-string terms](#)
- Virtual field **Current date and time**
- Virtual field **Date and time of latest status change**(available from version 2.1.22): subtracting it to **Current date and time** gives us the time current issue has rested in current status.
- Virtual field **Current user** and function **isInRole()**: used as backdoor for allowing transition being executed by certain roles, as **Administrator**.

## Example 1: Prevent a closed issue from being reopened after resting 1 week closed

We want to prevent a certain issue from being reopened after 7 days closed, unless user has role "**Administrator**" or "**Supervisor**".

We implement this use case by inserting a validator [Boolean validator with math, date-time or text-string terms](#) in transition "**Reopen Issue**" with the following configuration:

Boolean expression to be evaluated:
[ Line 1 / Col 135 ]
Syntax Specification and Examples
?

```

1  %{00016} = "Closed" AND {00057} - {00158} > 7 * {DAY} IMPLIES isInRole(%{00020}, "Administrator") OR isInRole(%{00020}, "Supervisor")

```

Logical connectives: **and**, **or** 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.

**Check Syntax**

### 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.

**Skip validation when:**  
Inhibit the validator under selected circumstances.

☐ Transition is triggered by a **bulk operation**.  
☐ Transition is triggered by a **JIRA Workflow Toolbox post-function**.  
☐ Current issue is being created by cloning. Only makes sense in *Create Issue* transition.

**Message to show when validation fails:**

Issues that have rested more than 7 days in 'Closed' status can only reopened by Administrators or Supervisors.

**Field code injector:**  
Summary - [Text] - %{00000}

Field codes with format **%{nnnnn}** can be inserted in the failure message and its translations, and they will be replaced with actual field values at runtime.

[Set translations for installed languages](#)

Boolean expression used in the example is:

```
%{00016} = "Closed" AND {00057} - {00158} > 7 * {DAY} IMPLIES isInRole(%{00020}, "Administrator") OR isInRole(%{00020}, "Supervisor")
```

We are using logical connective **IMPLIES** for clarity. An equivalent expression using only primitive logical connectives is:

```
%{00016} != "Closed" OR {00057} - {00158} < 7 * {DAY} OR isInRole(%{00020}, "Administrator") OR isInRole(%{00020}, "Supervisor")
```

Note that:

- **%{00016}** is field code for **Issue status**
- **{00057}** is code for numerical value of field **Current date and time**
- **{00158}** is code for numerical value of field **Date and time of latest status change** (available from version 2.1.22)
- **%{00020}** is field code for **Current user**

Once configured, transition **"Reopen Issue"** looks like this

DONE

Reopen

TO DO

Screen: None - it will happen instantly

Triggers 0

Conditions 1

Validators 1

Post Functions 7

The transition requires the following criteria to be valid

Add validator

Only if the following boolean expression is true: `{Issue status} = "Closed" AND {Current date and time} - {Date and time of latest status change} > 7 * {DAY} IMPLIES isInRole({Current user}, "Administrator") OR isInRole({Current user}, "Supervisor")`  
Message to show when validation fails: "Issues that have rested more than 7 days in 'Closed' status can only reopened by Administrators or Supervisors."

## Example 2: Ensure that issues rest in certain status at least 24 hours

As in the previous example, we allow users in "Administrator" or "Supervisor" project role to bypass this time restriction.

We insert **Boolean validator with math, date-time or text-string terms** in all the transitions with origin in the status we want to ensure 24 hours of permanence:

Boolean expression to be evaluated:

[ Line 1 / Col 112 ] Syntax Specification and Examples ?

```

1 {00057} - {00158} > 24 * {HOUR} IMPLIES isInRole({00020}, "Administrator") OR isInRole({00020}, "Supervisor")

```

**Logical connectives:** **and**, **or** 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.

**CHECK SYNTAX**

**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.

**Skip validation when:**  
Inhibit the validator under selected circumstances.

☐ Transition is triggered by a **bulk operation**.
 ☐ Transition is triggered by a **JIRA Workflow Toolbox post-function**.
 ☐ Current issue is being created by cloning. Only makes sense in *Create Issue* transition.

**Message to show when validation fails:**

Issue is in current status since %{00158}, and it should remain at least 24 h, this it can't yet leave the status.

**Field code injector:**  
 Summary - [Text] - %{00000} ▾

Field codes with format **%{nnnnn}** can be inserted in the failure message and its translations, and they will be replaced with actual field values at runtime.

[Set translations for installed languages](#)

Boolean expression used in the example is:

```
{00057} - {00158} > 24 * {HOUR} IMPLIES isInRole({00020}, "Administrator") OR isInRole({00020}, "Supervisor")
```

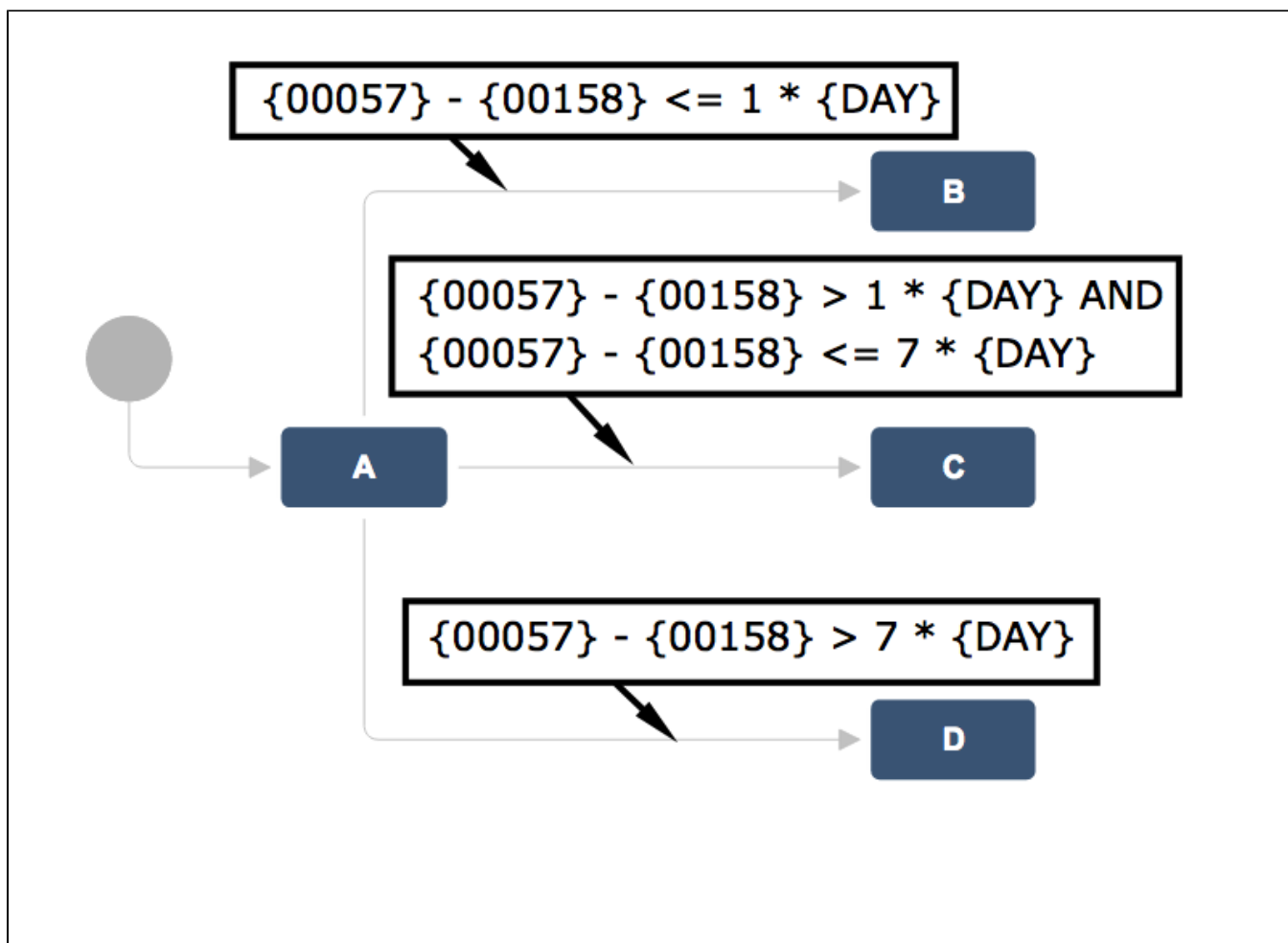
An equivalent expression is:

```
{00057} - {00158} <= 24 * {HOUR} OR isInRole({00020}, "Administrator") OR isInRole({00020}, "Supervisor")
```

## Example 3: Enroute issues to different statuses depending on the time they rested in current status

We have different transitions with origin in a certain status **A**, and with different destination statuses (**B**, **C** and **D**).

We use **Boolean condition with math, date-time or text-string terms** in each of the 3 transitions **A -> B**, **A -> C** and **A -> D** with different boolean expressions, in order to show in UI only one of these transitions depending on the time the issue has passed in status **A**, while hiding the other two transitions:



## 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

## Related Usage Examples

- Block or unblock a transition after an issue rested a specific time in a status
  - example
  - condition
  - validator
  - transition
- Block transition until all sub-tasks are in a specific status category
  - example
  - transition
  - condition
- Validation and condition based on time expressions
  - example
  - condition
  - validator
  - transition
- Set a condition in a global transition which only applies in a certain status
  - example
  - condition
  - transition
- 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

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

- condition
  - validator
  - sub-task
  - transition
- Block an epic's transition depending on linked issues status and due date
  - example
  - validator
  - issue-links
  - transition
- Moving story to "In Progress" when one of its sub-tasks is moved to "In Progress" (Transition issues)
  - example
  - post-function
  - 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
- Change parent's status depending on sub-task's summary (Transition issues)
  - example
  - post-function
  - transition
- Automatically close resolved sub-tasks when parent issue is closed
  - example
  - post-function
  - sub-task
  - transition
  - outdated