Close parent issue when all sub-tasks are closed

On this page

- Features used to implement the example
- Example: Close parent issue when all sub-tasks are closed
- Alternative implementation
- Other examples of that functions
- Related Usage Examples

Features used to implement the example

- **Condition and validation on sub-tasks**
- **Boolean condition and validator with math. date-time or text-string terms**
- **Copy parsed text to a field**

Example: Close parent issue when all sub-tasks are closed

We want to automatically **close parent** issue when the **all subtasks** has been **closed**. We are going to describe two different solutions to implement this behavior

This solution consist on a validator in parent's workflow, and a post-function in sub-tasks workflow:
A **validation** in **parent issue workflow** to prevent parent issue from being closed manually whenever there still are open sub-tasks. To do it we use **C ondition and validation on sub-tasks** in transition "**Done**" or "**Close Issue**" on **parent's workflow** with the following configuration:

Subtask's issue types:

- [ ] QA Sub-task
- [ ] Sub-task

Selected issue types will be allowed. Nevertheless, if you didn't select any issue type, and checked *"Allow unselected issue types"*, there won't be applied any filter by issue type, i.e., every issue type will be allowed.

Subtask's statuses:

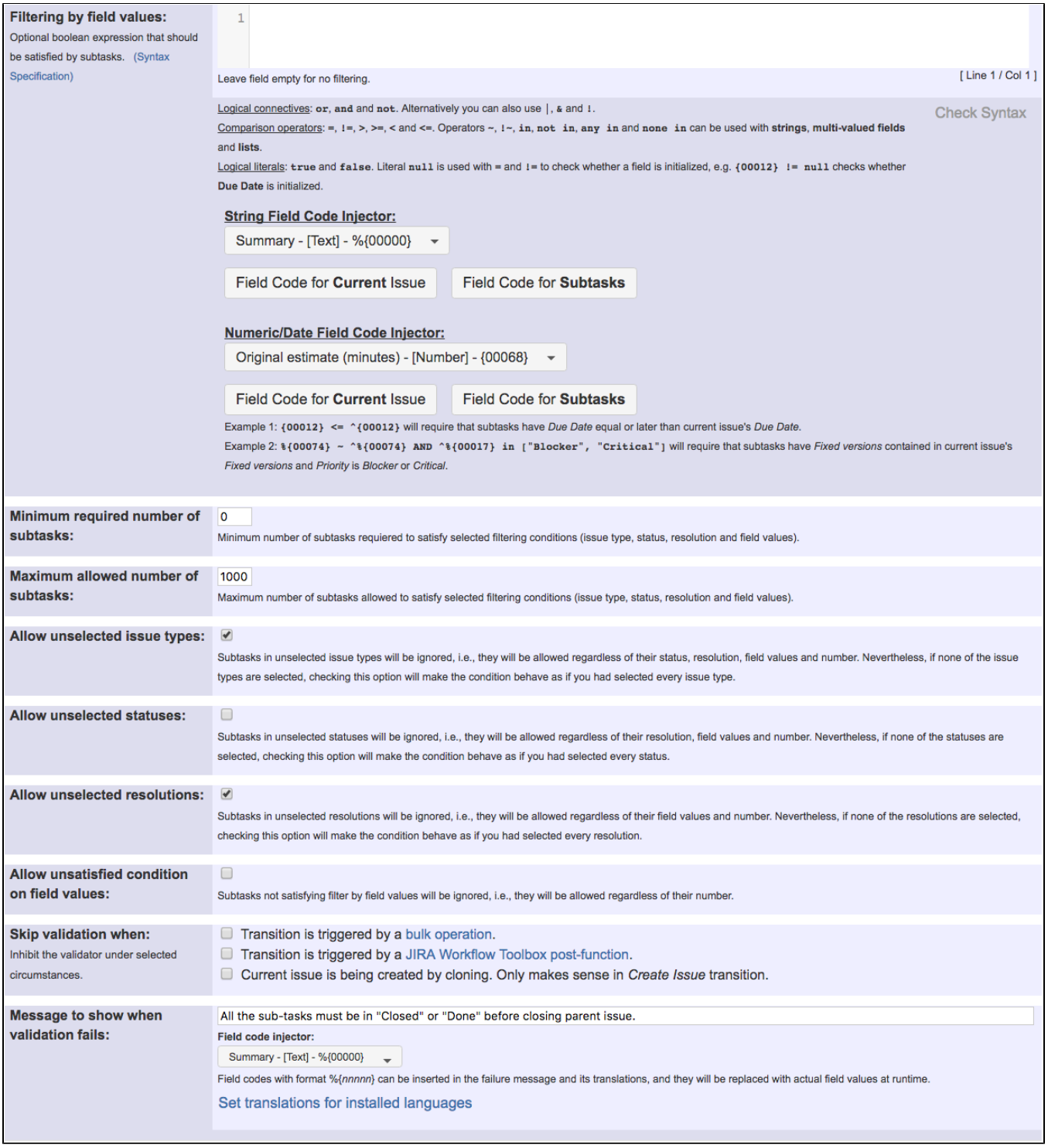
- [ ] Open
- [ ] In Progress
- [ ] Reopened
- [ ] Resolved
- [x] Closed
- [ ] To Do
- [x] Done
- [ ] Acceptance
- [ ] Fail
- [ ] Pass
- [ ] Retest
- [ ] Active
- [ ] Inactive

Subtasks in selected statuses will be allowed. Nevertheless, if you didn't select any issue status, and checked *"Allow unselected statuses"*, there won't be applied any filter by status, i.e., every issue status will be allowed.

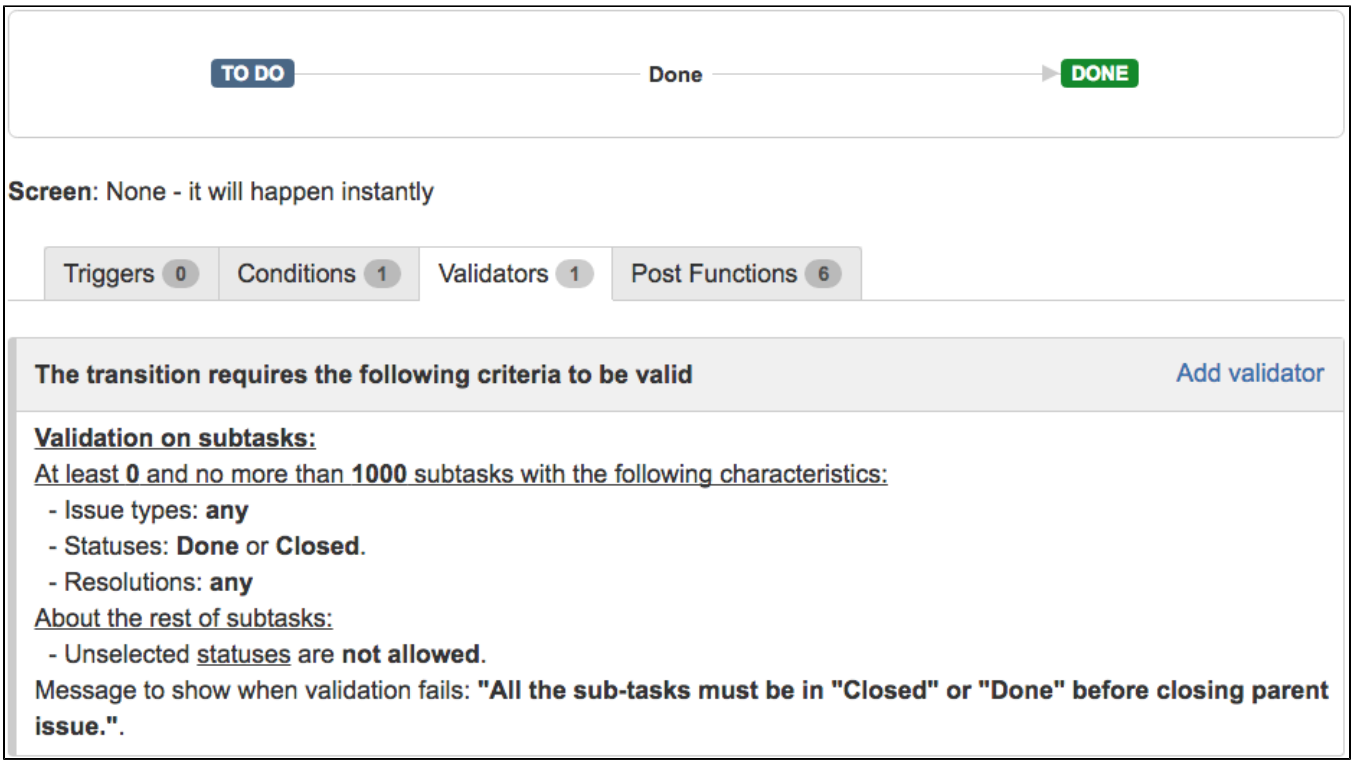
Subtask's resolutions:

- [ ] ***UNRESOLVED***, i.e. no resolution value
- [ ] Fixed
- [ ] Won't Fix
- [ ] Duplicate
- [ ] Incomplete
- [ ] Cannot Reproduce

Subtasks with selected resolutions will be allowed. Nevertheless, if you didn't select any issue resolution, and checked *"Allow unselected resolutions"*, there won't be applied any filter by resolution, i.e., every resolution will be allowed.



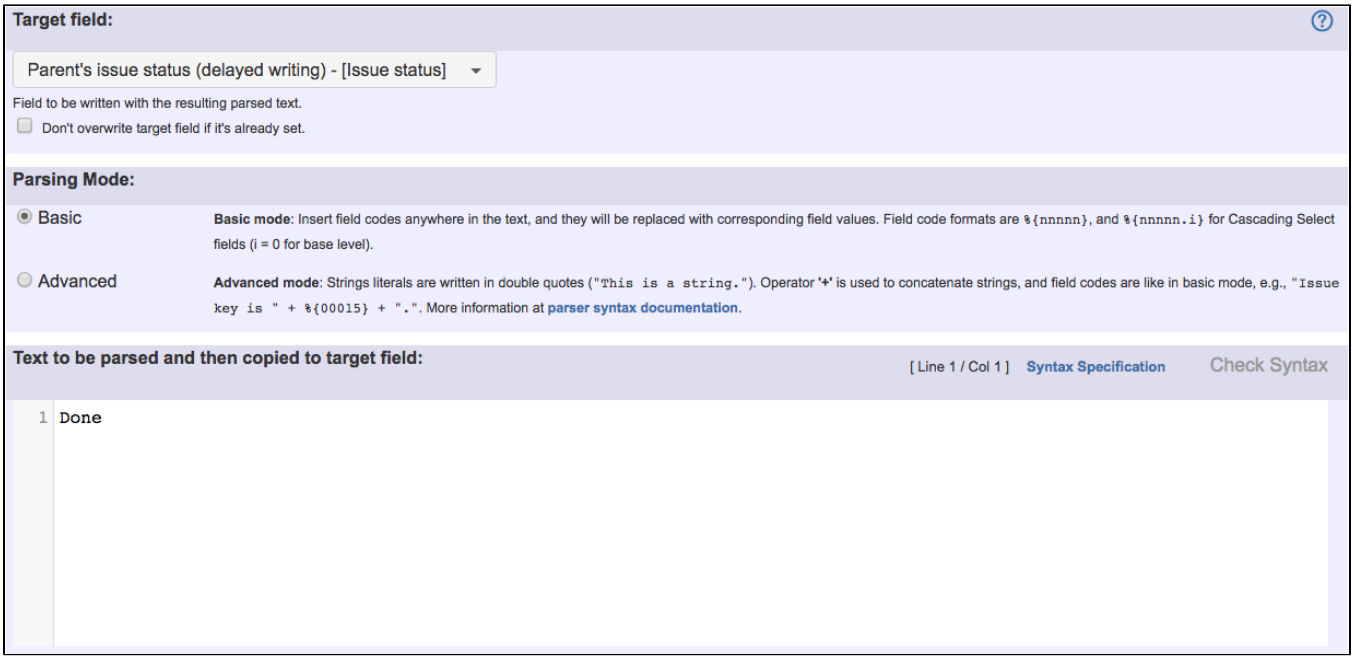
Once configured, transition "**Close Issue**" in parent's workflow will look like this:



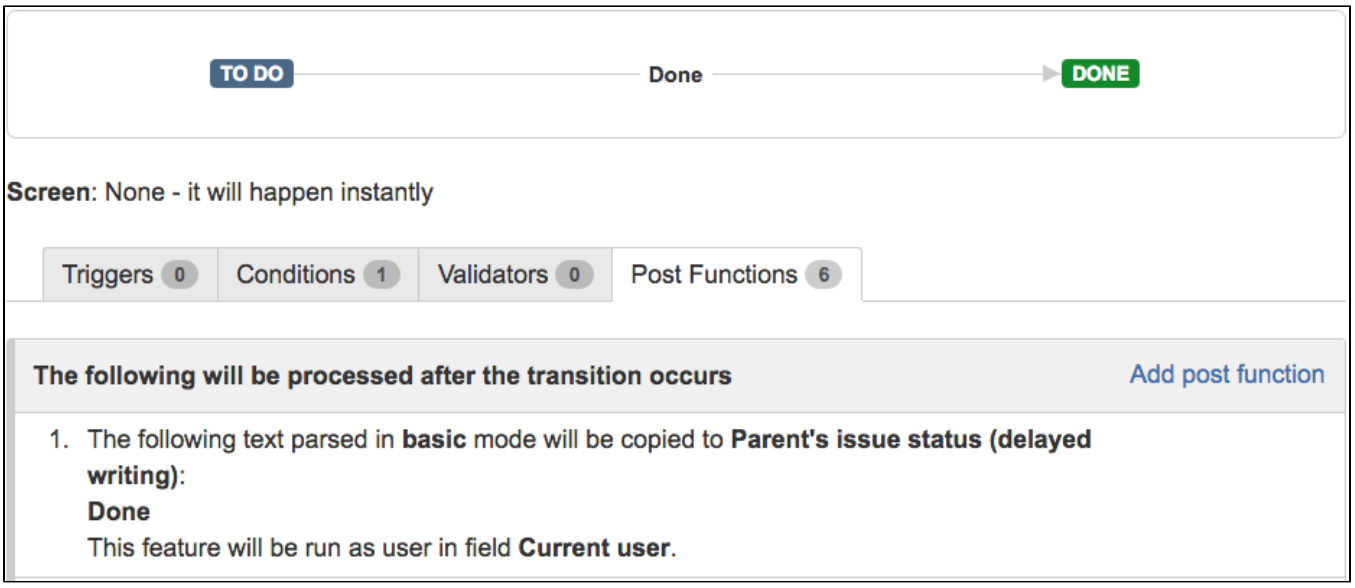
Alternatively, we can use **Boolean condition and validator with math. date-time or text-string terms** with the following boolean expression:

```
count(subtasks()) = count(filterByStatus(subtasks(), "Closed, Done"))
```

We add a **post-function** in **sub-task's workflow** for trying to close parent issue each time a sub-task is closed. To implement it we use post-function **Copy parsed text to a field** in transition "**Done**" or "**Close Issue**" on **sub-task's workflow** with the following configuration:



Once configured, transition "**Done**" or "**Close Issue**" in sub-task's workflow will look like this:



Note: If you are using Jira **7.0 or higher** with versions of **Jira Workflow Toolbox** older than **2.2.8**, you should input the following boolean expression in parameter **Conditional execution**:

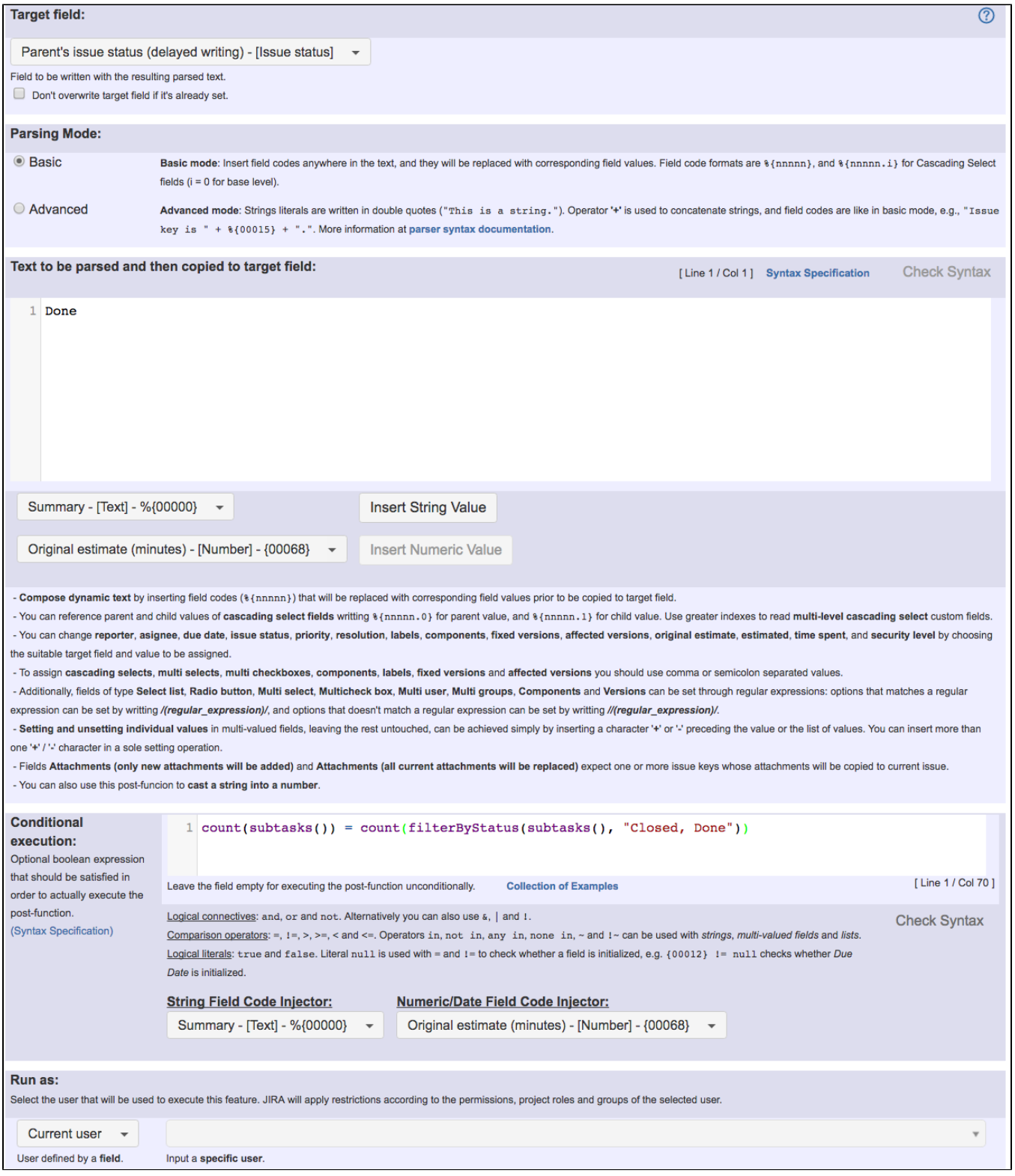
```
count(filterByStatus(siblingSubtasks(), "Closed, Done")) = count(siblingSubtasks())
```

This way we are preventing transition from being executed whenever the validation is not satisfied.

Alternative implementation

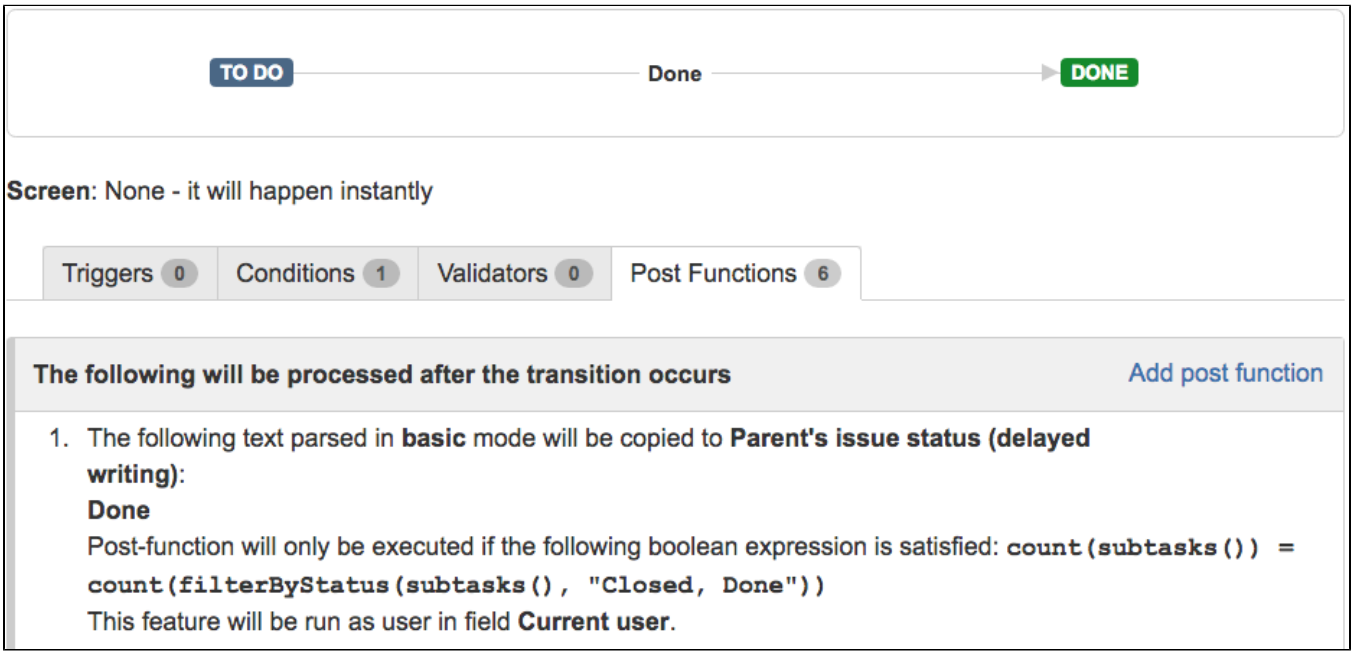
Using only a post-function in sub-tasks workflow, the other implementation is usually preferred since it also prevents parent issue from being closed until all sub-tasks are closed, even if you try to do it by manually triggering transition "**Done**" or "**Close Issue**". Anyway, sometimes you may want to allow manually overriding parent closing, even if there still are open sub-tasks.

This solution uses a post-function **Copy parsed text to a field** in transition "**Done**" or "**Close Issue**" in **sub-task's workflow** with the following configuration:



Conditional execution: **`count(subtasks()) = count(filterByStatus(subtasks(), "Closed, Done"))`**

Once configured, transition "**Close Issue**" in sub-task's workflow will look like this:



Other examples of that functions

Condition and validation on sub-tasks

Page: Close parent issue when all sub-tasks are closed
Page: Enforce certain type of sub-tasks to be "Resolved" to allow executing a transition
Page: Make linked issues, sub-tasks and JQL selected issues progress through its workflows
Page: Make parent issue progress through its workflow
Page: Proceed with a task only when all sub-tasks are completed
Page: Transition parent issue only when certain issue sub-task types are done

---

Boolean condition and validator with math. date-time or text-string terms

Page: Block a transition until all sub-tasks have certains 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

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 certains 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
- Automatically close resolved sub-tasks when parent issue is closed
  - example

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

Copy parsed text to a field

Page: Add all assignees of certain sub-task types to a "Multi-User Picker" custom field
Page: Add and remove a single or a set of items from multi valued fields
Page: Add current user to comment
Page: Add or remove request participants
Page: Add watchers from a part of the issue summary: "Summary_text - watcher1, watcher2, watcher3, ..."
Page: Assign issue based on the value of a Cascading Select custom field
Page: Assign issue to last user who executed a certain transition in the workflow
Page: Automatically close resolved sub-tasks when parent issue is closed
Page: Automatically reopen parent issue when one of its sub-tasks is reopened
Page: Calculate the time elapsed between 2 transition executions
Page: Close parent issue when all sub-tasks are closed
Page: Combine the values of several Multi-User picker fields
Page: Compose a parsed text including the "full name" or a user selected in a User Picker custom field
Page: Compose dynamic text by inserting field values in a text template
Page: Copy issue labels to a custom field
Page: Copy the value of a user property into a user picker
Page: Create a comment in sub-tasks when parent transitions
Page: Execute transition in epic
Page: Getting the number of selected values in a custom field of type Multi Select
Page: Limit the number of hours a user can log per day
Page: Make a sub-task's status match parent issue's current status on creation
Page: Make parent issue progress through its workflow
Page: Moving story to "In Progress" when one of its sub-tasks is moved to "In Progress"
Page: Moving story to "Ready for QA" once all its sub-tasks are in "Ready for QA" status
Page: Parse Email adresses to watchers list
Page: Parsing text from last comment and appending it to issue's summary
Page: Remove versions selected in a version picker custom field
Page: Replace certain issue link types with different ones
Page: Restrict parent issue from closing if it has sub-tasks that were created during a given parent issue status
Page: Set a Select or Multi-Select field using regular expression to express the values to be assigned
Page: Set assignee depending on issue type
Page: Set field depending on time passed since issue creation
Page: Set priority for issues that have been in a certain status for longer than 24 hours

- post-function
- sub-task
- transition
- outdated

- Change parent's status depending on sub-task's summary
  - 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
- 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
- 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
- Prevent transitioning when there is a blocking issue
  - example
  - validator
  - issue-links
  - 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

Page: Block a transition until all sub-tasks have certains 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: Cascading select comparer
Page: Check project property
Page: Close parent issue when all sub-tasks are closed
Page: Condition and validation based on JQL query
Page: Condition and validation based on regular expression
Page: Condition and validation on linked issues
Page: Condition and validation on sub-tasks
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: Except users in field
Page: Expression Parser
Page: If field A is populated then, field B must also be populated
Page: JWT Release Notes 2.1.26

Page: Set security level based on groups and project roles the reporter or creator are in
Page: Transition linked issues in currently active sprint
Page: Transition only a sub-task among several ones
Page: Transition parent issue only when certain issue sub-task types are done
Page: Update Cascading Select custom field with a value of the field in parent issue
Page: Update checkboxes custom field if a file has been attached during a transition
Page: Validation on issue attachments
Page: Validation on MIME types of issue attachments
Page: Writing a comment to blocked issues when blocking issues are resolved

Page: JWT Release Notes 2.1.29
Page: JWT Release Notes 2.2.8
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: Only users in a field
Page: Prevent issue creation if another issue with same field value already exists
Page: Project Properties
Page: Read a project property
Page: Read user property
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
Page: Virtual Fields