

# Auto-Transition Manager

The **Auto-Transitions Manager** is responsible of

- Evaluating if there is a Candidate Auto-Transition among available Transitions (only one)
- Transitioning the issue by passing the correct context
- Adding a comment related to this transition

Its behavior may be altered by **Auto-Transition Properties**.

## Evaluating Candidate Auto-Transition

For a specified Issue, the **Auto-Transitions Manager** will process as follow :

- Identification of all available Transitions (Step Transitions, Common Transitions and Global Transitions)
- Among these available Transitions, filter only transitions marked as Auto-Transition
- If only one Auto-Transition is available, it is the transition to execute

To qualify a Transition as a valid Auto-Transition, the Transition Property **allow.auto.transition** has to be append.

In the XML descriptor of the Workflow Transition, the **meta** tag has to be added ...

```
<action id="100" name="Close" view="resolveissue">
  ...
  <meta name="allow.auto.transition">true</meta>
  ...
</action>
```

You can also reserve the Auto-Transition to granted user, by adding the **ExtendedPermissionCondition** as Workflow Condition.

## Auto-Transitioning Arguments

Depending on the transition definition, a Screen may be associated. That means that values will have to be filled during the transition.

Since there is no human interaction, you will have to define the values to use on the transition.

To do that the **Auto-Transitions Manager** read all Transition Properties prefixed with **auto.transition.**, and assumes the end of the property as the field Id.

By example, if the Transition Screen has the resolution field, the wanted value (Resolution Id.) has to be defined using Transition Property **auto.transition.resolution** ...

```
<action id="100" name="Close" view="resolveissue">
  ...
  <meta name="auto.transition.resolution">1</meta>
  ...
</action>
```

Another example, if the Transition Screen has a customfield for a User Picker, the wanted value (User's name) has to be defined using Transition Property **auto.transition.customfield\_10010**...

```
<action id="100" name="Close" view="resolveissue">
  ...
  <meta name="auto.transition.customfield_10010">admin</meta>
  ...
</action>
```

Each defined field's values are injected in the context in order to perform the transition.

## Adding a comment related to this transition

### On this page:

- [Evaluating Candidate Auto-Transition](#)
- [Auto-Transitioning Arguments](#)
- [Adding a comment related to this transition](#)
- [Overview of Auto-Transition Properties](#)
- [Transition Sample](#)

When the Transition is performed, a comment is added to the Issue. But it is possible to modify this behavior using Transition Properties.

See Properties overview below

## Overview of Auto-Transition Properties

Properties	Values	Description
allow.auto.transition	True or False	True if the Transition may be assumed as a Auto-Transition
auto.transition. <b>fieldId</b>	Any valid value	Associated value is used in Transition Context for field identified by the Id. <b>fieldId</b>  If the field may have already a value before the transition, no change will be done if the special value <b>--unchanged--</b> is specified.
auto.transition.unchangeMissingFields		If presents, any missing field of the transition screen will be let unchanged with previous value
comment.auto.transition.enabled	True or False	True (default) if want a comment added for each performed Transition.
comment.auto.transition	Text to use as comment ...	This text support String substitution where {0} is the action Id. and {1} is the issue key.

## Transition Sample

This sample transition:

- is reserved to AutoTransitionService
- is performed if:
  - the user has the global administration permission
  - and if the Inactivity Condition on Worklog is confirmed,
- and since the associated screen is "resolveissue", the **Resolution** and **Assignee** fields have to be specified.
  - Resolution is set to **1** (Fixed)
  - Assignee keeps its previous value (due to the **--unchanged--** value).

```

<action id="100" name="Close" view="resolveissue">
  <meta name="jira.description">Close Transition reserved to
AutoTransitionService</meta>
  <meta name="jira.il8n.title">WFA_Close</meta>
  <meta name="allow.auto.transition">true</meta>
  <meta name="auto.transition.resolution">1</meta>
  <meta name="auto.transition.assignee">--unchanged--</meta>
  <meta name="auto.transition.unchangeMissingFields"></meta>
  <restrict-to>
    <conditions type="AND">
      <condition type="class">
        <arg name="class.name">com.atlassian.jira.
workflow.condition.InactivityCondition</arg>
        <arg name="onWorklog">15d</arg>
      </condition>
      <condition type="class">
        <arg name="class.name">com.atlassian.jira.
workflow.condition.ExtendedPermissionCondition</arg>
        <arg name="permission">admin</arg>
      </condition>
    </conditions>
  </restrict-to>
  <post-functions>
    <function type="class">
      <arg name="class.name">com.atlassian.jira.workflow.
function.issue.UpdateIssueStatusFunction</arg>
    </function>
    <function type="class">
      <arg name="class.name">com.atlassian.jira.workflow.
function.misc.CreateCommentFunction</arg>
    </function>
    <function type="class">
      <arg name="class.name">com.atlassian.jira.workflow.
function.issue.GenerateChangeHistoryFunction</arg>
    </function>
    <function type="class">
      <arg name="class.name">com.atlassian.jira.workflow.
function.issue.IssueReindexFunction</arg>
    </function>
    <function type="class">
      <arg name="eventTypeId">5</arg>
      <arg name="class.name">com.atlassian.jira.workflow.
function.event.FireIssueEventFunction</arg>
    </function>
  </post-functions>
</action>

```

- [Minyaa Suite](#)



#### Useful hint

*See also ...*