

Auto-Transition Management

What may be an Auto-Transition

Basically, in JIRA, is considered as Auto-Transition, any Workflow Transition where its [Initiator](#) is not directly raised by the user through the UI.

How JIRA Community does implement Auto-Transitions

In JIRA Community, there are different ways to implement an Auto-Transition, where different definitions of Auto-Transition coexist.

You will see Auto Transition done

- using a Jelly Script executed by the Jelly Script Runner Service ([Atlassian Jelly Escalation](#))
- using a Post-function, able to evaluate a Groovy Script (See different [Built-in Script](#) for [Groovy Script Runner](#), [Blog Post](#))
- using a Event Listener able to transition an action by its Id on an issue event ([Auto Transition Listener](#) of [JIRA Toolkit Plugin](#))
- using a Post-function, able to transition an action by its Id. (Implemented by **Transition All Subtasks Plugin**)

See Minyaa's Blog Post ...

Identifying the Type of Auto-Transitions to use

To implement Auto-Transitions, you have to identify

- **Which** transitions have to be executed ?
 - Any available Auto-Transition identified in linked Issue (sub-Task or any Linked Issues) of the current Issue
 - Any available Auto-Transition identified in any Issue
 - A Transition identified by its Id. in linked Issue (sub-Task or any Linked Issues)
- **When** they have to be executed ? An Auto-Transition can be raised by different events :
 - **Internal Transition** : the transition is launched automatically after another transition of the same issue
 - **External Transition** : the transition is launched automatically after a transition of the another issue
 - **Change Event** : the transition is launched automatically depending on an any event related to change in the context of the issue (any Field value, Actor, Project, Version, ...)
 - **Temporal Event** : the transition is launched automatically depending on an temporal event

Depending on these criteria, we are able to implement different types of Auto-Transition.

- [Transition with Automatic Actions](#)
- [Auto-Transition on Issue](#)
- [Remote-Transition by action Id](#)

And then, you will be able to decide **How to** implement your Auto-Transition.

Auto-Transitions with Minyaa

To cover these features, Minyaa provides different mechanisms ...

- [Auto-Transition Manager](#) : It is in charge of transitioning candidate Auto-Transitions
- [Auto-Transition Properties](#) : They allows to qualify an Auto-Transition and to alter a part of the transition's behavior.
- [Auto-Transition Service](#) : It is used to check periodically some issues.
- [Auto-Transition Post-Functions](#) : They allows to init some Auto-Transitions

On this page:

- [What may be an Auto-Transition](#)
- [How JIRA Community does implement Auto-Transitions](#)
- [Identifying the Type of Auto-Transitions to use](#)
- [Auto-Transitions with Minyaa](#)
- [Use Cases](#)

Use Cases

Some Use Cases where Auto-Transition may be encountered :

Use Case	Description	Implementation
Auto-Triage	A priority is applied depending different fields	Automatic Actions
Auto-Assignment	A user is assigned depending different fields (other than Component)	Automatic Actions
Need more Information	As soon as a new comment is added by the reporter /customer, the issue has to passed to the next step	Auto-Transition initiated from a Listener
Inactivity	No activity on the issue (Comment, Transition, Worklog, Edit) since a specified delay	Auto-Transition initiated from a Auto-Transition Service
Auto-Close of linked Issue	On Issue Close, we want see linked Issue, also closed.	Remote-Transition initiated from a Auto-Transition Post-Function
...



You should be able to find the good implementation to resolve your own Use Case ...

- [Getting started](#)



Useful hint

See also ...