Workflow Mappings

With Backbone, you can define workflow mappings based on issue status, allowing you to define fine-grained distributed workflows between synchronized issues. This guide will demonstrate how to map a distributed workflow between two synchronized issues.

Prerequisite: Upload XML workflow files to Backbone

Firstly, you need to upload your workflow data to Backbone, because Backbone cannot pull workflow data from your Jira instance directly.

You need to upload all workflows that are configured for the issue types that you will synchronize. To do this, you need to carry out the following steps for all synchronized projects:

1. For the first synchronized project, navigate to Project settings > Workflows
2. You will see a list of all the workflows configured for this project, separated by issue type. You need the workflow data for all the issue types you want to synchronize. For the relevant workflows, click the Actions pencil icon > Export > As XML
3. Repeat steps 1 and 2 for the second synchronized project. Even if the workflows are identical, you must get the separate XML files from the two different projects, because the status IDs, the transition IDs, etc, will be different
4. Navigate back to the Backbone configuration, and Workflows > XXX > Manage Jira workflows then upload the workflow files for both projects

Choose the correct workflow editor

Backbone includes two workflow mapping editors – the default (visual) editor and the advanced (text-based) editor. When you start configuring workflow mappings, it’s important to choose the correct one for your needs.

<table>
<thead>
<tr>
<th>Name</th>
<th>Functionality</th>
<th>Recommended use case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default editor</td>
<td>Lets you map workflow status between projects</td>
<td>All scenarios except when the advanced editor is recommended</td>
</tr>
<tr>
<td>Advanced editor</td>
<td>Lets you map the workflow status, and also select specific transitions</td>
<td>Two scenarios only:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- If you have already configured workflow mappings for a project in Backbone versions 3.3 or earlier, and you don't want to lose them. This is because these earlier versions only used the advanced editor – and you cannot switch back to the default editor in a project without losing workflows created in the advanced editor.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- There are certain transitions in your workflows that can be executed in multiple ways, and you need to define which exact path should be taken. This is because you can’t select which transitions between states should be made in the default editor.</td>
</tr>
</tbody>
</table>

Default editor

Let's create a workflow mapping with the default editor.

1. To get started, open the synchronization configuration and click Workflows > New workflow mapping:

   ![Workflow Mapping Interface](image)

   If you see a different view (with two columns for the two projects), this means that you've already created workflow mappings for this project with the advanced editor – you won’t be able to switch back to the simple editor without losing these.

2. Select which workflows you want to map, and what issue types you want to map the workflow for.
3. Now, you can use the editor to map what workflow state the partner project should be triggered to when your project sends them a certain workflow state:

In this example, when Project A sends Project B its current workflow status, Project B’s workflow status will transfer to the same status. If there is more than one path between two statuses, Backbone will always select the shortest path. If there are multiple possible shortest paths, you can select which Backbone should take by clicking the path selection button:

4. On the next screen, map what workflow status your project should transition to when the partner project sends a workflow state. You can configure this in the same way as step 3.

Advanced editor

The advanced editor lets you map workflow states between projects, and also select specific transitions between those states.

If you need to use the advanced editor, switch by clicking ... > Switch to advanced view:
If you want to create a workflow mapping for all issue types belonging to a project's default workflow, then click Default workflow mapping.

If you want to create a workflow mapping for (a) specific issue type/s, click New workflow mapping.

To define a workflow mapping, simply click then you can define transition rules for your project. When you define a transition rule, you can define what transition for project's workflow should be triggered when the workflow status in the partner project changes. The condition is always a combination of the local status and the status of the partner project. Whenever the status in the partner project changes, Backbone will search for a matching condition and perform the selected transition (or transitions). The list of transitions to select depends on the selected local status.

**Transition rule example**

Here, if Project A is in status In Progress, and Project B's workflow status changes to Done, then Project A's workflow status transitions to status In Review.