

Custom Workflow Paths

Custom Workflow Paths

For general information on workflow, see [Editing, Tracking, Workflow](#).

To create a custom workflow path, create a class that extends the `AbstractWorkflowPathHandler` class. Three elements are required to define a workflow path:

- Enumerating Workflow Types
- Enumerating Workflow Statuses
- Enumerating Workflow Path States
- Validating workflow actions
- Specifying the results of workflow actions

Enumerating Workflow Type

A workflow type specifies the general behavior of the workflow and the user interface relating to workflow. The currently defined workflow types are:

- `CONFLICT_PROJECT`: Two (or more) users finish work, where any conflicts arising are reviewed for a lead
- `REVIEW_PROJECT`: One user finishes work, which is then reviewed by a lead

If your new workflow functions in a significantly different way, extensive changes to the User Interface may be required.

To add new Workflow Types, edit the `WorkflowType` enum in the `mapping-model` project.

Enumerating Workflow Statuses

A Workflow Status defines the current workflow status of a map record, e.g. `NEW`, `EDITING_IN_PROGRESS`, etc.

To add new Workflow Statuses, edit the `WorkflowStatus` enum in the `mapping-model` project.

IMPORTANT: New Workflow Statuses must not be initial substrings of existing Workflow States, or the build will fail integration testing. That is, no Workflow Status can comprise the beginning of another Workflow Status. For example, `REVIEW` and `REVIEW_NEEDED` cannot coexist, while `NEW` and `REVIEW_NEW` can.

Enumerating Workflow Path States

A Workflow Path Handler consists of enumerated Workflow Path States. In the example below, the workflow has an initial state, an interim (editing) state, and a terminal state.

```
private static WorkflowPathState initialState, interimState, terminalState;
```

A Workflow Path State is defined by:

- The state name (used only for identification/logging purposes)
- Allowable workflow status combinations
- Allowable actions for this state

Setting allowable workflow status combinations

A Workflow Status Combination is a simple object containing a list of Workflow States. Each valid combination must be specified and added to the Workflow Path State, as in the following example:

```
initialState = new WorkflowPathState("Initial State");

initialState.addWorkflowCombination(new WorkflowStatusCombination(
    Arrays.asList(WorkflowStatus.REVISION, WorkflowStatus.NEW)
));
initialState.addWorkflowCombination(new WorkflowStatusCombination(
    Arrays.asList(WorkflowStatus.REVISION, WorkflowStatus.EDITING_IN_PROGRESS)
));
... add more combinations
```

Setting allowable actions for a workflow state

A Workflow Path State defines the allowable actions by a map of states -> set of allowable actions named "trackingRecordStateToActionMap". To set the allowable actions for a state, add it to the map as shown in the following example:

```
trackingRecordStateToActionMap.put(interimState,
    new HashSet<WorkflowAction>(
        Arrays.asList(WorkflowAction.SAVE_FOR_LATER, WorkflowAction.FINISH_EDITING
    ));
```

Validating Workflow Actions

The validity of a particular action by a particular user may be dependent on a number of factors, such as the user's role on a project, the status of particular record or records associated with a Workflow Path State, etc. The method `validateTrackingRecordForActionAndUser()` is the location to specify this logic.

```
@Override
public ValidationResult validateTrackingRecordForActionAndUser(
    TrackingRecord tr, WorkflowAction action, MapUser user) throws Exception {

    ... code omitted ...

    // check for null state retrieved
    if (state == null) {
        result
            .addError("Could not determine workflow path state for tracking record");
    }
    else if (state.equals(initialState)) {
        // YOUR CODE HERE
    } else if (state.equals(interimState)) {
        // YOUR CODE HERE
    } else if (state.equals(terminalState)) {
        // YOUR CODE HERE
    } else {
        result.addError("Could not determine workflow state for tracking record");
    }

    ... code omitted ...
}
```

Specifying the results of workflow actions

Currently, workflow actions are handled in the [Custom Project Specific Algorithm Handlers](#). See this section for instructions on how to add custom workflow action handling.

At the time of writing, it remains a development goal to move executing workflow actions into the Workflow Path Handlers.