

Workflow Configuration

0.1 Overview document

by Andreas Hartmann

1. Workflow Schemas

The workflow schema definition files of a publication are located at

{publication}/config/workflow/

A workflow schema definition looks as follows:

```
<workflow xmlns="http://apache.org/cocoon/lenya/workflow/1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://apache.org/cocoon/lenya/workflow/1.0
    ../../../../resources/entities/workflow/workflow.xsd">

  <state id="authoring" initial="true"/>
  <state id="live"/>
  <state id="trash"/>
  <state id="archive"/>

  <variable name="is_live" value="false"/>

  <transition source="authoring" destination="authoring">
    <event id="edit"/>
    <condition class="org.apache.lenya.cms.workflow.RoleCondition">
      edit, review, organize
    </condition>
  </transition>

  <transition source="authoring" destination="authoring">
    <event id="deactivate"/>
    <condition class="org.apache.lenya.workflow.impl.BooleanVariableCondition">
      is_live = true
    </condition>
    <condition class="org.apache.lenya.cms.workflow.RoleCondition">
      review, organize
    </condition>
    <assign variable="is_live" value="false"/>
  </transition>

  ...
</workflow>
```

The workflow namespace URI is

http://apache.org/cocoon/lenya/workflow/1.0

2. States

All states that are used in the workflow schema have to be declared using `<state>` elements. The initial state is marked with the `initial="true"` attribute.

3. Variables

All used variables have to be declared using `<variable>` elements. The initial value of the variable is assigned using the `value` attribute.

4. Transitions

A transition is declared using the `<transition>` element. The required attributes `source` and `destination` denote the states that are connected by this transition.

The transition element must contain one `<event>` element with an `id` attribute. Furthermore, it can contain an arbitrary number of `<condition>` and `<assign>` elements.

A transition element can have a `synchronized="true"` attribute. In this case, if the transition is triggered using a *SynchronizedWorkflowInstance*, it is invoked on all instances.

5. Variable Assignments

A variable assignment has the form

```
<assign variable="..." value="..." />
```

The variable must have been declared in this workflow schema. Because only boolean variables are supported, `value` must be either `true` or `false`.

6. Conditions

A condition has the form

```
<condition class="...">...</condition>
```

The `class` attribute contains the complete name (including the package) of the condition class. You can use the condition classes that ship with Lenya (see below) or implement your own conditions. All condition classes must implement the `org.apache.lenya.workflow.Condition` interface. The text inside the element is the expression that should be evaluated. It is passed as an argument to the `setExpression()` method.

6.1. BooleanVariableCondition

The *org.apache.lenya.workflow.impl.BooleanVariableCondition* requires an expression of the form

```
{variable-name} = {value}
```

`{variable-name}` is the name of a variable that was declared in the workflow schema. `{value}` is either `true` or `false`.

6.2. RoleCondition

The *org.apache.lenya.cms.workflow.RoleCondition* requires a comma-separated list of role IDs:

```
{role-id-1}, {role-id-2}, ...
```

The condition is complied when the current identity has one of these roles on the requested URL.

7. Assigning Workflow Schemas to Document Types

A workflow schema can be assigned to a document type in

```
{publication}/config/doctypes/doctypes.xconf
<doctypes>
  <doc type="content-page">
    ...
  <workflow src="workflow.xml" />
```

```
</doc>  
...  
</doctypes>
```