Defining Tasks

Overview document

Table of contents

All tasks to be used in a publication must be declared in the file `[publication-directory]/content/tasks.xconf`. A typical task configuration file looks like follows:

```
<?xml version="1.0"?>
<tasks>

  <!-- publish page -->
  <task id="publish">
    <label>Publish</label>
    <task id="publish" class="org.lenya.cms.publishing.DefaultFilePublisher">
      <parameter name="live-path" value="content/live"/>
      <parameter name="tree-live-path" value="content/live/tree.xml"/>
    </task>
    <task id="export" class="org.lenya.cms.publishing.StaticHTMLExporter">
      <parameter name="export-path" value="resources/export/pending"/>
      <parameter name="substitute-regexp" value="s/\/lenya\/your-publication//g"/>
    </task>
  </task>
</tasks>
```

Every task must have a unique `id` attribute. This identifier is used to address the task from the sitemap. Every top-level task must have a child element `<label>`. The content of this element is used, e. g., by the scheduler to let the user choose a task from a list. The actual implementation of a task is identified using the `class` attribute.

1. Parameters

Usually, a task needs some parameters to be executed properly. There are two possibilities to pass parameters to a task:

1. You can define all parameters or a subset of them in the `tasks.xconf` file.
2. The remainig parameters can be handled to the `TaskAction` or the `TaskJob` that executes the task. Request parameters always have a higher priority than parameters from the `tasks.xconf` file.

To simplifiy the passing of parameters to tasks, the `TaskAction` and `TaskJob` objects create some default parameters based on the request and session objects:

- `Task.PARAMETER_SERVLET_CONTEXT` - the path of the servlet context `/home/user_id/build/jakarta-tomcat/webapps/lenya`
- `Task.PARAMETER_SERVER_URI` - the server URI `http://www.yourhost.com:8080/lenya/publication/index.html`
- `Task.PARAMETER_SERVER_PORT` - the server port `http://www.yourhost.com:8080/lenya/publication/index.html`
- `Task.PARAMETER_CONTEXT_PREFIX` - the part of the URI that precedes the publication ID `http://www.yourhost.com:8080/lenya/publication/index.html`
- `Task.PARAMETER_PUBLICATION_ID` - the publication ID `http://www.yourhost.com:8080/lenya/publication/index.html`

2. Task Sequences

Tasks can be nested using so-called *task sequences*. Sub-tasks of other tasks don't need to have `<label>` elements because they can't be addressed independently. Whenever a task sequence is executed, all sub-tasks are executed in the same order as they are declared.

When you group tasks, the enclosing `<task>` element does not need a `<class>` attribute. If you omit it, the `TaskSequence` class is used as default. If you want to implement your own task grouping mechanism using a subclass of `TaskSequence`, you can append a class attribute.

All parameters that are passed to a TaskSequence are forwarded to all tasks in the sequence. By creating a subclass of

`TaskSequence` you could implement a parameter selection mechanism, e. g. using namespace prefixes.

3. Assigning Tasks to Document Types

Every document type can support a set of tasks. The labels of these tasks are displayed on the scheduler screen of a document of this type. To assign a task to a document type, you have to edit the file `[publication-directory]/config/doctypes/doctypes.xconf`:

```
<doctypes>
  <doc type="Simple-Document">
    <tasks>
      <task id="publish"/>
      <task id="backup"/>
      ...
    </tasks>
  </doc>
  ...
</doctypes>
```

You can associate every task with an arbitrary number of document types.