

# Rename

## Table of contents

- 1 Introduction..... 2
- 2 Implementation..... 2
- 3 Parameters..... 2
- 4 usecase..... 2
  - 4.1 Rename screen..... 2
  - 4.2 Rename confirmation..... 2
- 5 Ant Task ..... 3

## 1 Introduction

The rename operation corresponds to a move operation

## 2 Implementation

Similar to the operation [move](#) ( move.html)

## 3 Parameters

The parameters are string-value parameters

Required parameters:

- the area for the source document
- the document id for the source document
- the area for the destination document
- the name for the destination document
- the task id

## 4 usecase

They are implemented in the usecase sitemap (core)

### 4.1 Rename screen

URL :

```
{document-URL}?lenya.usecase=rename&lenya.step=showscreen
```

usecase sitemap:

```
<map:match pattern="rename" type="usecase">
  <map:match pattern="showscreen" type="step">
    <map:generate src="content/info/rename.xsp" type="serverpages"/>
    <map:transform src="xslt/info/rename.xsl"/>
    <map:call resource="style-cms-page"/>
  </map:match>
</map:match>
```

The parameters for the source are get with the serverpage through the page envelope input module. A form (build with the xslt transformation) sends then the parameters as request parameters with the new URL.

### 4.2 Rename confirmation

URL :

```
{document-URL}?lenya.usecase=rename&lenya.step=rename&...{source parameters}
```

usecase sitemap:

```
<map:match pattern="rename" type="usecase">
```

```
<map:match pattern="rename" type="step">
  <map:act type="task">
    <map:redirect-to session="true" uri="{request:contextPath}/{../../..1}/
{../../..2}"/"/>
  </map:act>
</map:match>
</map:match>
```

The action `org.apache.lenya.cms.cocoon.acting.TaskAction` calls the execution of the ant task.

## 5 Ant Task

The ant target `renameDocument` is in the publication :

```
{publication}/config/tasks/targets.xml
```

and depends on the different targets

- `firstareaproperties`, to set the needed properties dependent of the source area
- `secareaproperties`, to set the needed properties dependent of the destination area
- `newnamdocumentid`, to compute the unique destination id with the new name
- `firstdocumentpath`, to compute the directory of the source contents (Needed for the revisions and the rcml files)
- `secdocumentpath`, to compute the directory where are the destination contents (Needed for the revisions and the rcml files)
- `move`, to execute the different move operations

More about ant task, see the documentation [Ant Task](#) ( `../tasks/anttask.html`) and the [Javadoc](#) ( `../apidocs/1.2/index.html`)