

# Annotated forrest.xmap

## Table of contents

- 1 sitemap.xmap..... 2
  - 1.1 Definition of File-Resolver-Resource..... 4
  - 1.2 Second Match for '\*\*.xml'..... 4

## 1 sitemap.xmap

```

<?xml version="1.0"?> <!-- =====
Pipelines defining all Source XML types Forrest can handle. Converts from arbitrary
formats to standard Forrest doc-v12 XML. First, determines if a simple format file
exists. Candidates are: .html .ihhtml .txt .sxw .jspxwiki Otherwise, the default *.xml
handling is used. The XML's DTD type is discovered using the SourceTypeAction,
and an appropriate transformation applied. Thus to add new XML types, just
define a new <sourcetype> and an appropriate <map:when> handler. Generates :
Source XML, Body HTML for *.ehhtml Example URL: http://localhost:8888/index.xml
Used by : *.html and *.pdf pipelines in sitemap.xmap Uses : content/xdocs/**/*.xml
$Revision: 1.8 $ =====
--> <map:sitemap xmlns:map="http://apache.org/cocoon/sitemap/1.0">
<map:components> <map:generators default="file"> <map:generator
name="html" src="org.apache.cocoon.generation.HTMLGenerator">
<jtidy-config>WEB-INF/jtidy.properties</jtidy-config> </map:generator>
<map:generator name="text2xml" logger="sitemap.generator.textgenerator"
src="org.apache.cocoon.generation.TextGenerator"> <parameter
name="localizable" value="true"/> </map:generator> <map:generator
name="exception" src="org.apache.cocoon.generation.ParseExceptionGenerator"/
> </map:generators> <map:serializers default="html"/> <map:matchers
default="wildcard"/> <map:transformers default="xslt"> <map:transformer
name="pattern" src="org.apache.cocoon.transformation.PatternTransformer"
logger="sitemap.transformer.pattern"> <parameter name="groups"
value="true"/> </map:transformer> <map:transformer name="lexer"
src="org.apache.cocoon.transformation.LexicalTransformer"
logger="sitemap.transformer.lexer"> <parameter name="localizable"
value="true"/> </map:transformer> <map:transformer name="parser"
src="org.apache.cocoon.transformation.ParserTransformer"
logger="sitemap.transformer.parser"> <parameter name="flatten" value="true"/
> <parameter name="recovery" value="true"/> <parameter name="localizable"
value="true"/> </map:transformer> <map:transformer name="extparser"
src="org.apache.cocoon.transformation.ExtendedParserTransformer"
logger="sitemap.transformer.extendedparser"> </map:transformer> </map:transformers>
<map:actions> <map:action logger="sitemap.action.sourcetype" name="sourcetype"
src="org.apache.forrest.sourcetype.SourceTypeAction"> <sourcetype name="document-
v10"> <document-declaration public-id="-//APACHE//DTD Documentation V1.0//EN" /
> </sourcetype> <sourcetype name="howto-v10"> <document-declaration public-id="-//
APACHE//DTD How-to V1.0//EN" /> </sourcetype> <sourcetype name="todo-v10">
<document-declaration public-id="-//APACHE//DTD Todo V1.0//EN" /> </sourcetype>
<sourcetype name="changes-v10"> <document-declaration public-id="-//APACHE//DTD
Changes V1.0//EN" /> </sourcetype> <sourcetype name="document-v11"> <document-
declaration public-id="-//APACHE//DTD Documentation V1.1//EN" /> </sourcetype>
<sourcetype name="howto-v11"> <document-declaration public-id="-//APACHE//DTD How-
to V1.1//EN" /> </sourcetype> <sourcetype name="todo-v11"> <document-declaration
public-id="-//APACHE//DTD Todo V1.1//EN" /> </sourcetype> <sourcetype name="changes-
v11"> <document-declaration public-id="-//APACHE//DTD Changes V1.1//EN" /> </
sourcetype> <sourcetype name="todo-v12"> <document-declaration public-id="-//APACHE//
DTD Todo V1.2//EN" /> </sourcetype> <sourcetype name="changes-v12"> <document-

```

```

declaration public-id="-//APACHE//DTD Changes V1.2//EN" /> </sourcetype> <sourcetype
name="howto-v12"> <document-declaration public-id="-//APACHE//DTD How-to V1.2//
EN" /> </sourcetype> <sourcetype name="todo-v13"> <document-declaration public-id="-//
APACHE//DTD Todo V1.3//EN" /> </sourcetype> <sourcetype name="changes-v13">
<document-declaration public-id="-//APACHE//DTD Changes V1.3//EN" /> </sourcetype>
<sourcetype name="howto-v13"> <document-declaration public-id="-//APACHE//DTD How-
to V1.3//EN" /> </sourcetype> <sourcetype name="document-v20a"> <document-declaration
public-id="-//APACHE//DTD Documentation V2.0a//EN" /> </sourcetype> <sourcetype
name="document-v20"> <document-declaration public-id="-//APACHE//DTD Documentation
V2.0//EN" /> </sourcetype> <sourcetype name="todo-v20"> <document-declaration public-
id="-//APACHE//DTD Todo V2.0//EN" /> </sourcetype> <sourcetype name="changes-v20">
<document-declaration public-id="-//APACHE//DTD Changes V2.0//EN" /> </sourcetype>
<sourcetype name="howto-v20"> <document-declaration public-id="-//APACHE//DTD How-
to V2.0//EN" /> </sourcetype> </map:action> <map:action logger="sitemap.action.locale"
name="locale" src="org.apache.cocoon.acting.LocaleAction"> </map:action> </map:actions>
<map:selectors default="parameter"> <map:selector logger="sitemap.selector.parameter"
name="parameter" src="org.apache.cocoon.selection.ParameterSelector" /
> <map:selector logger="sitemap.selector.exception" name="exception"
src="org.apache.cocoon.selection.ExceptionSelector"> <exception
name="syntax" class="net.sourceforge.chaperon.process.ParseException"/
> <exception class="java.lang.Throwable" unroll="true"/> </map:selector>
<map:selector logger="sitemap.selector.config" name="config"
src="org.apache.cocoon.selection.SimpleSelector" /> </map:selectors> </map:components>
<map:resources> <map:resource name="transform-to-document"> <map:act
type="sourcetype" src="{src}"> <map:select type="parameter"> <map:parameter
name="parameter-selector-test" value="{sourcetype}" /> <map:when test="document-v10">
<!-- FIXME (JJP): This should use the v12 now--> <map:transform src="{forrest:stylesheets}/
docv10todocv11.xml" /> </map:when> <map:when test="document-v20a"> <map:transform
src="{forrest:stylesheets}/docv20todocv12.xml" /> </map:when> <map:when
test="document-v20"> <map:transform src="{forrest:stylesheets}/docv20todocv13.xml" />
</map:when> <map:when test="howto-v10"> <map:transform src="{forrest:stylesheets}/
howto2document.xml" /> </map:when> <map:when test="howto-v11"> <map:transform
src="{forrest:stylesheets}/howto2document.xml" /> </map:when> <map:when
test="howto-v12"> <map:transform src="{forrest:stylesheets}/howto2document.xml" /> </
map:when> <map:when test="howto-v13"> <map:transform src="{forrest:stylesheets}/
howto2document.xml" /> </map:when> <map:when test="howto-v20"> <map:transform
src="{forrest:stylesheets}/howto2document.xml" /> </map:when> <map:when
test="todo-v10"> <map:transform src="{forrest:stylesheets}/todo2document.xml" /> </
map:when> <map:when test="todo-v11"> <map:transform src="{forrest:stylesheets}/
todo2document.xml" /> </map:when> <map:when test="todo-v12"> <map:transform
src="{forrest:stylesheets}/todo2document.xml" /> </map:when> <map:when test="todo-
v13"> <map:transform src="{forrest:stylesheets}/todo2document.xml" /> </map:when>
<map:when test="todo-v20"> <map:transform src="{forrest:stylesheets}/todo2document.xml" /
> </map:when> <map:when test="changes-v10"> <map:transform src="{forrest:stylesheets}/
changesv10tochangesv11.xml" /> <map:transform src="{forrest:stylesheets}/
changes2document.xml" /> </map:when> <map:when test="changes-v11"> <map:transform
src="{forrest:stylesheets}/changes2document.xml" /> </map:when> <map:when
test="changes-v12"> <map:transform src="{forrest:stylesheets}/changes2document.xml" />
</map:when> <map:when test="changes-v13"> <map:transform src="{forrest:stylesheets}/

```

```

changes2document.xml" /> </map:when> <map:when test="changes-v20"> <map:transform
src="{forrest:stylesheets}/changes2document.xml" /> </map:when> <map:otherwise /> </
map:select> </map:act> </map:resource>

```

### 1.1 Definition of File-Resolver-Resource

```

<map:resource name="file-resolver"> <map:select type="exists"> <map:when
test="{project:content.xdocs}{uri}.ihtml"> <map:generate src="{project:content.xdocs}
{uri}.ihtml" type="html" /> <map:transform src="{forrest:stylesheets}/
html2document.xml" /> <map:transform type="idgen" /> <map:serialize
type="xml-document"/> </map:when> <map:when test="{project:content.xdocs}
{uri}.html"> <map:generate src="{project:content.xdocs}{uri}.html" type="html" />
<map:transform src="{forrest:stylesheets}/html2document.xml" /> <map:transform
type="idgen" /> <map:serialize type="xml-document"/> </map:when> <map:when
test="{project:content.xdocs}{uri}.txt"> <map:match type="regex" pattern="^(.*?)
([^\/*]*).xml$" /> <map:generate type="text2xml" src="{project:content.xdocs}{1}{2}.txt" /
> <map:transform src="{forrest:stylesheets}/text2document.xml"> <map:parameter
name="filename" value="{2}" /> </map:transform> <map:serialize type="xml-document"/
> </map:match> </map:when> <map:otherwise> <map:select type="exists"> <map:when
test="{project:temp-dir}/input.xmap"> <map:mount uri-prefix="" src="{project:temp-dir}/
input.xmap" check-reload="yes" pass-through="true"/> </map:when> </map:select>
<map:generate src="{project:content.xdocs}{uri}.xml" /> <map:call resource="transform-
to-document"> <map:parameter name="src" value="{project:content.xdocs}
{uri}.xml" /> </map:call> <map:serialize type="xml-document"/> </map:otherwise> </
map:select> </map:resource> </map:resources> <map:pipelines> <map:pipeline> <!--
===== --> <!--
INTERMEDIATE FORMATS --> <!--
===== --
> <map:match pattern="**body-*.html"> <map:select type="exists"> <map:when
test="{project:content.xdocs}{1}{2}.ehhtml"> <map:generate src="{project:content.xdocs}
{1}{2}.ehhtml" /> <map:transform src="{forrest:stylesheets}/html2htmlbody.xml" /
> <map:serialize type="xml" /> </map:when> </map:select> </map:match> <!--
===== --> <!--
SOURCE FORMATS --> <!--
===== -->

```

### 1.2 Second Match for '\*\*.xml'

```

<map:match pattern="**.xml"> <map:select type="config"> <map:parameter name="value"
value="{defaults:i18n}" /> <map:when test="true"> <map:act type="locale"> <map:call
resource="file-resolver"> <map:parameter name="uri" value="{../1}_{language}" /> </
map:call> </map:act> </map:when> <map:otherwise> <map:call resource="file-resolver">
<map:parameter name="uri" value="{1}" /> </map:call> </map:otherwise> </map:select> </
map:match> </map:pipeline> </map:pipelines> </map:sitemap>

```