

XSLT Transformer (2.1 legacy document)

Table of contents

1 Comments.....4

Table of contents

1 Trax/XSLT Transformer.....	3
2 The XSLT Processor.....	3

Warning:

This document was copied as is from the Cocoon 2.1 documentation, but has not yet been fully reviewed or moved to its new home.

1. Trax/XSLT Transformer

The xslt transformer reads an xsl document from the local file system or from any url. It transforms the sax stream using this stylesheet.

The xslt transformer is the default transformer .

- Name : xslt
- Class: org.apache.cocoon.transformation.TraxTransformer
- Cacheable: yes - uses the last modification date of the xsl document for validation.

The xslt transformer is configurable. You can specify one or more of the following configuration information:

- use-request-parameters: true|false - Setting this to true makes all request parameters available in the XSLT stylesheet. Note that this might have issues concerning cachability of the generated output of this transformer, the caching algorithm not only checks the last modification date but also all values of the request parameters. This property is false by default. If set to true the values of a request parameter is available using a variable in the xslt with the name of the parameter.
- use-browser-capabilities-db: true|false - This configuration forces the transformer to make all properties from the browser capability database available in the XSLT stylesheet as. Note that this might have issues concerning cachability of the generated output of this transformer as the caching algorithm adds this values to the validation phase. The default for this property is false.
- use-cookies: true|false - This configuration forces the transformer to make all cookies from the request available in the XSLT stylesheetas. Note that this might have issues concerning cachability of the generated output of this transformer. This property is false by default.
- xslt-processor-role: [role name] - This configuration allows to specify the XSLT processor (see below) that will be used by its role name. This allows to have several XSLT processors in the configuration (e.g. Xalan and Saxon) and choose one or the other depending on the needs of stylesheet specificities. This property defaults to "org.apache.cocoon.components.xslt.XSLTProcessor" which is the standard role name for an XSLTProcessor.

The "use-request-parameters" and "use-browser-capabilities-db" configuration of a transformer can be changed for one single pipeline by specifying parameters with the same name:

```
<map:transform src="stylesheet.xsl" type="xslt"/> <!-- The type attribute can be omitted as it is the default transformer. -->
```

The "use-request-parameters" and "use-browser-capabilities-db" configuration of a transformer can be changed for one single pipeline by specifying parameters with the same name:

```
<map:transform src="stylesheet.xsl"> <map:parameter name="use-request-parameters" value="true"/> </map:transform>
```

In addition all other parameters to the transformer are available in the stylesheet as <xsl:param/>s (These values are also used in the caching algorithm.)

2. The XSLT Processor

The XSLT Transformer uses a component called XSLTProcessor. This component is configured in the

cocoon.xconf. You can configure it as follows:

- use-store: true|false - If set to true it forces the xslt processor to put the generated templates from the XSLT stylesheet into the system store. This property is true by default.
- transformer-factory: [class name] - tells the transformer to use a particular implementation of javax.xml.transform.TransformerFactory. This allows to force the use of a given TRAX implementation (e.g. xalan or saxon) if several are available in the classpath. If this property is not set, the transformer uses the standard TRAX mechanism (TransformerFactory.newInstance()).

1. Comments

add your comments