

VelocityTransformer

by Don Brown

Transformer features and configuration

Table of contents

1. Overview.....	1
2. Features.....	2
3. Configuration.....	2
3.1. Global.....	2
3.2. Transform-specific.....	3
3.3. Examples.....	3

1. Overview

This transformer uses [Velocity](#), specifically [Anakia](#) to transform the XML with Velocity templates. Anakia uses JDOM XML objects to make it easy to work with the XML and further extends the Element object to add direct support for XPath evaluations.

The following objects will be available to each Velocity template:

Name	Description
\$req	The HTTP request object
\$res	The HTTP response object
\$ses	The HTTP session object
\$app	The servlet context
\$root	The root element of the stxx-created XML document
\$msg.getMessage(\$string)	Retrieves the localized message from the

	appropriate application resources
<code>\$relativePath</code>	The absolute path to the web application directory
<code>\$escape.getText(\$string)</code>	This context object will convert HTML Entities in the <code>\$string</code> that is passed into it and it will return the converted String. This is good for dealing with CDATA. The entities that are converted are: " -> " ; < -> < ; > -> > ; & -> &

Table 1: Context Objects

All node lists returned from Anakia (extended JDOM) objects through `$element.selectNodes`, `$element.content`, and `$element.children` have two special features:

- they support the `selectNodes` method just as a single element does that applies an XPath expression to all nodes in the list, and
- when inserted into template output by simply specifying `$list`, they produce the XML fragment consisting of all nodes they contain. This eliminates much of the `#foreach` code in templates.

For more information, visit the [Anakia](#) page.

2. Features

- The features of Velocity
- Easier to pick up than XSL
- Supposedly faster than XSL transformations

3. Configuration

3.1. Global

Property	Value	Status
<code>contentType</code>	The mime type of the transformed content	Required
<code>templateEncoding</code>	The conventional encoding specification for the Velocity templates as supported by your JVM, for example "UTF-8" or "ISO-8859-1".	Optional
<code>outputEncoding</code>	The conventional encoding specification for the output as supported by your JVM, for	Optional

	example "UTF-8" or "ISO-8859-1".	
--	----------------------------------	--

Table 1: Global stxx.properties properties

3.2. Transform-specific

These parameters are specified in stxx 1.2+ pipeline configuration format. See [Configuring with Struts 1.1](#) for more information.

Parameter	Value	Status
path	The path to the Velocity template file	Required
debug	true false (default)	Optional

Table 1: Transform-specific parameters

3.3. Examples

Global configuration:

```
stxx.transformer.velocity.class=com.oro.ad.stxx.transform.VelocityTransformer
stxx.transformer.velocity.mimeType=text/html
stxx.transformer.velocity.encoding=ISO-8859-1
```

Transform-specific configuration:

```
<pipeline match="velocity/*.dox">
  <transform type="velocity">
    <param name="path" value="/{1}.vm" />
  </transform>
</pipeline>
```