

Saxon

Using Saxon

(This procedure is for both Saxon6.5 and Saxon8, tested Saxon9b with 2.1.8 and had no issues)

Add saxon8.jar to lib/local/. Note that with Cocoon 2.1.5.1 (and probably other versions as well) you need to remove the META-INF directory from the saxon jar, see this [discussion](#) for more info.

Edit src/webapp/WEB-INF/cocoon.xconf and declare the component for Saxon after the other xslt-processor components ...

```
<component logger="core.xslt-processor"
  role="org.apache.excalibur.xml.xslt.XSLTProcessor/saxon"
  class="org.apache.excalibur.xml.xslt.XSLTProcessorImpl">
  <parameter name="use-store" value="true"/>
  <parameter name="incremental-processing" value="false"/>
  <parameter name="transformer-factory" value="net.sf.saxon.TransformerFactoryImpl"/>
</component>
```

Note: For Saxon7 and Saxon8 use "transformer-factory" net.sf.saxon.TransformerFactoryImpl and for Saxon6 use com.icl.saxon.TransformerFactoryImpl

Edit src/webapp/samples/hello-world/sitemap.xmap to add the "xslt-saxon" transformer ...

```
<map:components>
  <map:transformers default="xslt">
    <map:transformer name="xslt-saxon" pool-grow="2" pool-max="32" pool-min="8"
      src="org.apache.cocoon.transformation.TraxTransformer">
      <use-request-parameters>false</use-request-parameters>
      <use-browser-capabilities-db>false</use-browser-capabilities-db>
      <xslt-processor-role>saxon</xslt-processor-role>
    </map:transformer>
  </map:transformers>
</map:components>
```

Add a sitemap match to use the xslt-saxon transformer ...

```
<map:match pattern="hello.html">
  <map:generate src="content/hello.xml"/>
  <map:transform type="xslt-saxon" src="style/xsl/page2html.xsl"/>
  <map:serialize type="html"/>
</map:match>
```

If you want to see Saxon being used then raise the [ExploringTheLogs](#) and look in core.log

Other Issues

There is a problem with XSP in combination with Saxon7. When the Saxon libraries are in your classpath, the root element of a page generated by XSP gets two extra namespace attributes:

```
xmlns:xml="http://www.w3.org/XML/1998/namespace" xmlns:xsp="http://apache.org/xsp"
```

This is only noticeable if the XSP generator is immediately followed by a serializer. In this case, Internet Explorer will complain: *The namespace prefix is not allowed to start with the reserved string "xml"*. Firefox won't mind these namespace prefixes. A solution is to add an identity transformer (using Xalan or Saxon) after the generator.

Mailing list discussions

Some other discussion points (be careful, there are some furbies which may confuse you):

1. Cocoon documentation: [Trax/XSLT Transformer](#)
2. Discussion: [Re: Transforming Big XML Files](#)
3. Discussion: [Re: Multiple XSL Transformers](#)
4. Discussion: [Re: Saxon and Cocoon](#)

5. Discussion: [Re: default xsl transformer?](#)
6. Discussion: [Re: BUG: EmptyStackException](#) - Timothy Larson describes how he configured Saxon
7. Discussion: [Re: Using Saxon 7.9 with Cocoon?](#) (note Vadim's comment about needing CVS version of excalibur-xmlutils)

More information

- [XSLT 2.0 - An Interview with Michael Kay](#)