

# CreateDocBookDocumentation

## Create DocBook XML based documentation with Ant

### Validate with jing

```
<java jar="lib/jing/jing.jar" fork="true" failonerror="true">
  <jvmarg value="-Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.parsers.
XIncludeParserConfiguration" />
  <jvmarg value="-Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl" />
  <arg value="\${docbook.basedir}/rng/docbook.rng" />
  <arg line="\${xml.fs.property}" />
</java>
```

I am using java here rather than the jing task . I found the idea to use the jvmargs by reading this bug report of jing-trang <http://code.google.com/p/jing-trang/issues/detail?id=38>

### Transform into HTML

Jars needed :

- for xslt
- saxon9.jar xercesImpl.jar xml-apis.jar xml-resolver.jar

```
<xslt basedir="\${sample.docs.xml.dir}" force="true" destdir="\${sample.docs.html.dir}" style="\${docbook.xsl.
basedir}/html/docbook.xsl">
  <include name="*.xml" />
  <classpath>
    <fileset dir="lib/xslt" />
  </classpath>
  <sysproperty key="org.apache.xerces.xni.parser.XMLParserConfiguration" value="org.apache.xerces.parsers.
XIncludeParserConfiguration" />
</xslt>
```

### Transform into PDF

Thanks to Wendy Smoak for the wiki page <http://wiki.wsmoak.net/cgi-bin/wiki.pl?DocBookToPDF> which put me on the right track.

Jars needed :

- for xslt
- saxon9.jar xercesImpl.jar xml-apis.jar xml-resolver.jar
- for fop
- download fop 1.0. This contains fop-1.0.jar avalon-framework-4.2.0.jar batik-all-1.7.jar commons-io-1.3.1.jar commons-logging-1.0.4.jar serializer-2.7.0.jar xalan-2.7.0.jar xercesImpl-2.7.1.jar xml-apis-1.3.04.jar xml-apis-ext-1.3.04.jar xmlgraphics-commons-1.4.jar

```

<xslt style="${docbook.xsl.basedir}/fo/docbook.xsl" extension=".fo" basedir="${sample.docs.xml.dir}"
includes="sample-user.xml" destdir="${sample.docs.fo.dir}">
  <classpath>
    <fileset dir="lib/xslt" />
  </classpath>
  <!-- these parameters make bookmarks work in the generated PDF -->
  <param name="fop.extensions" expression="0"/>
  <param name="fopl.extensions" expression="1"/>
  <!-- this allows xinclude processing to work -->
  <sysproperty key="org.apache.xerces.xni.parser.XMLParserConfiguration" value="org.apache.xerces.parsers.
XIncludeParserConfiguration" />
</xslt>
<taskdef name="fop" classname="org.apache.fop.tools.anttasks.Fop">
  <classpath>
    <fileset dir="${project.lib}/fop">
      <include name="*.jar" />
    </fileset>
  </classpath>
</taskdef>
<fop format="application/pdf" outdir="${sample.docs.pdf.dir}">
  <fileset dir="${sample.docs.fo.dir}">
    <include name="*.fo" />
  </fileset>
</fop>

```