AntennahouseSerializer

PDF creation out of a cocoon pipline is a powerful and easy way to convert your html pages into printable form. Standard cocoon embeds fop using the FOPSerializer. When speed comes into account one might be interested in alternatives to fop. There are differnet commercial xml-fo-renderers available. XEP comes with a cocoon serializer, Antennahouse Formatter does not.

In the scenario (cocoon 1.8) I am currently working with we use Antennahouse in the following way: write documents to harddisc, call the commandline script and then load the pdf from harddisc.

While migrating to cocoon 2.1 I rendered this scheme to the serializer-notation, i.e. streaming the xml into the serializer and a pdf out of it. I did not get any JAVA-API or the like from Antennahouse so the Serializer relies on the shell-script and a valid Antennahouse installation. The given solution will work with any (other) serialization process that is able to get its input from stdin and stream its output to stdout.

You use it like this:

Serializer definition (the shell command is given as cmd-tag inside the defintion):

```
<map:serializer logger="sitemap.serializer.antennahouse2pdf" mime-type="application/pdf" name="
antennahouse2pdf" src="de.abs.efonds24.sitemapElements.AntennaHouseSerializer">
  <cmd>/bin/bash /usr/XSLFormatterV3/run.sh -extlevel 3 -pea</cmd>
</map:serializer>
```

Serializer usage (well known std pipeline):

```
<map:match pattern="pdffile.pdf">
  <map:generate type="file" src="source-xml-fo.xml"/>
  <map:transform type="xslt" src="stylesheet.xsl"/>
  <map:serialize type="antennahouse2pdf"/>
</map:match>
```

The Serializer code is here: AntennaHouseSerializer.java

The code does not include any licensed code from the Antennahouse Inc. but you will have to have a valid license to test or use the Serializer.

I hope to hear from you if you like the idea, how you like the code and whether it works for you or not. Concerning speed we have to take into account that with this shell script based version, additional parsing of the xml-fo is necessary since no sax stream but the actual xml byte stream has to be piped into the serialization process. Also there are buffers involved.

– MichaelWirz <<DateTime(2005-12-15T10:48:04Z)>>
MichaelWirz