Pipeline Processing

Table of contents

Pipeline Processing

Overview

Kabeja provides a small processing system, where you can setup the different steps of the DXF conversion. The processing system uses a pipeline concept and is borrow from the XML web development framework Cocoon. A pipeline is splitted into "post processing" -> "XML-SAX generation" -> "XML SAX Filtering" -> "XML SAX Serialization". You are able to setup different pipelines for you need, e.g. one pipeline for DXF->SVG->PDF or a second for DXF->SVG->XSLT->JPEG.

will parse your CAD data. At the moment the Kabeja project provides only a DXF parser.

a PostProcessor will work direct with the parsed CAD data and can modify,delete or add data.

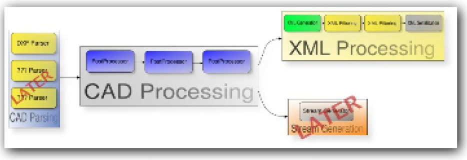
will generate an XML Stream (SAX events here) from the CAD data. The Kabeja project provides at the moment only an SVG generator. The dxf2calc project provides an statistics generator, which collects the bounds and length information of all containing geometries.

will filter or transform the XML data of the pipeline. We have here a XSLTFilter, which will apply your xslt-stylesheets. For embeding images into the SVG we also provide a special filter.

will output the XML format to any other format. The Kabeja project provides XML serializer, which will simple output the XML. By using the Batik project we can provide SVG2PDF, SVG2PNG, SVG2JPEG, SVG2TIFF and more serializers.

will direct output the parsed CAD data to any format. At the moment the Kabeja project does not provide a Stream generator. In one of the next releases we will add an Hatchpattern extractor, for extraction the used hatch pattern of a CAD drawing.

The following figure tries to give you an overview of the processing system (as PDF).



Usage

Pipeline Processing

The processing system will setup in a XML file. Take a look at "conf/process.xml". To use the processing system go into the Kabeja folder and type:

```
java -jar launcher.jar  -nogui -pp conf/process.xml   -pipeline svg  my.dxf my.svg
```

to invoke the pipeline "svg" ("-pipeline" switch) from the "conf/process.xml" ("-pp" switch) configuration file.

You can also process a complete directory, e.g. you have a pipeline for PDF-generation then use:

```
java -jar launcher.jar -nogui -pp conf/process.xml   -pipeline pdf   C:\my_dxf_folder
```

to convert the complete directory into pdf files.

Example pipelines

The following table explains the example processing, which comes with the Kabeja release.

Pipeline name	Comment
svg	Converts the file/directory to SVG
jpeg	Converts the file/directory to SVG -> JPEG
png	Converts the file/directory to SVG -> PNG
tiff	Converts the file/directory to SVG -> TIFF
pdf	Converts the file/directory to SVG -> PDF
debug	Converts the file/directory to SVG and mark the entities and bounds, which enlarge the draft.
mozilla	Converts the file/directory to SVG and apply a filter, which set fixed values for SVG-attribute stroke-width and font-size, otherwise Mozilla/Gecko based have problems to show the SVG.
embed	Embeds images and fonts (if they are setup) into the SVG-output.