Ant task

FOP provides an Ant task for automating the document build process.

1. Description

The FOP Ant task will convert XSL-FO documents to PDF/PS/PCL/MIF/RTF output (see Output formats for available formats).

To call FOP tasks within Ant, first add a FOP task definition to your Ant build file. One method of defining the task is as follows:

```
<property name="fop.dir" value="....path to your FOP jar files..."/>

<taskdef name="fop"
         classname="org.apache.fop.tools.anttasks.Fop">
         <classpath>
            <pathelement location="${fop.dir}\fop.jar"/>
            <pathelement location="${fop.dir}\avalon.jar"/>
            <pathelement location="${fop.dir}\batik.jar"/>
         </classpath>
</taskdef>
```

Then create FOP tasks within your Ant build file, using the FOP task parameters listed below.

2. Parameters for FOP Ant task

Attribute	Description	Required
fofile	XSL-FO file to be rendered	Yes, if no fileset nested element is used
outfile	Output filename	Yes, when fofile is used. (This attribute is not valid for filesets.)

format	Possible output formats:
	`application/pdf`
	`application/postscript`
	`application/vnd.mif`
	`application/rtf`
	`application/vnd.hp-PCL`
		No, defaults to `application/pdf`

	`text/plain` `text/xml`	
outdir	Output directory	Required if a fileset is used to specify the files to render; optional for fofile. (Can alternatively specify the full path in the fofile value.)
userconfig	User configuration file (same as the FOP "-c" command line option)	No
messagelevel	Logging level Possible values: `error`, `warn`, `info`, `verbose`, `debug`	No, defaults to `verbose`
logFiles	Controls whether the names of the files that are processed are logged (`true`) or not (`false`)	No, default is `true`

Table 1: Parameters specified as attributes

Attribute	Description	Required
fileset	FileSets are used to specify multiple XSL-FO files to be rendered.	Yes, if no fofile attribute is supplied

Table 2: Parameters specified as nested elements

3. Examples

The following example converts a single XSL-FO file to a PDF document:

```
<target name="generate-pdf" description="Generates a single PDF file">
   <fop format="application/pdf"
        fofile="c:\working\foDirectory\foDocument.fo"
        outfile="c:\working\pdfDirectory\pdfDocument.pdf" />
</target>
```

This example converts all XSL-FO files within an entire directory to PostScript:

```
<target name="generate-multiple-ps"
        description="Generates multiple PostScript files">
   <fop format="application/postscript"
        outdir="${build.dir}" messagelevel="debug">
        <fileset dir="${fo.examples.dir}">
           <include name="*.fo"/>
        </fileset>
   </fop>
</target>
```