

## Adding a font



### Chapter 13. Print customizations



## Adding a font

There are three basic font families built into all the tools: Times Roman, Helvetica, and Courier. These correspond to the `serif`, `sans-serif`, and `monospace` generic families, respectively, that FO processors recognize. You can specify these font names or generic names in stylesheet parameters such as `body.font.family` and they will just work. But what if your print design calls for other fonts, or you just want to try something a little different?

Using other fonts requires three steps:

- You have to locate the font file in a format your XSL-FO processor will accept.
- You have to configure your XSL-FO processor to work with the new font.
- You have to include the new font name in the FO output.

Most of the XSL-FO processors can handle other fonts. What kind of fonts they can handle and how they are configured depend entirely on the processor, so you should consult the documentation for your XSL-FO processor. Included here are examples for FOP and XEP.

### Locate the font file

Before you can use a new font, you must locate a font file that contains the diagrams of the font's characters. Font files come in various formats, not all of which will work with a given XSL-FO processor. Typical font file formats include TrueType and PostScript Type 1. If you have a Windows system, you can find several TrueType fonts under `\\WINDOWS\\FONTS` with a `.ttf` filename extension.

It is generally best to embed extra fonts in the generated PDF file, since the new fonts are unlikely to be resident and available on a given printer. With embedded fonts, the character diagrams are included in your PDF file, which makes your document portable. But the font file you use must permit the font to be embedded.

### Configuring fonts in XSL-FO

Each of the XSL-FO processors has their own configuration process for fonts. For the latest information on your processor, see the processor's documentation.

#### Configuring a font in FOP

FOP will accept PostScript Type 1 and TrueType fonts. There are some restrictions and

several options for using fonts in FOP. See

<http://xmlgraphics.apache.org/fop/0.93/fonts.html> for more details about configuring fonts in FOP. The following procedure configures a Garamond TrueType font, including its bold and italic variations.

1. Generate a FOP font metrics file from the TrueType font file. FOP provides a Java command for doing so. The following example reads the Windows GARA.TTF file and generates a `garamond.xml` font metrics file in a `fonts` subdirectory of the current directory. Replace the `version` strings with the actual version numbers for the files in your FOP distribution.

```
java -cp "../fop-0.93/build/fop.jar;../fop-0.93/lib/serializer-version.jar;W
../fop-0.93/lib/commons-logging-version.jar;../fop-0.93/lib/commons-io-version.jar" W
org.apache.fop.fonts.apps.TTFReader W
/WINDOWS/FONTS/GARA.TTF fonts/garamond.xml
```

Your Java CLASSPATH must include the `fop.jar` and the other jar files shown in the example. These files are included with the FOP distribution.

2. Do the same for the bold and italic variations of the font.

```
java -cp "../fop-0.93/build/fop.jar;../fop-0.93/lib/serializer-version.jar;W
../fop-0.93/lib/commons-logging-version.jar;../fop-0.93/lib/commons-io-version.jar" W
org.apache.fop.fonts.apps.TTFReader W
/WINDOWS/FONTS/GARABD.TTF fonts/garamond-bold.xml
```

```
java -cp "../fop-0.93/build/fop.jar;../fop-0.93/lib/serializer-version.jar;W
../fop-0.93/lib/commons-logging-version.jar;../fop-0.93/lib/commons-io-version.jar" W
org.apache.fop.fonts.apps.TTFReader W
/WINDOWS/FONTS/GARAIT.TTF fonts/garamond-italic.xml
```

3. Register the font with a FOP configuration file, which can have any filename such as `userconfig.xml`. The following example is for FOP 0.93:

```
<fop version="1.0">
  <renderers>
    <renderer mime="application/pdf">
      <fonts>
        <font metrics-url="file:///c:/xml/fonts/garamond.xml"
          kerning="yes"
          embed-url="file:///c:/Windows/fonts/GARA.TTF">
          <font-triplet name="Garamond" style="normal" weight="normal"/>
        </font>
        <font metrics-url="file:///c:/xml/fonts/garamond-bold.xml"
          kerning="yes"
          embed-url="file:///c:/Windows/fonts/GARABD.TTF">
          <font-triplet name="Garamond" style="normal" weight="bold"/>
        </font>
        <font metrics-url="file:///c:/xml/fonts/garamond-italic.xml"
          kerning="yes"
          embed-url="file:///c:/Windows/fonts/GARAIT.TTF">
          <font-triplet name="Garamond" style="italic" weight="normal"/>
        </font>
      </fonts>
    </renderer>
  </renderers>
  ...
</fop>
```

You specify the font metrics file and the path to the .TTF font file using attributes. You can enter the paths as absolute paths, or as relative to a path specified in a `base` element in the file. You specify the name by which you will reference the font in the `font-triplet` element. The triplet refers to the unique combination of name, weight, and style for each variation of the font. Those triplets are used by FOP to switch fonts for inline font changes. See the example configuration file `conf/fop.xconf` included with the FOP distribution.

4. Process your FO file with FOP using the `-c userconfig.xml` option:

```
../fop-0.93/fop.bat W
-c userconfig.xml W
booktest.fo W
booktest.pdf
```

## Configuring a font in XEP

The XEP documentation provides details for configuring XEP to use other fonts. The simplest case is to install the font directly in the `afm` directory in the XEP installation area on your system. Then you can just edit the `xep.xml` file that is in the XEP installation (or `etc/fonts.xml` prior to version 4). For example, these lines add the TrueType Palatino fonts to a version 4 installation.

```
<font-group xml:base="file:/C:/Windows/Fonts/">
  <font-family name="Palatino" embed="true"
    ligatures="&#xFB00; &#xFB01; &#xFB02; &#xFB03; &#xFB04;">
    <font>
      <font-data ttf="pala.ttf"/>
    </font>

    <font style="italic">
      <font-data ttf="palai.ttf"/>
    </font>

    <font weight="bold">
      <font-data ttf="palab.ttf"/>
    </font>

    <font weight="bold" style="italic">
      <font-data ttf="palabi.ttf"/>
    </font>

  </font-family>
</font-group>
```

The pathname to the .ttf file is taken to be relative to the `xml:base` path in the `font-group` container. Or place the fonts in the `fonts` subdirectory in the XEP installation, where they will be found automatically without an `xml:base`.

XEP uses the `style` and `weight` attributes in `font` elements to switch to bold and italic for inline font changes.

## Configuring a font in Antenna House

If you are using Antenna House's XSL Formatter on Windows, then you can automatically use all the fonts that are registered in the Windows Fonts utility. If you are on another platform, then you add fonts to the Font Configuration file as described in the Antenna House documentation.

## Adding a new font to FO output

To add a new font to your stylesheet, you may be able to do so with a stylesheet parameter. Otherwise, you will need to write a customization. The FO stylesheet provides a few parameters to specify font family names. These include:

Parameter name	Description
<code>body.font.family</code>	Used for ordinary text in paragraphs and such.
<code>title.font.family</code>	Used for book, chapter, and section titles.
<code>monospace.font.family</code>	Used for <code>programlisting</code> , <code>literal</code> , and other elements where a monospace font is called for.

If you want to change any of these, then you just need to set the corresponding parameter in a [customization layer](#) or on the command line. The value must match the name in the font configuration file for the FO processor. For example, to set the body font to Garamond, add this line to your customization layer:

```
<xsl:param name="body.font.family">Garamond</xsl:param>
```

When processed with the stylesheet, the FO output includes the following:

```
<fo:root font-family="Garamond" ... >
```

If you then process the FO output with FOP or XEP as configured above, your PDF output should use Garamond for the body font. If you also configured the bold and italic font files, then those should automatically be used when needed.

The three stylesheet font parameters may not meet your needs if you need finer control. For example, the `title.font.family` applies to all titles, but you may want your book and chapter titles in one font, and your section titles in a different font. The section title font is controlled by a couple of attribute-sets. See [the section "Section titles"](#) for a description of those attribute sets. In a customization layer, you only need to specify the new font in the appropriate attribute-set. The following example sets the font for all levels of section titles:

```
<xsl:attribute-set name="section.title.properties">
  <xsl:attribute name="font-family">Garamond</xsl:attribute>
  <xsl:attribute name="font-weight">bold</xsl:attribute>
</xsl:attribute-set>
```

These properties will be merged with the attribute-set of the same name in the

stylesheet, overriding the default font settings while leaving the other properties in effect.

If the element for which you want to change the font does not have an attribute-set in the stylesheet, you will have to customize the template that formats the element. For an example of customizing an element's font, see [the section "Line annotations"](#).

DocBook XSL: The Complete Guide - 4th  
Edition

[PDF version available](#)

Copyright © 2002-2007 Sagehill  
Enterprises



Multi-columns and spans



Numbering paragraphs