

Embedding Fonts into Casabac's FOP/PDF Processing

Casabac's FOP/PDF processing is a framework that allows you to create PDF documents and printable documents in a simple way. Apache FOP (Formatting Objects Processor) is used as basis.

One of the most common questions is: how can I embed new fonts that are not provided as PDF-standard fonts (such as Arial, Times Roman, Courier) into PDF documents? - This document explains how to do so and how to reference the fonts from your XML layout definition that you create within Casabac.

Actually Casabac is not really part of what you will get explained in this document - it "just" has to do with normal FOP. Please also view the FOP documentation that is available via <http://xml.apache.org/fop/>.

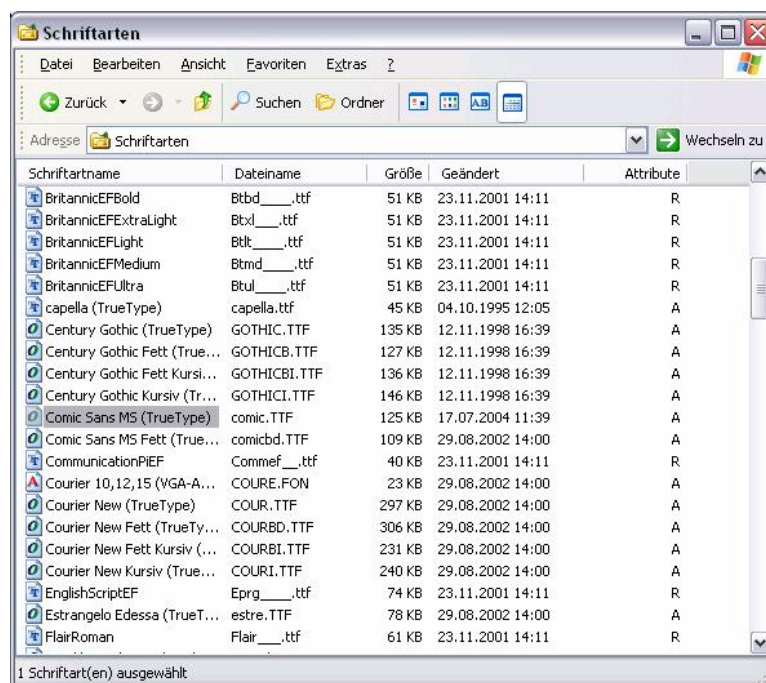
Overview

In order to use new fonts you have to...

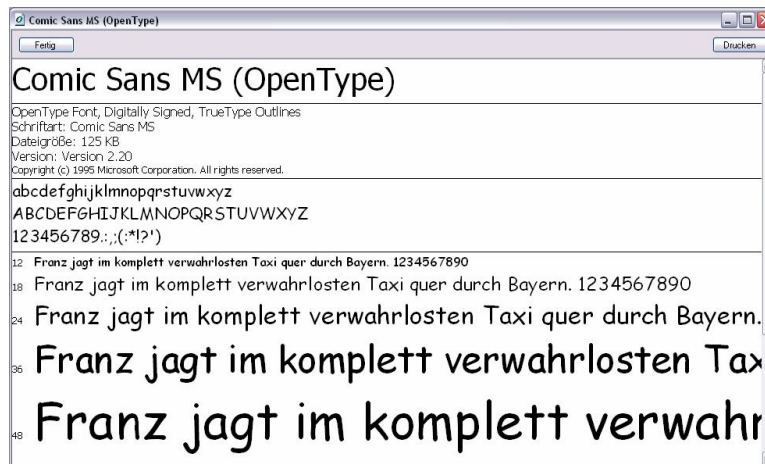
- Create a so called font metrics file.
- Embed the font into the FOP processing.
- Reference the font inside the Casabac layout definition of a PDF page.
- Make FOP use the right configuration file.

Creating the Font Metrics File

In Windows environment fonts are typically stored in ".ttf" files (true type fonts). You can see the fonts that come with the operating system by opening the fonts configuration:



If clicking e.g. on Comic Sans MS then you see how the font looks like:



The Comic Sans Font is normally not part of the fonts that are supported by default processing. We will add it as consequence. To no mix up with maybe already existing font definitions we will assign the font name "ffffffff".

Copy the comic.ttf file from the window above into a directory of your choice and rename it to "ffffffff.ttf".

```
<directory>
  ffffffff.ttf
```

Now start a Java program in your directory that creates the font metrics file from the ttf-file. The starting is done in the following way:

```
java -cp <..>\fop.jar;<..>\avalon-framework.jar org.apache.fop.fonts.apps.TTFReader ffffffff.ttf
fffffff.xml
```

Replace "<..>" with the directory in which the jar-libraries are contained. Casabac includes the libraries within its web application below the folder "WEB-INF/lib". - After running the program a file "fffffff.xml" should be created inside your directory:

```
<directory>
  ffffffff.ttf
  ffffffff.xml
```

Embed the Font into FOP Processing

The FOP processing can be configured using a certain XML configuration file. In this file you can define fonts to be embedded. Create an own configuration file, called "fffffffuserconfig.xml" inside your directory that contains the following text:

```
<configuration>
<font>
  <font metrics-file="c:\temp\fontinfo\fffffff.xml" embed-file="c:\temp\fontinfo\fffffff.ttf"
  kerning="yes">
    <font-triplet name="ffffffff" style="normal" weight="normal" />
    <font-triplet name="ffffffff" style="normal" weight="bold" />
    <font-triplet name="ffffffff" style="italic" weight="normal" />
    <font-triplet name="ffffffff" style="italic" weight="bold" />
  </font>
</font>
</configuration>
```

The font files are named with their absolute path. "C:\temp\fontinfo\" should be replaced with the location of your directory. - You could also name the files relative to the start directory of the Java application that uses FOP.

Reference the Font within your Page Definition

The font can now be used inside the XML page definition that you create for Casabac pages.

```
<casafopage2 pageheight="29.7cm" pagewidth="21cm" margin="1cm" marginbottom="0.5cm"
marginleft="1cm" marginright="1cm" headerheight="1.5cm" footerheight="1.3cm">
  <casafoclasses2>
  </casafoclasses2>
  <casafopageheader2>
    <casaforowtextblock2 text="Header" textalign="center">
    </casaforowtextblock2>
  </casafopageheader2>
  <casafopagefooter2>
    <casafoline2>
    </casafoline2>
    <casafovdst2 height="0.3cm">
    </casafovdst2>
    <casaforowtextblock2 text="Page" textalign="center">
      <casafopagenumber2 separator=" of ">
      </casafopagenumber2>
    </casaforowtextblock2>
  </casafopagefooter2>
  <casafobody2>
    <casaforowtable2 columnwidths="15cm">
      <casaforow2>
        <casafocelltext2 text="ABCDEFGHJKLMNOPQRSTUVWXYZ" fontfamily="fffffffff">
        </casafocelltext2>
      </casaforow2>
    </casaforowtable2>
  </casafobody2>
</casafopage2>
```

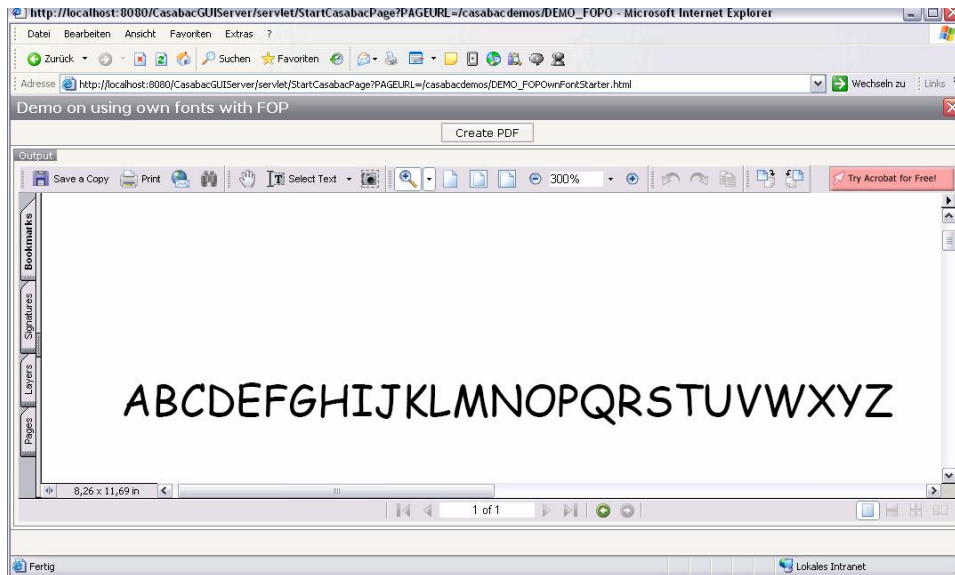
You see that in the CELLTEXT-line the font-family "fffffffff" is referenced. But - you will not immediately see the correct font because you have to make FOP aware of using the right configuration file.

Make FOP use the right Configuration File

Have a look onto the following code that is part of an adapter that creates a PDF output:

```
public void onCreatePDF()
{
  try
  {
    // read XML layout
    String fileName = Params.getApplicationDirectoryName("casabacdemos") +
      "xml/DEMO_FOPOwnFont.xml";
    String fopxml = FileManager.readFileIntoString(fileName);
    // set configuration file to be taken
    File configFile = new File("c:/temp/fontinfo/fffffffffuserconfig.xml");
    Options fopOptions = new Options(configFile);
    // create pdf
    IPDFFOPService pfs = PDFFOPServiceFactory.createPDFFOPService(fopxml, this);
    byte[] pdfBytes = pfs.generatePDF();
    m_pdfCounter++;
    m_pdfpage = findCasabacSessionContext().getSessionBuffer().
      addPDF("J0J0"+m_pdfCounter, pdfBytes);
  }
  catch (Throwable t)
  {
    {
      outputMessage(MT_ERROR, t.toString());
    }
  }
}
```

You see two lines of code in which the updated configuration file is referenced an set as the one to be used. Afterwards the pdf will be rendered with the following result:



Please pay attention: you should only use one and the same FOP configuration file within your application! If multiple users are processed on the server in parallel threads setting different configuration files then "the last one will win".