
使用Apache FOP将Docbook文档转换成PDF

花费了两天时间，总算把Apache FOP转换Docbook文档到PDF的环境搭建起来了，期间走了不少弯路。

本文简单介绍一下在Debian Lenny上搭建FOP转换环境的过程。下一篇文章介绍一下如何在Windows上搭建FOP环境。另外还会再写一篇文章，列出我在搭建环境过程中遇到的问题以及解决方法，希望对遇到类似问题的人有帮助。

按下面的步骤搭建环境，你可以在这里 [http://easwy.com/blog/uploads/2009/04/fop_conf_easwy.zip] 下载到Easwy的FOP配置及示例文件。

1. 安装相关软件

在Debian上安装软件是非常容易的事，只要下面一条命令，就可以把Docbook相关的软件全部安装完成。

```
aptitude install docbook-xml docbook-xsl xsltproc fop
```

上面的命令会把它们所依赖的软件包也一并装好。安装完成后，主要软件所在的目录如下：

- docbook-xsl - /usr/share/xml/docbook/stylesheet/nwalsh/，这是Docbook的XSL转换样式表，Lenny中带的是1.73.2版本，最新版本为1.74.3。不过我觉得Lenny中的已经够用了
- xsltproc - /usr/bin/xsltproc，这个是Docbook转换工具，用C语言写的，转换速度非常快
- fop - /usr/bin/fop，Apache的FOP转换工具，主要用来把xsltproc输出的fo格式文件，转换成PDF文件，Lenny中带的是0.94，最新版本是0.95

2. 设置FOP

首先创建一个目录来存放FOP的配置信息，在这里我创设使用/etc/fop/目录。

FOP已经能够很好的支持中文，但是如果想让FOP使用Type1类型字体以及TrueType类型字体的话，还需要为这些字体生成Metrics文件，并在FOP的配置文件中设置。

在Debian Lenny中提供了一个名为/usr/bin/fop-ttfreader的脚本，它其实只是对org.apache.fop.fonts.apps.TTFReader的封装，关于org.apache.fop.fonts.apps.TTFReader程序的用法我在下篇文章里讲，你也可以参考文尾的第二篇参考文档。

在Debian Lenny中配置中文 [http://easwy.com/blog/archives/chinese_in_debian_lenny/] 一文中，我已经把Windows下的宋体拷贝到了Lenny中，要想FOP把这个字体嵌入到PDF文档中，需要使用/usr/bin/fop-ttfreader来生成TTF字体的Metrics文件。

我的字体文件为/usr/local/share/fonts/msfonts/simsun.ttc，先用下面的命令看看这个字体集合中有几个字体：

```
/usr/bin/fop-ttfreader /usr/local/share/fonts/msfonts/simsun.ttc /etc/fop/simsun.xml
TTF Reader for Apache FOP 0.94
```

```
Parsing font...
Reading /usr/local/share/fonts/msfonts/simsun.ttc...
This is a TrueType collection file with 3 fonts
```

Containing the following fonts:

SimSun

NSimSun

SimSun-PUA

Error while building XML font metrics file.

java.lang.IllegalArgumentException: For TrueType collection you must specify which font to select

忽略上面显示的异常，我们的目的只是看这个字体集中有几种字体，从上面的输出中可以看到有三个字体：SimSun、NSimSun和SimSun-PUA。

好，接下来我们为SimSun字体生成一个Metrics文件，在刚才那条命令中加上参数-ttcname SimSun参数(注意，字体名区分大小写)，再执行一次，这一次输出如下：

```
/usr/bin/fop-ttfreader -ttcname SimSun /usr/local/share/fonts/msfonts/simsun.ttc /etc/fop/simsun
TTF Reader for Apache FOP 0.94
```

Parsing font...

Reading /usr/local/share/fonts/msfonts/simsun.ttc...

This is a TrueType collection file with 3 fonts

Containing the following fonts:

SimSun <-- selected

NSimSun

SimSun-PUA

Font Family: SimSun

Creating xml font file...

Creating CID encoded metrics...

Writing xml font file /etc/simsun.xml...

This font contains no embedding license restrictions.

XML font metrics file successfully created.

这一次执行成功了，生成的Metrics文件为/etc/fop/simsun.xml。你可以继续用这条命令为其它你想用的TTF或Type1字体生成Metrics文件。

接下来设置FOP，我是直接把FOP 0.95自带的缺省配置文件拷贝了一份到/etc/fop/fop.xconf，然后再进行修改(我下载了一个FOP 0.95，不过只用了它的配置文件 ^_^)：

首先在文件中找到<renderer mime="application/pdf">这一行，然后在此标签中加入关于字体的配置。在FOP 0.94中已经支持对字体的自动检测，不过如果你的字体不在标准字体目录里，还是需要指明字体的目录。例如，我在配置文件中加入以下设置：

```
<font>
  <font metrics-url="file:///etc/fop/simsun.xml" kerning="yes" embed-url="file:///usr/local/share/fonts/msfonts/simsun.ttc">
    <font-triplet name="SimSun" style="normal" weight="normal"/>
    <font-triplet name="SimSun" style="normal" weight="bold"/>
    <font-triplet name="SimSun" style="italic" weight="normal"/>
    <font-triplet name="SimSun" style="italic" weight="bold"/>
  </font>

  <font metrics-url="file:///etc/fop/simhei.xml" kerning="yes" embed-url="file:///usr/local/share/fonts/zhongwen/simhei.ttf">
    <font-triplet name="SimHei" style="normal" weight="normal"/>
```

```
<font-triplet name="SimHei" style="normal" weight="bold"/>
<font-triplet name="SimHei" style="italic" weight="normal"/>
<font-triplet name="SimHei" style="italic" weight="bold"/>
</font>

<directory recursive="true">/usr/share/fonts/</directory>
<auto-detect/>
</fonts>
```

上面的配置加入了SimSun和SimHei两种字体，并加入字体目录/usr/share/fonts/和/usr/local/share/fonts/，同时打开了auto-detect。FOP只在第一次运行时扫描字体，然后为之生成缓存，以后就不必再扫描了。

3. 配置XSL转换样式表

接下来，我们要配置XSL转换样式表，让xsltproc在转换Docbook文档时，使用我们指定的中文字体。在下面我给出了一个简单的样式表，假设存为docbook_fo.xsl：

```
<?xml version='1.0'?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:exsl="http://exslt.org/common"
  xmlns:fo="http://www.w3.org/1999/XSL/Format"
  xmlns:ng="http://docbook.org/docbook-ng"
  xmlns:db="http://docbook.org/ns/docbook"
  exclude-result-prefixes="db ng exsl"
  version='1.0'>

  <xsl:import href="/usr/share/xml/docbook/stylesheet/nwalsh/fo/docbook.xsl"/>
  <xsl:param name="body.font.family">SimSun</xsl:param>
  <xsl:param name="monospace.font.family">SimSun</xsl:param>
  <xsl:param name="title.font.family">SimHei</xsl:param>
</xsl:stylesheet>
```

在这个转换样式表里，首先导入了Docbook的FO样式表，然后重新设置了其中的字体参数，改成我们所需要的字体名称。需要注意的是，这里的字体名字需要和/etc/fop/fop.xconf中设置的font-triplet中的名字完全一致。

我之所以配置两天才配好FOP，就是因为在这里犯了个错误。我最初的docbook_fo.xsl中，在导入FO样式表时，使用的是<xsl:include ...>，它和<xsl:import ...>的区别在于，使用import导入时，先使用本文件中参数设置，当本文件中未对参数进行设置时，才使用所导入文件中的设置；但使用include导入却不是这样。所以，即使我在docbook_fo.xsl中对输出字体进行了重新设置，但实际上我的修改未生效，xsltproc输出的.fo文件中字体用的根本不是我所指定的。最终的结果就是文章中的中文，都被显示成了###。

4. 输出PDF文件

上面的配置完成以后，就可以输出PDF文件了。只需要使用下面两条命令即可：

```
/usr/bin/xsltproc -o render_docbook_xml_to_pdf_by_apache_fop.fo docbook_fo.xsl render_docbook.xml
/usr/bin/fop -c /etc/fop/fop.xconf render_docbook_xml_to_pdf_by_apache_fop.fo -pdf render_docbook.pdf
```

上面两条命令会有大量的错误提示，暂时不管它，去看看输出的pdf文档吧，尽管还有诸多不满意，不过总算能输出出来了。接下来需要解决的问题还很多，比如如何分别为中英文指定字体族、如何更好的断字等，留待以后慢慢解决。

参考文档

- Apache FOP: Configuration [<http://xmlgraphics.apache.org/fop/0.95/configuration.html>]
- Apache FOP: Fonts [<http://xmlgraphics.apache.org/fop/0.95/fonts.html>]
- 使用Apache FOP将Docbook转换成PDF文档 [<http://blog.chinaunix.net/u/1096/showart.php?id=1808039>]
- 中文docbook转换成PDF终于实现 [<http://gentoogle.blogspot.com/2008/03/docbookpdf.html>]