Java实现HTML页面转PDF解决方案

[Java](http://www.open-open.com/lib/tag/Java)   2012-07-10 09:00:12 发布

|  |  |  |  |
| --- | --- | --- | --- |
| **您的评价**: |  | **4.5** |  |

[收藏](javascript:void(0))     3收藏

最近在工作中遇到了一个很囧的事情。要把HTML页面转换成PDF或者tiff图片等其他形式，目前tiff还是一头雾水，如果各位大师有真的可行的想法（不要是别人那边抄的，百度最近搜索的质量严重的打击了我对国货的信心。）希望能够告诉我，先谢了。  
言归正传。所以，我就做了把HTML转换成PDF的形式。目前可以完美支持一比一转换，下面就把我的解决方案贴出来。  
首先，当然是找到能够解析PDF的完美组件，百度和谷歌不约而同的告诉我们。IText是王道。而目前开源的组件中，Itext的确是一个First Choice，如果各位单纯是做把图片转成PDF或者自己写了Velocity或者FreeMarker模板生成了HTML是非常推荐直接用Itext来进行的。而如果，大家像我这样已经有前人写好了HTML页面或者懒得写FreeMarker模板的话。可以直接看下一段。  
由于他们已经写好了HTML页面，而且显示已经很完美了。那我要做的就是能完美解析HTML+CSS的PDF生成工具。这时候flying-saucer进入了我的选择范围中。  
<http://code.google.com/p/flying-saucer/>  
上面是网址，这个工具托管在GoogleCode上面，作者做他们能够做下面的工作：  
Flying Saucer takes XML or XHTML and applies CSS 2.1-compliant stylesheets to it, in order to render to PDF (via iText), images, and on-screen using Swing or SWT。  
不难看出工作原理，就是解析XML或者XHTML并且包括css样式表，并且用Swing或者SWT的组件生成PDF的功能。这解决了页面的显示问题。IText自身的一个很严重的问题就是解析CSS有很大的问题。而这个解决了。下面就是用Flying Saucer来实现的代码：

public boolean convertHtmlToPdf(String inputFile, String outputFile)

throws Exception {

OutputStream os = new FileOutputStream(outputFile);

ITextRenderer renderer = new ITextRenderer();

String url = new File(inputFile).toURI().toURL().toString();

renderer.setDocument(url);

// 解决中文支持问题

ITextFontResolver fontResolver = renderer.getFontResolver();

fontResolver.addFont("C:/Windows/Fonts/SIMSUN.TTC", BaseFont.IDENTITY\_H, BaseFont.NOT\_EMBEDDED);

//解决图片的相对路径问题

renderer.getSharedContext().setBaseURL("file:/D:/");

renderer.layout();

renderer.createPDF(os);

os.flush();

os.close();

return true;

}

上面这段代码是这样的，输入一个HTML地址URL = inputFile，输入一个要输出的地址，就可以在输出的PDF地址中生成这个PDF。

注意事项：

1.输入的HTML页面必须是标准的XHTML页面。页面的顶上必须是这样的格式：

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

并且HTML页面的语法必须是非常严谨的，所有标签都必须闭合等等（由于flying-Saucer做了XML解析的工作，不严谨会报错的。），这是对页面的第一个要求。

2.要用到图片的地方写相对路径的形式，比如：

<img src="a.jpg" alt="323" width="252" height="80" />

而它的图片位置则必须在Java代码中指定。

renderer.getSharedContext().setBaseURL("file:/D:/");

也有另一种方法就是直接在<img>标签中写绝对路径。

3.Flying-Saucer在解析tiff格式的图片的时候会报错。具体原因我还没找到。希望大家能够指点我。

4.如果在页面中有中文字体的话。必须在HTML代码中的样式中写上某种字体的css，并且必须是用英文的，然后在Java代码中写上对应的文件位置。

ITextFontResolver fontResolver = renderer.getFontResolver();

           fontResolver.addFont("C:/Windows/Fonts/SIMSUN.TTC", BaseFont.IDENTITY\_H, BaseFont.NOT\_EMBEDDED);

上面的方法是添加了宋体。也可以添加其他字体。

以上就是解决方案。

下面给出这几个包的下载地址。大家可以直接下载。

[下载地址，猛点我吧！](http://download.csdn.net/detail/jasonchris/4403777)

转自：http://blog.csdn.net/jasonchris/article/details/7705631