在Web应用中，有时需要按照固定的模板将数据导出到Word，如流程审批单，在流程处理完成后将处理过程按照流程单的要求导出，有时程序中需要实 现生成标准Word文档，要求能够打印，并且保持页面样式不变，常见的方案有POI、iText、JACOB、JSP几种方式，POI读取Word文档比 较适合、对于生成文档样式比较难控制，iText操作Excel还可以，对Word的操作功能有限，JACOB操作Word实现复杂，并且无法将服务部署 到Linux平台，要求安装office，对于实现固定格式的报表实现困难，对于JSP直接输出方式样式控制难。

Word从2003开始支持XML格式，用XML+Freemarder还做就很简单了，大致的思路是先用office2003或者2007编辑好 word的样式，然后另存为xml，将xml翻译为FreeMarker模板，最后用java来解析FreeMarker模板并输出Doc。经测试这样方 式生成的word文档完全符合office标准，样式、内容控制非常便利，打印也不会变形，生成的文档和office中编辑文档完全一样。具体实现过程如 下：

1、  首先用office【版本要2003以上，以下的不支持xml格式】编辑文档的样式，将需要动态填充的内容使用Freemarker标签替换：Word文档样式如下：



2、  将Word文档另存为XML格式，将后缀名“xml”修改为“ftl”

3、  使用Freemarker填充内容，代码如下：

package com.test.freemarker.report;

import java.io.BufferedWriter;

import java.io.File;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.OutputStreamWriter;

import java.io.Writer;

import java.util.HashMap;

import java.util.Map;

import freemarker.template.Configuration;

import freemarker.template.Template;

import freemarker.template.TemplateException;

public class DocumentHandler {

   private Configuration configuration = null;

   public DocumentHandler() {

      configuration = new Configuration();

      configuration.setDefaultEncoding("utf-8");

   }

   public void createDoc() {

      // 要填入模本的数据文件

      Map dataMap = new HashMap();

      getData(dataMap);

      // 设置模本装置方法和路径,FreeMarker支持多种模板装载方法。可以重servlet，classpath，数据库装载，

      // 这里我们的模板是放在com.havenliu.document.template包下面

      configuration.setClassForTemplateLoading(this.getClass(),

            "/com/test/freemarker/report");

      Template t = null;

      try {

         // test.ftl为要装载的模板

         t = configuration.getTemplate("test.ftl");

         t.setEncoding("utf-8");

      } catch (IOException e) {

         e.printStackTrace();

      }

      // 输出文档路径及名称

      File outFile = new File("D:/test.doc");

      Writer out = null;

      try {

         out = new BufferedWriter(new OutputStreamWriter(

                new FileOutputStream(outFile), "utf-8"));

      } catch (Exception e1) {

         e1.printStackTrace();

      }

      try {

         t.process(dataMap, out);

         out.close();

      } catch (TemplateException e) {

         e.printStackTrace();

      } catch (IOException e) {

         e.printStackTrace();

      }

   }

   /\*\*

    \* 注意dataMap里存放的数据Key值要与模板中的参数相对应

    \*

    \* @param dataMap

    \*/

   private void getData(Map dataMap) {

      dataMap.put("title\_name", "用户信息");

      dataMap.put("user\_name", "张三");

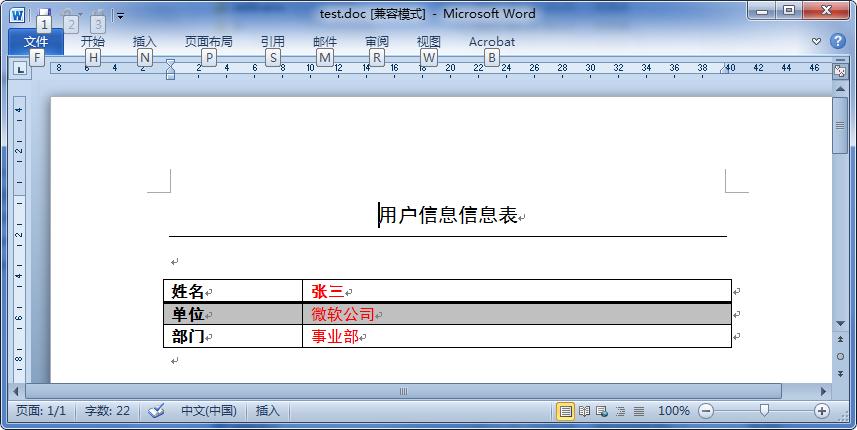
      dataMap.put("org\_name", "微软公司");

      dataMap.put("dept\_name", "事业部");

   }

}

  
4、  生成的Word结果如下：



对于复杂的报表样式可以在Word中编辑后保存，如果需要输出列表类型数据可以参考Freemarker的循环或逻辑控制。