DOI:10.16101/j.cnki.cn32-1749/z.2010.04.020

基于XML的Excel数据提取和集成研究

夏开建[1],严小泉[2]

(1.常熟理工学院 计算机科学与工程学院,江苏 常熟 215500;
2.江苏国光信息产业股份有限公司,江苏 常州 213015)

摘 要:异构数据源的集成问题是当前数据处理领域内研究的一个热点,它能更有效地利用信息资源、更好地实现数据共享.针对当前企业对异构数据库中Excel数据源集成的迫切需求,本文提出一个基于XML模板的Excel数据源数据映射方案,能使数据转换更加灵活通用.该方案是实现异构数据源之间数据交换的通用方法,实现了Excel数据向标准XML文件的相互转化,有效提高了整个应用系统的性能.

关键词:Excel数据源;XML模板;数据映射;异构数据源
中图分类号:TP391 **文献标识码**:A **文章编号**:1008-2794(2010)04-0116-05

随着Internet技术的飞速发展和网络计算模式的广泛应用,异构数据源间的数据集成和转换日益频繁.异构数据源在多个应用系统中的格式、语义和层次不同,导致整个企业数据的不一致性.面对残酷的竞争和频繁的合并与收购行为,许多企业都在力图解决数据碎片所带来的问题,整合这些支离破碎的异构数据源是企业之间或企业内部各部门之间协同合作的需要,超过30%的IT预算被用于构建和维护遗留系统间Excel数据源的集成.

Excel简单易用,其丰富的格式控制和数据处理能力对各种信息都十分适合,比如姓名清单、产品清单、金融数据等等,也是保存统计数据的最流行的电子表格格式.但与关系数据库或XML数据库相比,Excel在数据集成、数据查询、数据分析、数据冗余等方面明显不足.许多公司和行业使用Excel来准备、编辑和保存数据,但是它不适合后期查询分析与统计,因此Excel数据源的集成和转换迫在眉睫.Java提供的跨平台语言和XML提供的跨平台数据格式的完美结合将成为最佳的Excel数据集成的解决方案.本文利用JAVA技术、使用XML[1](Extensible Markup Language)作为Excel数据的模板配置数据和转换格式,实现Excel数据与XML数据的相互转换,降低了程序开发难度和开发成本,实现异构数据源信息系统的无缝集成.

1 Excel数据转换方法

1.1 Excel数据转换现状

企业中历史遗留系统中Excel数据量通常情况下非常大,纯人工以单元格为单位将Excel数据转换成XML数据或导入到关系数据库中代码量繁重、可靠性差,代价极其巨大.因此考虑用JAVA来操作Excel,将数据转换成异构数据标准信息XML,该技术编程量小、准确度高且便于维护和集成处理,Excel和其它数据源集成[2]和转

收稿日期:2010-02-04
作者简介:夏开建(1983—),男,江苏宿迁人,常熟理工学院计算机科学与工程学院助教,硕士,研究方向:计算机图形学、图形图像处理.

换如图1所示，其中每个数据源对应一个包装器，由包装器来与其封装的数据源交互，提取各本地数据源的XML元数据，最终经过数据清除、数据集成等操作融合数据，达到异构数据源集成的目的，本文只讨论Excel数据源与XML数据的相互转换.

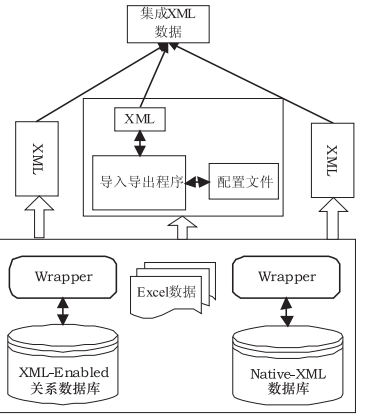

图1 Excel和其它数据源集成与转换

在Web应用日益盛行的今天，通过Web来集成转换Excel文件的需求越来越强烈. 目前较为流行的处理Excel数据的方法主要有三种：

(1)不操作实际的Excel文件，而是在JSP或Servlet中创建一个CSV[3](comma-separated value)文件，CSV是用来交换电子表格文件的常用格式，任何适当的电子表格都可以通过 CSV文件导出和导入，它在头文件中以application/vnd.ms-excel类型返回给浏览器，接着浏览器调用Excel显示或者下载Excel文件，但这不能算是真正意义上的操作Excel文件.

(2)利用第三方工具来实现Excel数据与关系数据库表文件的导入导出. 例如：Microsoft公司的SQLServer 2005提供的导入导出辅助工具，能够将格式比较简单的Excel数据自动导入到关系数据库中或者将数据库表记录导出到Excel文件中；开源数据库MySQL的辅助软件MySQL-Front也提供了Excel文件导入导出功能. 但是此类工具的功能有限，只能实现行列规范的Excel数据和关系数据库表文件的转换.

(3)利用Java Excel API操作和转换Excel，转换成XML文件使它可以运行于任何平台，并且很容易地实现异构数据源集成；格式复杂的图表输出，如表1(单元格合并、对齐、字体格式)所示：

表1 复杂格式的表格

出版商	作者	编号	书名	价格	出版日期
北京出版社	Arman Danesh	7-3020-285-67	Java开发指南	45	2000-3-24
机械工业出版社	Bruce Eckel	7-111-16220-X	Java编程思想	95	2005-5-10
电子工业出版社	Zapawa.T	7-121-02137-4	Excel高级报表宝典	38	2006-1-16
北京出版社	陈为国	2-1139-0160-7	C++入门教程	33.5	2000-5-13

1.2 JXL和JDOM介绍

JXL[4] JAR包，是一个开放源码项目Java Excel API，通过它就可以方便地动态读取Excel文件的内容、创建新的Excel文件、更新已经存在的Excel文件.

JDOM是一个开源项目的Java XML API，它基于树型结构，利用纯JAVA技术对XML文档实现解析、生成、序列化以及多种操作，有效地结合SAX和DOM的优点，弥补了DOM及SAX在实际应用当中的不足，隐藏操作XML过程中的复杂性，JDOM的结构如图2所示：

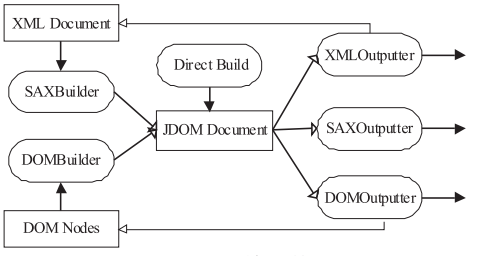

图2 JDOM的结构图

JXL和JDOM提供了Excel文件与XML文件转换的真正Java执行过程. Excel与XML数据的转换策略都依赖于具体的应用、根据具体问题编写的导入导出程序.

2 Excel数据-XML数据解决方案

2.1 基于XML的Excel信息描述

XML是W3C的XML工作组为适应Internet的发展提出的半结构化可扩展标记语言. XML具有开放性、良好的扩展性、自描述性、跨平台等特征，已逐渐成为Internet上数据表示与交换的标准.

通过XML配置文件实现Excel数据与XML文件的转换，使其更容易扩展，能适应各种复杂环境，基于XML的Excel数据转换主要有以下几个优点：

(1)引入XML配置文件使导入程序与Excel数据具体格式相互独立.

(2)引入JDOM使得解析、生成XML文档更为容易.

(3)生成的XML文件为异构数据源集成、查询、分析带来极大地便利.

2.2 XML配置文件设计

Excel文件包含两类数据:一是表头、列名等静态数据;二是动态变化的数据,此数据是通过程序循环读取并写入到XML文件中.

在Excel文件数据转化为XML的过程中,首先要针对该Excel文件定义一个XML配置文件,使导入导出程序与具体的Excel数据独立,在Excel与XML之间形成一个语义层[5],降低了它们之间的耦合度.

配置文件(excelmapping.xml)的主要代码如下所示:

```
<excel-mapping>
<class classname="cn.edu.sytu.ExcelBean">
<column excelCol="出版商" name="publisher">
<attribute excelCol="电话" colIndex="1" name="phone" type="LABEL" colwidth="13" />
<value excelCol="出版商" colIndex="2" name="publisher" type="LABEL" colwidth="15" />
</column>
<column excelCol="书名" name="bookname">
<attribute excelCol="作者" colIndex="3" name="author" type="LABEL" colwidth="13" />
<attribute excelCol="编号" colIndex="4" name="ISBN" type="LABEL" colwidth="14" />
<value excelCol="书名" colIndex="5" name="bookname" type="LABEL" colwidth="17" />
</column>
<column excelCol="价格" colIndex ="6" value="price" type="NUMBER" colwidth="10" />
<column excelCol="出版日期" colIndex="7" value="date" type="DATE" colwidth="11" />
</class>
</excel-mapping>
```

其中class:对应Excel的配置信息,classname存储Excel每一行数据的类;attribute:对应XML中父节点column的值信息,excelCol对应为Excel文件中的表头列名,colIndex对应为Excel中的列索引值,name对应ExcelBean的属性,type为Excel数据的类型,在本文的实验数据中只有(LABEL,NUMBER,DATE)这三种类型的数据,colwidth为Excel单元格的宽度. value:对应为XML中父节点column的值信息,其各属性值意义同attribute的属性意义.

2.3 通过Java从Excel文件中读取数据

读取Excel时首先读取excelmapping.xml,取出映射类和属性的映射关系. 循环Excel中每一行的每一列,取出每列的列名,在映射关系中查找与之对应的类属性,利用JAVA的反射机制将列数据写入类属性中,读完每行数据之后将类写入集合中即可得到与Excel相对应的类集合. 先读取配置文件到内存中,然后循环columns获取它的子元素和属性,保存到ExcelManager(String classname,Map propertyMap)类中的相应属性,即可完成配置文件的读取工作. Excel中的文件、工作表、行、单元格对应于JXL中的Workbook、Sheet、Cell数组、Cell. Java Excel API既能读取本地文件系统的一个文件,又能从输入流中读取Excel数据表. 读取Excel数据表的步骤如下:

(1)获取Workbook(工作簿)

(2)操作Excel Sheet(工作表)

(3)操作Cell(单元格)

```
 for (int i = 1; i < sheet.getRows(); i++) {//得到该sheet的行数
for (int j = 0; j < sheet.getColumns(); j++) {//得到该sheet的行数
Cell cell = sheet.getCell( j,i ); // 得到第j列第i行的单元格
```

```
String excelCol=excelTitles[j];
String name=columnmap.get(excelCol).get("name");
String mname= methods[i].getName().substring(3).toLowerCase();//截取的方法名
String cont0=null; double cont1=0.00; Date cont2=null;
Class a=Class.forName(columnmap.get("classname"); //获取映射类
Method [] methods=a.getDeclaredMethods();
Object o=a.newInstance();//生成ExcelBean类的实例对象
for(int i=0;i<methods.length;i++)
{//根据单元格的类型以及Java反射机制来设置ExcelBean的属性值
    if (mname.equals(name) && cell.getType() == CellType.DATE){
DateCell datec = (DateCell) cell;
        cont2 = datec.getDate();
        methods[i].invoke(o, cont2); //调用setXXX()方法 }
    if (mname.equals(name) && cell.getType() == CellType. NUMBER){
  NmberCell numc = (NumberCell) cell;
    cont1= numc10.getValue();
        methods[i].invoke(o, cont2); }
    if (mname.equals(name) && cell.getType() == CellType. LABEL){
LabelCell labelcell = (LabelCell)cell;
        cont0= labelcell.getString();
        methods[i].invoke(o, cont2); } }}}
```

循环Excel每行的单元格即可得到其对应的一个ExcelBean对象,之后将其添加到List javabens集合中,这样就能完成Excel数据与JavaBean属性的转换. CreateXML(List javabeans)方法利用JDOM将JavaBeans转换为XML,publisher元素节点的构建代码如下所示:

```
Element publisher=new Element("publisher");//构建book的子元素publisher
publisher.setAttribute("phone","010-62752979");//设置属性
publisher.setText("北京出版社");//设置publisher的文本值
book.addContent(publisher);//添加子节点
  root.addContent(book);
XMLOutputter XMLOut=new XMLOutputter(format);
XMLOut.output(doc,new FileOutputStream("C:\\books.xml"));
```

本文采用book元素描述Excel每行数据,并作为根元素books的一个子元素,books将包含一个或多个book元素,最后生成的XML文件.

2.4　将XML生成新的Excel文件

首先读取excelmapping.xml配置文件,取出映射类和属性的映射关系. 将映射文件中配置的excelCol写入Excel的第一或第二行并根据配置文件中的column合并单元格,然后利用JDOM解析books.xml,将元素book的子元素(Element)的属性(Attribute)和值(value)依次写入到Excel的行中.

文中涉及字符串的格式化(字体、粗细、字号等元素设置),这些功能主要由WritableFont和WritableCellFormat类来负责. 单元格的操作主要涉及行高设置、列宽设置、单元格合并,合并既可以是横向的,也可以是纵向的,合并后的单元格不能再次合并,否则会触发异常. 将(0,0)到(1,0)和(5,0)到(5,1)的单元格合并的代码如下所示:WritableSheet.mergeCells(0,0,1,0), WritableSheet.mergeCells(5,0,5,1).

当执行完所有转换操作之后,必须要先调用write()方法,因为先前的操作都是存储在缓存中的,所以要通过该方法将操作的内容保存在文件中.

其中解析books.xml将每个book元素中的publisher节点文本值、phone、bookname节点文本值、author、price、date插入到Excel的每行中主要由方法Insert(WritableWorkbook wwb,String xmlpath)来完成,最终生成Excel文件.

3 结束语

本文所实现的功能是利用开源项目组件JDOM和JXL、基于XML配置文件进行异构数据库系统间的数据交换的探索,JDOM和JXL的结合使用使得提取Excel数据变得更简练、更灵活、更有效,具有较强的通用性和扩展性. 本方法的不足之处在于JDOM创建大于10M的XML文档时容易造成内存溢出,此问题在今后的研究工作中有待进一步完善和解决.

参考文献:

[1] W3C. Extensible Markup Language (XML) [EB/OL]. http://www.w3.org/TR/xml11/, 2006-08.

[2] 罗作民,李悦,孙淑海. 基于Excel及数据转换服务的异构数据集成方法[J]. 计算机应用,2007,27(3):574-578.

[3] 李少军,夏红霞,詹芹. 基于Java技术的Web环境下Excel的应用[J]. 微机发展,2005,15(7):114-117.

[4] JXL. Java Excel API (XML)1.1 [EB/OL]. http://www.andykhan.com/jexcelapi/, 2003-07.

[5] 王喆,郭艳军. 基于XML元数据和Schema的Excel信息提取研究[J]. 计算机工程与应用,2008,44(43):135-138.

Excel Data Extraction and Integration Based on XML

XIA kai-jian[1], *YAN xiao-quan*[2]

(1.School of Computer Science and Engineering, Changshu Institute of Technology, Changshu 215500, China;
2.Jiangsu GuoGuang Electronic Information Technology Co, Ltd, Changzhou 213015, China)

Abstract: The integration of heterogeneous data sources is a hotspot in the field of current data processing research which can make effective use of the information sources and share data better. In view of the urgent demand of the current enterprise for integration of Excel data sources in heterogeneous data systems, an Excel data mapping scheme is proposed based on XML template, making the data conversion more flexible and general. The program is an effective exploration to achieve heterogeneous data sources for data exchange between the common methods, and it is also a forceful exploration realizing the data exchange between heterogeneous data sources and bringing the data transformation between Excel data and the standard XML documents into effect, which effectively improve the performance of the entire application system.

Key words: Excel data sources; XML template; data mapping; heterogeneous data sources