

文章编号: 1005—0523(2004)04—0082—04

使用 java 技术实现异构系统间的数据交换的探索

王海霞¹, 俞卯君²

(华东交通大学 1. 现代教育技术中心; 2. 国际合作交流处, 江西 南昌 330013)

摘要: 阐述在 Enhydra 平台上, 使用 java 技术实现异构系统间的数据交换的方法, 并给出了相应的实例.

关 键 词: Enhydra; java; Excel 文件; 文本文件; 数据库

中图分类号: TP311.1 文献标识码: A

0 引 言

在我们日常所搜集、整理的数据中, 有很多是 Excel 格式的文件和文本格式的文件. 但是, 随着数据量的不断增大, 我们希望能将这些数据保存到一个标准的数据库中去, 以便于对数据进行更好的管理. 这就需要将 Excel 格式的文件和文本格式的文件导入到数据库中去.

在开发华东交通大学的校友通讯系统时, 由于数据是从教务系统中导出的 Excel 文件或文本文件, 而我们的后台数据库使用的是标准的数据库, 而且只是需要部分字段, 因此也需要解决类似的问题, 实现不同数据库系统之间的数据交换.

使用 java 语言通过 web 方式在 Enhydra 平台上实现了这一功能.

1 数据交换的解决方案

数据从教务系统中导出后, 使用磁盘或网络的方式传至校友通讯系统的管理人员手中. 他只要将该数据文件提交到 web 页面上, 点击“导入”, 即可自动实现数据的导入.

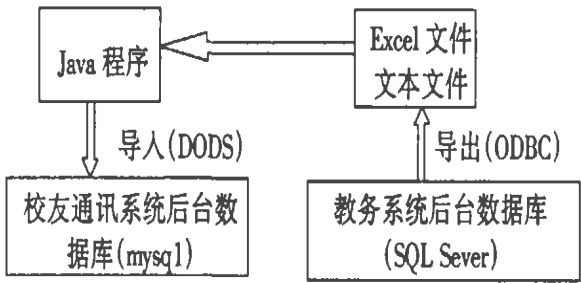


图 1 数据交换的实现流程

2 Enhydra 简介

Enhydra 是一种以 Java 语言为基础的可进行 WEB 开发的应用程序服务器.

2.1 Enhydra 的特点

- 1) 可跨平台使用: Enhydra 是纯 Java 的应用程序, 保证了在各种平台下的可移植性.
- 2) 高度的灵活性和可扩展性: Enhydra 利用 XML 和 Java 技术, 将 Web 页面的处理与逻辑处理分离开来, 为 Web 项目的开发提供了一种更加方便的模式. 同时, 增强了 Web 项目的可维护性, 为实现功能模块的扩展提供了良好的接口.
- 3) 开放源码 (Open Source): 开放源码的特点使得 Enhydra 能够在全世界众多开发者的帮助下不断发展.

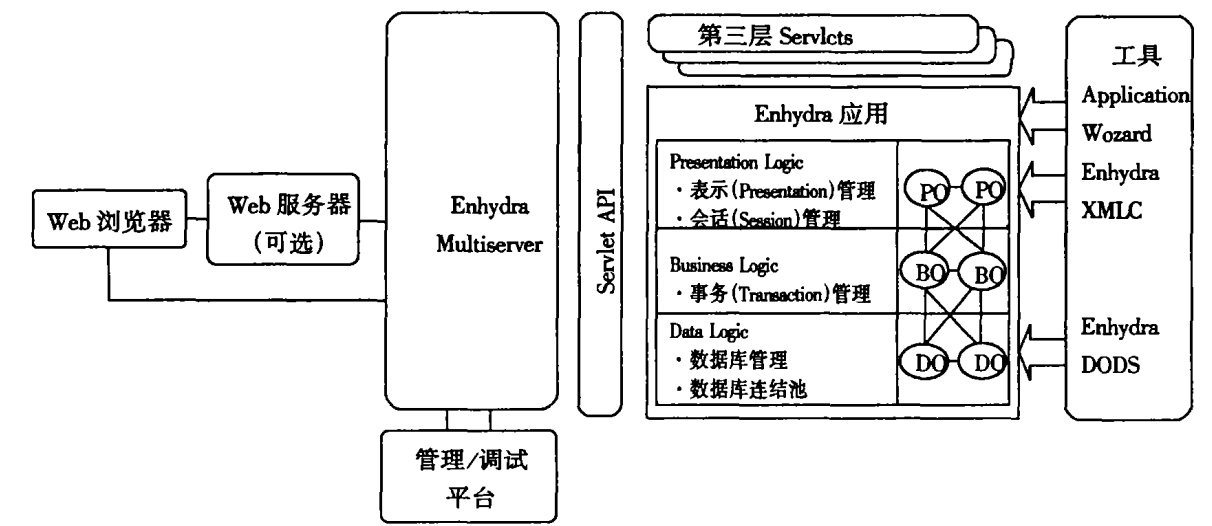


图 2 Enhydra 应用程序服务器的体系结构

2.2 Enhydra 应用程序服务器的体系结构(图 2)

- 1) Multiserver/Web server: 主要提供 Enhydra 应用程序执行的环境.
- 2) 应用程序框架 (Application Framework): 负责对页面、逻辑以及数据对象的处理. 作为一个 Web 项目的开发人员, 主要任务就是构建应用程序框架.
- 3) 各种工具程序: 帮助开发人员加快开发应用程序的速度.

3 JAVA EXCEL API

笔者是使用 Java Excel API 来对 Excel 文件进行操作的.

Java Excel 是一个开放源码的项目. Java 开发人员可以通过它来对 Excel 文件进行操作, 包括读取 Excel 文件的内容、创建新的 Excel 文件、更新已经存在的 Excel 文件. 由于 Java 语言的跨平台性, 在非 Windows 平台上也可以使用该 API 来操作 Excel 文件. Java Excel 为我们使用 Java 语言操作 Excel 文件提供了良好的接口. 在这里我们只用到了读取 Excel 文件的功能, 以下对其使用方法做简单说明.

第一步: 创建工作簿 (Workbook)

```
import java.io.*;  
import jxl.*; // 导入 Java Excel API 包
```

```
try  
{  
    // 构建只读 Workbook 对象  
    // 方法 1: 直接从本地文件创建 Workbook
```

```
jxl.Workbook rwb = Workbook.getWorkbook(new File(sourcefile));  
// 方法 2: 从输入流创建 Workbook  
InputStream is = new FileInputStream(sourcefile);  
jxl.Workbook rwb = Workbook.getWorkbook(is);  
}  
catch (Exception e)  
{  
    e.printStackTrace();  
}
```

第二步: 访问 Excel 工作表 (Sheet)

```
Sheet rx=rwb.getSheet(0); //获取第一张 Sheet 表
```

第三步: 访问 Excel 单元格 (Cell)

```
//获取第一行, 第一列的值  
Cell c00=rs.getCell(0, 0);  
String strc00=c00.getContents();  
//获取第一行, 第二列的值  
Cell c10=rs.getCell(1, 0);  
String strc10=c10.getContents();
```

注意:

- 1) getCel() 方法中的第一个参数是列号, 第二个参数是行号, 这与我们平时的习惯有所不同.
- 2) 我们用 getContents() 方法得到的单元格的值都是 String 类型的. 若要得到单元格内容的确切类型的值, 可以在得到 Cell 对象后, 通过 getType() 方法可以获得该单元格的类型, 然后与 API 提供的基本类型相比较, 强制转换成相应的类型, 最后调用相应的 getXXX() 方法, 从而得到确定类型的值.

例: 示例代码中 strc10 为数字型, 则可通过以下语句得到 strc10 的值
double strc10=0.00;

```
Cell c10=rs.getCell(1,0);
if(c10.getType()==CellType.NUMBER){
    NumberCell numc10=(NumberCell)c10;
    stre10=numc10.getValue();
}

第四步:操作完成,关闭对象,释放占用的内存空间
rwb.close();
```

4 Java 中对文件的读写操作

在 JDK1.4 以上的版本中,我们常用的对文件进行读写操作的类是 FileReader、FileWriter 以及 FileInputStream、FileOutputStream。其中,FileReader、FileWriter 可以用来对文本文件进行操作,而 FileInputStream、FileOutputStream 既可以对文本文件进行操作,也可以对二进制文件进行操作。

```
Java 中读取文本文件中的内容:
FileReader fr = new FileReader(file); //创建读文件对象

BufferedReader br = new BufferedReader(fr); //创建读文件缓冲区

String record = new String();
record = br.readLine(); //读取文本文件中的一行
```

5 主要的功能实现

要实现 web 方式的数据导入,首先要将你要操作的 Excel 文件或文本文件,通过 form 表单上传到服务器上一个特定的目录中,然后再对该文件进行操作,将数据导入到数据库中。

我定义了一个 Import 类,文件的上传、Excel 文件的导入、文本文件的导入这三个功能我分别写成了 Import 类的三个静态方法,以下为源代码:

5.1 文件的上传

```
//将文件上传至服务器,并得到它在服务器上的文件绝对路径

public static String getFilePath( HttpPresentationComms comms)
throws
    Exception {
    .....
    return newfile;
}
```

5.2 Excel 文件的导入

```
//导入学生信息
public static int importStudent1(String file, HttpPresentationComms
```

```
comms) throws
    Exception {
    int cc = 0;
    //得到 Excel 文件中的工作表
    Workbook workbook = Workbook.getWorkbook( new File
(file));
    Sheet rs1 = workbook.getSheet(0);
    int maxRow=rs1.getRows();
    String now=DateFormat.getDateInstance().format( new java.util.Date());
    Enhydra.getLogChannel().write(Logger.DEBUG, “开始导入学生记录;” + new java.util.Date());
    for (int i=1;i<=maxRow-1;i++) {
        //读取 Excel 工作表中单元格中的数据
        Cell a1 = rs1.getCell(0, i); //证书编号,getCell(列号,行号)
        String gradu_id =a1.getContents();
        Cell a2 = rs1.getCell(1, i); //姓名
        .....
        .....

        //将数据写入数据库(此处用到了使用 Enhydra 的 DODS 工具生成的进行数据库操作的接口,stu_infoQuery 为查询学生信息表的接口,stu_infoDOI 为对应与学生信息表中一条记录的接口)
        stu_infoQuery query1 = new stu_infoQuery();
        query1.setQueryGradu_id(gradu_id);
        query1.setQueryName(name);
        //System.out.println(“====gradu_id:”+gradu_id);
        if (query1.getNextDO() == null) {
            stu_infoDOI DOI1 = stu_infoDO.createVirgin();
            DOI1.setGradu_id(gradu_id);
            DOI1.setName(name);
            .....
            .....
            //保存导入到数据库中的数据
            DOI1.save();
            cc++;
        }
    }
}

workbook.close();
.....
return cc;
```

```
}

5.3 文本文件的导入
//导入学生信息
public static int importStudent2(String file, HttpPresentationComms
comms) throws
    Exception {
    int cc = 0;
    //创建文件操作对象,准备对文本文件的读取
    String record = null;
    FileReader fr = new FileReader(file);
    BufferedReader br = new BufferedReader(fr);
    record = new String();
    String now = DateFormat.getDateInstance().format(new java.
util.Date());
    Enhydra.getLogChannel().write(Logger.DEBUG,“”开始导
入学生记录:” + new java.util.Date());
    //每读取一行文本文件的内容便将相应的内容导入到
数据库
    while ((record = br.readLine()) != null){
        //对所读取的一行文本文件进行分解,取得所要的数
据(该文本文件导出时,使用“,”对字段进行了分隔)
        String[] a=record.split(“,”);
        String gradu_id=a[35].trim();
        String name=a[2].trim();
        .....
        .....
        //将数据写入数据库(此处用到了使用 Enhydra 的 DODS
工具生成的进行数据库操作的接口,stu_infoQuery 为查询学
生信息表的接口,stu_infoDOI 为对应与学生信息表中一条
记录的接口)
        stu_infoQuery query1 = new stu_infoQuery();
        query1.setQueryGradu_id(gradu_id);
```

```
query1.setQueryName(name);
if (query1.getNextDOI() == null) {
    stu_infoDOI DOI1 = stu_infoDO.createVirgin();
    DOI1.setGradu_id(gradu_id);
    DOI1.setName(name);
    .....
    .....
    DOI1.save();
    cc++;
}
}

br.close();
fr.close();
.....
.....
return cc;
}
```

6 小 结

本文所实现的功能只是对使用 java 技术进行不同数据库系统间的数据交换的一点探索,它在一定程度上改善了非专业计算机人员在进行数据操作时的效率. 经过测试,该程序使用效果良好.

参考文献:

[1] 利用 JAVA 操作 EXCEL 文件[Z] Rubber (tim@trend.com.cn) 2003.1 <http://www-900.ibm.com/developerWorks/cn/java/1-javaExcel/index.shtml>.
[2] <http://www.andykhan.com/jexcelapi/>[Z].
[3] Java Excel API 文档[Z].
[4] JDK1.4.0 文档[Z].

A Study on the Data Exchange between Different Systems with Java

WANG Hai-xia¹, YU Mao-jun²

(1.Center of Modern Education and Technology;2. Office of International Collaboration and Exchange, East China Jiaotong Univ., Nanchang 330013, China)

Abstract: This article describes the method of accomplishing the data exchange between different system with Java on enhydra platform and gives an example.

Key words: enhydra; java; excel; text; database