

基于 B/S 的化检数据管理系统研究

Assaying Data Management System Based on B/S

余志书 张祖平 (中南大学信息科学与工程学院 长沙 410083)

摘要:论文在论述化检数据管理系统需求及技术方案的基础上,重点阐述了系统的设计和实现技术,详细分析了系统中多条记录录入与缓冲、基于过滤器的安全性控制、自动计算等关键技术的实现。系统的设计采用基于 Jsp、Servlet 的 B/S 模式,以 Oracle9i 为数据库服务器,Tomcat 提供应用服务。

关键词:化检数据管理系统 B/S 模式 JSP 权限过滤器

1 前言

化检数据管理在氧化铝的生产过程中起着非常重要的作用,其特点是数据分析量大、涉及的部门多、各部门传递的信息量大。近年来,氧化铝的需求旺盛,而随着社会生产规模的持续扩大,生产过程中的分析数据滞后、准确性不够、信息不畅通等问题日趋严重,导致生产调度人员缺乏准确、及时的生产过程实时数据和生产状况分析数据,最终导致决策失误、调度滞后,浪费了企业大量的人力物力。

本文针对氧化铝企业的生产特点和用户的需求及操作风格,提出了基于 B/S 模式的化检数据管理系统。由于系统数据更新频率快,数据安全性要求高,而这正是 B/S 模式本身难以逾越的缺陷,本文在满足用户需求的前提下,深入分析了 B/S 模式的不足,总结了系统的一些实现难点和关键技术,并提出了有效的解决方案。

2 系统需求分析与设计

2.1 系统需求分析

该系统需要简单方便的用户界面,精确快速的计算模块,强大准确的查询功能,将主要完成分析室的化验分析数据的录入、存储、浏览、查询及分析统计等功能和其它各单位部门的查看、查询、分析统计报表生成、数据导出等功能。

2.2 系统设计

(1) 系统选型。B/S 模式是用户的直接要求,它本身也具有明显的特点。相比 C/S 模式,B/S 模式具

有^[4]:① 操作具有分布性,可以随时随地进行业务处理;② 业务扩展及系统维护简单方便,只需要改变服务器,即可实现所有用户的同步更新;③ 操作简单方便,易于与企业的 OA 网结合,共享性强。但同时也在系统实现中也具有一些难点,如大数据量的录入、系统的及时响应、系统的安全性控制等,这些都是我们在基于 B/S 模式设计系统时需要解决的。

综合考虑用户的需求,我们提出的化检数据管理系统遵循 J2EE 标准^[2],并在三层结构的基础上再一次细分^[5],使每层的功能更加单一化,层与层之间的接口更加清晰。在系统的核心部分采用了 Java 平台的组件技术,遵循 MVC (Model - View - Controller) 设计模式^[1]的特点,通过映射 MVC 三部分到 J2EE 体系结构中相应的组件。系统总体的架构如图 1 所示。

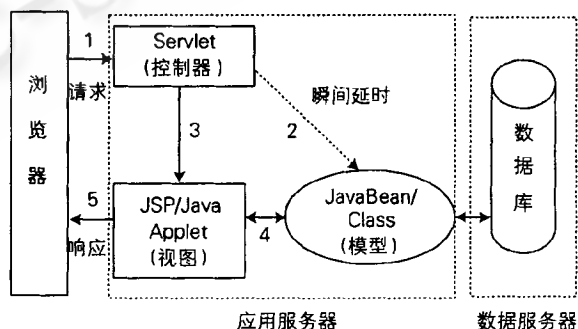


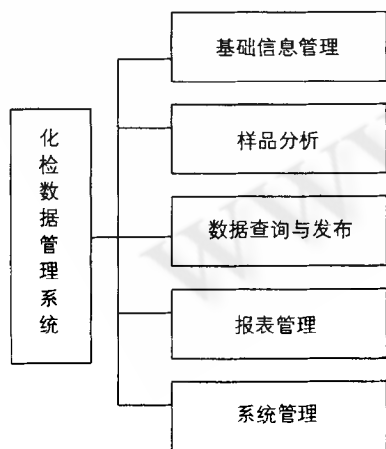
图 1 基于 J2EE 平台和 MVC 模式的系统模型

(2) 数据库设计。在对化检数据管理系统数据库进行逻辑设计时,首先从关系的定义开始进行,通过概

念设计结果的实体联系图进行关系模式的转换,其中包括实体的转换和实体间联系的转换。对转换后的关系模式,进行了规范化处理。

根据本系统的需求分析,最终采用如下的表结构设计思路,将数据表分为两大类:基础信息表、业务数据存储表,其中:基础信息表主要为各类编码数据和基本信息等的存储表格,包括系统设置相关的数据表格。业务数据存储表主要保存手工录入的业务数据及分析结果数据,如分析报告数据等需要较长时间存储的数据。考虑到系统的数据量巨大,而查询频率最高的数据是最新的化检数据,所以在数据库设计时,同时设计了报表数据表和快报数据表,两者表结构相同,只是后者中保存着最新的化检数据,用户直接查询此表,在很大程度上提高了系统的响应性能,有效地解决了系统效率偏低的问题。

(3) 系统主要功能。经过仔细的生产调研和需求分析,系统采用模块化的设计方法。系统主要包括基础信息管理、样品分析、数据查询与发布、报表管理、系统设置五个子系统。模块化的设计使系统可以随企业的应用规模扩展而扩展,各部分既可以相对独立应用,也可组合为一个整体来应用,可以满足企业全面管理的需要。整个系统的开发以业务流程为基础,采用面向对象的技术进行分析、设计,采用组件技术构建软件。各子系统一体化设计,功能流畅、操作方便,界面美观友好。系统功能模块图如图 2 所示。



3 系统关键技术及实现

在系统的实现过程中,存在着许多实现难点和关键技术,下面将对其中一些给出具体的实现技术。

3.1 多条记录录入及缓冲的实现

在样品分析信息录入模块中,实现了多行记录的录入和数据的缓冲,有效地解决了大批量数据录入系统的难题。多行记录录入在 C/S 模式中是很容易实现的,如在用开发工具 VB、PB、Delphi 等开发的系统中。但在 B/S 模式中却存在着很多的困难,因为浏览器是一瘦客户端,一般不具有计算和缓存等功能,而多行记录录入恰恰需要缓存、数据有效性检验且一些指标的自动生成需要大量计算;同时网络的不稳定性也增加了数据的输入和保存难度。

系统主要的数据录入涉及到四个表的插入操作,分别是:

- (1) 样品分析记录主信息
- (2) 样品分析记录明细信息
- (3) 样品分析快报主信息
- (4) 样品分析快报明细信息

其中,1、2 表为一个主明细结构,3、4 表为一个主明细结构,两个结构具有完全相同的定义,只是 3、4 表只保留最新结果。这四个表的插入操作我们设计成一个事务操作,然而在编程实现中,我们不可能做到把这四个表的输入操作写在一个页面里。因为四个表中每一个表的字段都很多,而且还涉及到对应样品的分析指标信息的提取和计算指标的自动计算等问题,所以在输入时,要分页面来实现输入操作,这样就需要把前面的记录保存下来以便在最后一个页面中来完成数据库的插入操作,所以需要使用缓冲技术。在样品分析记录明细信息输入页面中,由于对应一个样品有多个基本分析指标,并且有不同时段化检结果,所以这里有多行记录的录入操作存在。下面将对整个录入的实现思路进行阐述。

系统使用了 Java 的面向对象特性,对应样品分析记录主信息和样品分析记录明细信息这两个表中的记录,分别定义了相应的 Java 类^[3],分别为:recordmaster.java 和 recorddetail.java。每个类的定义很简单,就是定义了相应于每个表中字段的数据元素,这样就把输入界面中的 HTML 表单的数据对应到 Java 中的类了。接着就是许多基本指标记录的缓冲保存,相应的

定义了一个 `JavaBean (recordbuffer.java)` 类来作为缓冲结构,在其中使用了一个向量数据结构。因为主记录与明细记录是一对多的关系,所以不需要定义向量结构。在输入完主记录表单后,提交到一个 `jsp` 页面。在其中把数据定义到 `recordmaster` 对象中,并把它添加进缓冲 `recordbuffer` 对象中。对于明细记录的相关操作,唯一不同的就是他们是多条记录,所以要添加进向量结构中。多条记录输入的实现方法是:在表单中,同一列的输入框取相同的名字,这样就取得了所有输入的基本指标分析结果值,并用循环把它们(定义成 `recorddetail` 对象)存入缓冲中,最后在数据库操作的 `jsp` 页面中只要取出缓冲中的数据就可以了。

3.2 基于过滤器的操作权限控制

系统在设计时把用户分为不同的角色^[6],而不同角色具有不同的模块访问权限(即具有不同的操作页面),比如,非化检业务人员就不能访问化检数据录入部分。而在 `B/S` 模式的系统实现中,用户的操作权限和访问控制一直是一个比较难以解决的问题。在本系统实现中,必须要进行很好的权限访问控制,这样才能确保系统的可用性、正确性、完整性和可审记性。为此,系统采用了 `JSP` 中的过滤器机制来实现此功能,用它作为系统的统一入口,控制用户的访问操作,从而很好的阻止了用户对非授权页面的非法访问问题。

在进行系统的规划和设计时,我们主张把逻辑上的一个功能模块映射到物理磁盘中的一个文件夹,即一个文件夹就是一个完整的功能模块,这也是本系统中用过滤器来实现权限控制的基本方法,因为在角色与模块菜单的对应关系中,只能把一个模块的首页地址与角色进行对应(即保存在数据库中),用首页地址来代表本模块,最后在过滤器中来实现比较。

其具体实现方法是:在用户登录验证页面中,通过验证后把他所具有的所有操作模块的首页地址放入一向量中,并把它作为一属性放入 `session` 中。在用户访问本系统中的其它模块时,都需要经过统一入口——过滤器的检验。若用户进行非法访问时,如在 `IE` 地址栏中直接敲入不具有访问权限的页面地址,过滤器就可以发现(通过与 `session` 中该用户具有的访问地址进行比较),并及时给予阻拦与警告。此过滤器是系统所有业务模块的统一入口,其在 `web.xml` 中需作如下设置:

```
<filter>
```

```
<filter-name> glofilter </filter-name>
```

```
<filter-class> enterfilter.myfilter
```

```
</filter-class>
```

```
</filter>
```

```
<filter-mapping>
```

```
<filter-name> glofilter </filter-name>
```

```
<url-pattern> *.jsp </url-pattern>
```

```
</filter-mapping>
```

3.3 自动计算功能的实现

在系统的业务需求中,一些计算指标值必须由基本指标值进行自动计算求得,其求解公式由在指标项目字典中的计算公式中。系统在样品信息录入及分析模块中的样品分析报告生成中形成计算指标时就运用了自动计算。由于 `JAVA` 不具有对字符串表达式进行解析和计算的类和方法,所以要使用其它办法,在本系统的实现中,采用了一个 `JAVA` 开源项目—`JEP`(即 `Java Mathematical Expression Parser`)。

`JEP` 是一个解析和计算数学表达式字符串的 `JAVA` `API` 库,通过这个 `API` 类库,我们可以让用户输入任意的公式字符串,接着便可以很快地进行计算。`JEP` 支持用户定义的常量、变量和函数,一些数学常量和函数也被包括在了 `API` 中。使用方法:在官方网站下载其压缩包版本(2.24),解压并将其中的 `jep-2.24.jar` 归档文件拷贝到 `WEB` 应用程序的 `lib` 目录中,现在就可以使用了。

本系统中我们采用如下方法:

(1) 首先定义一个解析器对象:

```
<%
```

```
org.nfunk.jep.JEP myParser = new org.nfunk.
```

```
jep.JEP();
```

```
%>
```

(2) 由于有多个基本指标,并且每个计算指标公式中的变量(基本指标名)不同,这样给计算增加了一定的难度。在这里的方法是,用两个字符串分别用来记录基本指标名和其输入的值(从缓冲中获得)。代码如下:

```
<%
```

```
zbstr = null; //对应指标名字符串(以逗号作为分隔符)
```

```

zbvalstr = null; //对应指标值字符串 (以逗号作为隔
符)
for (int i = 0; i < data.length; i++) {
    //data 是从缓冲中取得的明细记录数组
    if (data[i].itemresulttype.equals("0")) //是基本指
    标
    {
        zbstr = zbstr + ",".concat(data[i].itemname);
        zbvalstr = zbvalstr + ",".concat(data[i].itemresult);
    }
}
% >

```

需要说明的是,上面的 0 是在样品分析明细记录中的一个标志字段,0 表示基本指标,1 表示计算指标。

(3) 对于每一个计算指标公式循环遍历上面的数组,并用其值取代其指标名,最后计算。部分代码如下:

```

<% //把上面的两个辅助字符串转化成数组
String[] zbararray = zbstr.split(",");
//基本指标名数组
String[] zbvalarray = zbvalstr.split(",");
//基本指标值数组
String formula = null;
for (int j = 0; j < data.length; j++) {
    if (data[j].itemresulttype.equals("1"))
    //是计算指标
    {
        formula = data[j].itemresult;
        //获取计算公式
        for (int k = 0; k < zbararray.length; k++) {
            if (formula.indexOf(zbararray[k]) >= 0)
                formula = formula.replaceAll(zbararray[k], zbvalarray[k]);
        } //以上代码完成循环遍历指标名数组,
        //并用相应指标值取代指标名变量
        myParser.parseExpression(formula);
        //解析表达式
        formula = String.valueOf(Math.round(myParser.
        getValue() * 100.00) / 100.00)
    }
}

```

```

}
% >

```

至此,就完成了计算指标的自动计算功能。

4 结束语

B/S 模式由于具有操作方便、综合性好、易于维护等特点,广泛为用户和设计者所推崇,然而对于数据更新频率快、安全性要求高、访问量大的系统,其缺点也是十分明显的,为系统的设计与开发增加了难度。

论文根据流程制造行业自身的特点,按照 J2EE 的体系结构和 MVC 的设计思想,提出了一个以组件为核心,紧密结合业务流程的化检数据管理系统的框架模型,并详细介绍了其中的多条记录录入与缓冲、基于过滤器的操作权限控制和自动计算等关键技术的设计与实现,有效地克服了 B/S 模式本身的缺陷,又很大程度地满足了用户要求。这些关键技术的实现具有一定的代表性,可以为基于 B/S 模式的企业信息化系统建设提供较好的参考作用。

系统在某氧化铝生产企业一年多的成功应用表明,系统加快了企业化检数据的处理速度,满足了用户的业务需求,很大程度上提高了企业的生产效率和管理水平,为调度决策管理的准确性和企业的现代化管理奠定了良好的基础。

参考文献

- 1 陆荣幸、郁洲、阮永良、王志强, J2EE 平台上 MVC 设计模式的研究与实现[J], 计算机应用研究, 2003, 3: 144-146.
- 2 胡海燕、贺贵明, J2EE 的安全机制及其应用研究[J], 计算机应用, 2003, 23(12): 153-154.
- 3 潘利群, JavaBean 在 JSP 中的应用研究[J], 武汉理工大学学报, 2003. 025(005), 65-67.
- 4 唐卫宁、李艳、耿国华, 基于 C/S 与 B/S 混合结构的科技项目申报管理系统的设计与实现[J], 计算机应用与软件, 2004, 21(9): 46-47.
- 5 Chen S, et al. Web-based Simulations of Power Systems[J], IEEE Computer Applications in Power, 2002, 15(1): 35-40.
- 6 罗雪平、郑奕莉、徐国定, 一种扩展的基于角色的访问控制模型, 计算机工程, 2001, 27(6): 106-107.