

DOI :10.3880/j.issn.1006-7647.2008.06.016

基于 Hibernate 的辽河流域防洪调度系统

梁国华^{1,2}, 李娜¹, 彭勇¹, 吕涑琦¹

(1. 大连理工大学土木水利学院 辽宁 大连 116024; 2. 河海大学水文水资源与水利工程科学国家重点实验室 江苏 南京 210098)

摘要 :首先 ,分析目前 3 层结构防洪调度系统的优缺点及其应用存在的问题 ;其次 ,采用面向对象持久化技术和 Java 技术 ,进行基于 Hibernate 的 4 层结构防洪调度系统研究、设计与开发 ;最后 ,将研究成果应用到辽河流域防洪调度系统中。应用结果表明 ,对于不同的数据库管理系统 ,只要修改数据库的配置文件 ,不需要修改任何程序 ,系统也能满足应用要求 ,从而大幅度减少了系统的重复开发量 ,可以为其他水利信息化系统提供借鉴。

关键词 :防洪系统 ;对象持久化 ;Hibernate ;辽河流域

中图分类号 :TV87 文献标识码 :A 文章编号 :1006-7647(2008)06-0062-04

Flood control system for Liaohe Basin based on the Hibernate framework//LIANG Guo-hua^{1,2}, LI Na¹, PENG Yong¹, LV Su-qi¹(1. Department of Civil Engineering, Dalian University of Technology, Dalian 116024, China 2. State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Hohai University, Nanjing 210098, China)

Abstract :In consideration of the disadvantages of the three-tier flood control system structure, a four-tier structure based on the Hibernate framework was designed and developed by use of object persistence and Java techniques. Application in Liaohe Basin shows that this four-tier flood control system structure can satisfy the requirements of different database management systems as well as amending the configuration files, without requiring that the procedure be modified. Hence, it can greatly reduce the duplicate development of the flood control system. Research in this paper provides a basis for the design of other water information systems.

Key words :flood control system; object persistence; Hibernate; Liaohe Basin

随着信息技术的发展 ,Internet 在水利信息化中得到了广泛应用^[1-4]。现存的绝大部分水利信息化应用系统 ,尤其是防洪调度系统应用软件的体系结构几乎都是采用典型的 3 层应用系统^[2,4] ,即表述层、业务逻辑层和数据库层 ,其层次结构如图 1 所示。其中 ,表述层提供与用户的交互界面 ,GUI 和 Web 页面是表述层的典型应用 ;业务逻辑层实现各种水利计算的逻辑 ,它不仅负责业务逻辑的处理 ,而且直接访问数据库 ,提供对业务数据的保存、更新、删除和查询操作 ;数据库层负责存放和管理数据。

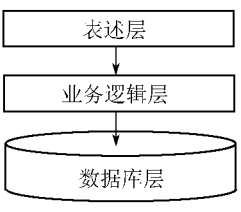


图 1 3 层应用系统

在 Java 技术应用领域中 ,业务逻辑层可以直接

通过 JDBC 编程访问数据库。JDBC 可以说是访问关系数据库的最原始、最直接的方法。这种方式的优点就是运行效率高 ,但存在如下缺点 :①实现业务逻辑的代码和数据库访问代码掺杂在一起 ,使程序结构不清晰 ,可读性差 ;②在程序代码中嵌入面向关系的 SQL 语句 ,使开发人员不能完全运用面向对象的思维来编写程序代码 ;③业务逻辑与关系数据模型绑定 ,如果关系数据发生变化则必须手工修改代码中所有相关的 SQL 语句 ,这样就增加了系统维护的难度 ;④若程序代码中 SQL 语句包含语法错误 ,在编译时不能检查出这种错误 ,只有在运行时才能发现这种错误 ,因此系统开发人员很难判断引起错误发生的位置 ,从而增加了调试程序的难度。

根据用户的实际需求 ,目前的防洪调度系统中往往是多种数据库管理系统并存且不断变化^[3]。这给系统开发和维护带来了极大的困难 ,也给系统的安全运行带来了隐患。因此 ,必须研究采用新技术、

新途径去适应数据库层的不断变化,当数据库管理系统发生变化而且数据库表结构保持不变或者变化不大时,不需要修改系统的业务逻辑层和表述层也能确保已经调试好的系统正常运行。基于这种思想,本文采用 Java 技术方案,实现基于 Hibernate 的辽河流域防洪调度系统设计与开发。

1 辽河流域防洪调度系统结构体系设计

在 3 层结构的防洪调度系统中,业务逻辑层和数据库层中间加入了持久化层,变成了 4 层体系结构的应用系统,如图 2 所示。当用户发出或提交请求时,表述层下载到浏览器端与用户进行交互,通过事件响应,把用户指令通过网络提交给 Web 服务器,再由业务逻辑层对指令和数据进行处理,最后将处理结果返回到浏览器端显示给用户,这样多次往返地提交响应来完成整个调度作业。系统中表述层将防洪信息以最符合用户习惯的图表方式直观地表达给决策者,业务逻辑层根据用户的不同请求,实现数据提取、信息查询、前期影响雨量计算与修改、防洪形势分析、洪水预报、洪水调度、方案管理、防汛会商和数据库维护与管理等功能,持久化层就是为了把数据访问细节和业务逻辑分开,把访问数据作为单独的持久化层,它包括与数据库相关的各种操作,如保存、更新、删除、加载、查询等。持久化层封装了数据访问细节,为业务逻辑层提供了面向对象的 API。该系统采用 Java 技术开发方案,由此可以认为持久化层就是 Java 技术应用程序和关系数据库进行数据处理与交换的场所。

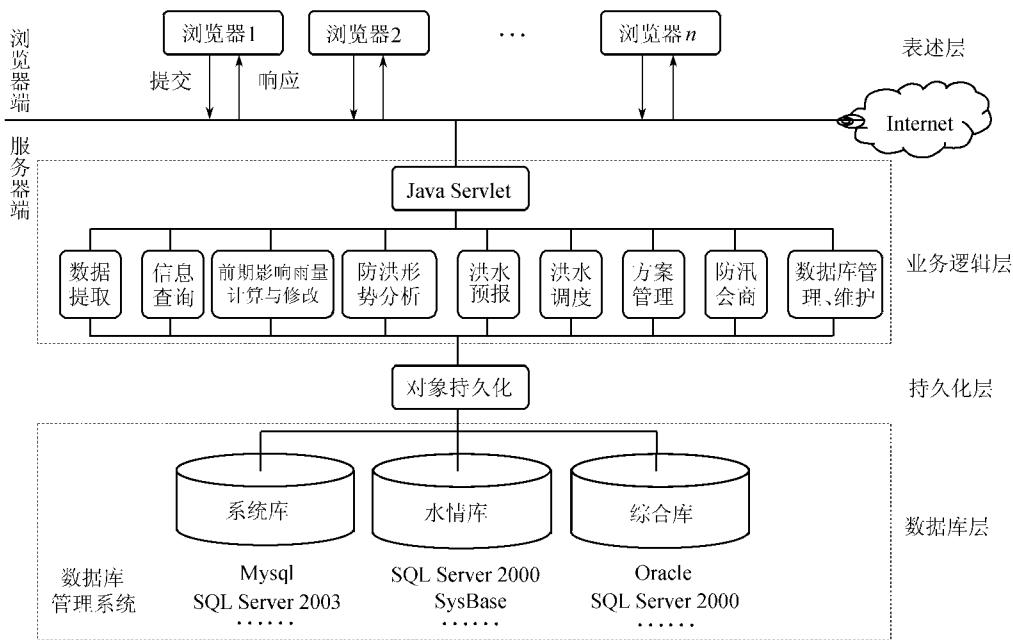


图 2 基于 Hibernate 的防洪调度系统逻辑结构

2 系统持久化开发过程

2.1 持久化技术方案选择

目前,Hibernate 是对象持久化技术应用最为广泛的技术方案^[5-6]。它是一个开放源代码 ORM (object/relational mapping) 影射映射框架,其对 JDBC 进行了非常轻量级的对象封装,使得 Java 技术程序员可以随心所欲地使用对象编程思维来操纵数据库,从而大幅度减少了操纵数据库的工作量。Hibernate 可以应用在任何使用 JDBC 的场合,既可以在 Java 技术的客户端程序使用,也可以在 Servlet/JSP 的 Web 应用中使用,最重要的是 Hibernate 可以整合到应用 EJB 的 J2EE 系统作为持久化层架构中。Hibernate 通过设置配置文件 hibernate.properties 和创建 XML 映射,提供持久对象给应用程序,可以代替大量复杂的 JDBC 编程。目前 Hibernate 支持 17 种数据库,同时也支持连接池应用。

Hibernate 具有如下特点:① Hibernate 是 JDBC 轻量级的对象封装,它是一个独立的对象持层框架。Hibernate 自带功能强大的查询语言(HQL),HQL 与 SQL 非常相似,并且它是完全面向对象的,使用简单,功能灵活且强大,运行方便快捷,可以代替 JDBC 的编程应用场合。② Hibernate 具有自己的事务管理,但是,Hibernate 事务管理实际是 JDBC Transaction 的封装,或者是 JTA Transaction 的封装,因为 Hibernate 是对 JDBC 的轻量级对象封装。由此,Hibernate 事务管理可以在 JDBC Transaction 和 JTA Transaction 之间进行选择。③ Hibernate 是一个和 JDBC 密切相关的框架,Hibernate 的兼容性只与

JDBC 驱动和数据库有一定的关系 ,与 Java 技术程序或者 App Server 没有任何关系 ,不存在兼容性问题。

下面以控制站洪水过程表 RiverFloodsR 的配置为例 ,介绍基于 Hibernate 应用系统的开发配置过程。

2.2 配置 Hibernate

在对 Hibernate 的配置中 ,主要涉及 XML 文档的类型定义、数据库连接的各种参数以及提供 Hibernate 管理事务、产生 SQL 和 JDBC 连接所需的数据库影射文件等。配置文件 hibernate.cfg.xml 代码如下 :

```
< ? xml version = "1.0" encoding = "utf - 8" ? >
< ! DOCTYPE hibernate - configuration
PUBLIC " - //Hibernate/Hibernate Configuration DTD 3.0//
EN"
"http ://hibernate. sourceforge. net/hibernate - configuration -
3.0. dtd" >
< hibernate - configuration >
< session - factory >
< ! —— - 数据库 URL —— >
< property name = " connection. url" >
jdbc :oracle :thin :@ localhost :1521 :lmsys//Oracle 连接 ,
不同数据库需要修改此处
< /property >
< ! —— - 数据库 JDBC 驱动 —— >
< property name = " connection. driver_ class" >
oracle. jdbc. driver. OracleDriver//Oracle 驱动 ,不同数据库
需要修改此处
< /property >
< property name = " connection. username" > User < /property >
< property name = " connection. password" > Mypass < /
property >
< ! —— - dialect ,每个数据库都有其对应的 Dialect 以匹
配其平台特性 —— >
< property name = "dialect" >
org. hibernate. dialect. OracleDialect//Oracle ,不同数据库需
要修改此处
< /property >
< mapping resource = "hibernate/RiverFloodsR. hbm. xml" / >
< /session - factory >
< /hibernate - configuration >
```

在该配置文件中 ,根据不同的数据库管理系统对相应的连接参数进行修改 ,就可以实现不同数据库的连接。

2.3 创建映射文档

在上述 hibernate.cfg.xml 文件中 ,mapping 映射文件配置部分 ,定义了表 RiverFloodsR 对应的映射文件 RiverFloodsR.hbm.xml ,它用到的映射资源可以随着系统开发进行灵活的加载与更新。数据库表所对应的映射文件包含对象/映射关系所需的元数据。元数据中包含了持久化类的声明 ,以及类中各个属性到数据库表各个字段的映射关系。以下是映射文件的部分内容 :

```
< ? xml version = "1.0" ? >
< ! DOCTYPE hibernate - mapping PUBLIC
" - //Hibernate/Hibernate Mapping DTD 3.0//EN"
"http ://hibernate. sourceforge. net/hibernate - mapping -
3.0. dtd" >
< hibernate - mapping >
< class name = " hibernate. RiverFloodsR" table = "
RiverFloodsR" >
< composite - id name = " comp_ id" class = " hibernate.
RiverFloodsRPK" >
< key - property name = "idcomp" column = "idcomp" type = "
java. lang. String" length = "25" >
< /key - property >
< key - property name = "dt" column = "dt" type = "java. sql.
Timestamp" length = "23" >
< /key - property >
< /composite - id >
< property name = "z" type = "java. math. BigDecimal" column
= "z" not - null = "true" length = "7" >
< /property >
< property name = " flow" type = " java. math. BigDecimal"
column = "flow" not - null = "true" length = "9" >
< /property >
< /class >
< /hibernate - mapping >
```

在映射文件中 ,业务逻辑层并不知道当前所连接的数据库类型 ,这样就使得系统与数据库管理平台无关。

2.4 持久化类定义

持久化类是指利用 Hibernate 将实例持久化到数据库中的类。持久化类符合 Java Bean 规范 ,它包含有符合统一标准的属性 ,以及与之相对应的 getXXX() 和 setXXX() 方法。在持久化类中有一个 ID 主键属性 ,用来标识持久化类的每个对象。由于持久化类将映射到数据库中对应的表 ,所以主键属性的设置在类中是必不可少的。RiverFloodsR 表主键是复合主键 ,因此 ,在持久化时必须先定义持久化类中的复合主键。以下是定义 RiverFloodsR 持久化类的部分代码 :

```
public class RiverFloodsRPK implements Serializable { //持久
化类中的复合主键
private String idcomp ;
private Date dt ;
public RiverFloodsRPK( String idcomp , Date dt ) {this. idcomp
= idcomp ; this. dt = dt ;}
.....
}
public class RiverFloodsR implements Serializable { //持久化类
private hibernate. RiverFloodsRPK ID ;
private BigDecimal z ;
private BigDecimal flow ;
public RiverFloodsR( ) {}
public hibernate. RiverFloodsRPK getID( ) {return this. ID ;}
public void setComp_ id( hibernate. RiverFloodsRPK ID )(this.
```

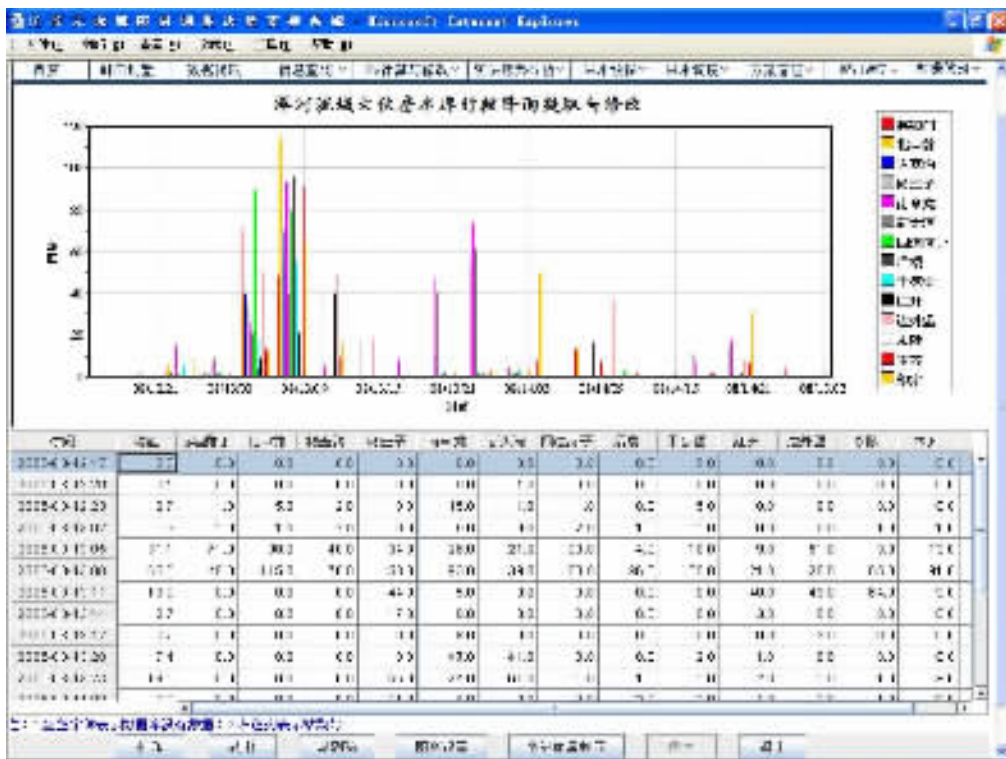


图 3 基于 Hibernate 的雨量提取与维护界面

```
ID = ID ;}  
.....  
}
```

上述创建映射文档和定义持久化类可以通过专门的软件自动完成,在实际应用中不需要人工进行对象持久化操作。因此,不需要增加更多的工作量。

3 系统应用实例

辽宁境内的辽河、浑河、太子河是辽河流域的三大主要支流。清河、柴河、南城子、大伙房、观音阁、侵窝、汤河等大型水库与河道以及 1 500 多 km 堤防组成的防洪工程体系保护着辽宁中部地区城市群及下游地区人民群众的生命和财产安全。

目前辽宁省直属水库已经建立了完善的水雨情自动测报系统,并且各自成系统。每个水库管理局已经与省水利厅构成企业内部网,当进行防洪调度决策时,则需要从水库获取雨情、水情和工情等信息。由于建设时期不同,每个水库数据库系统版本及数据库管理系统软件不尽相同。如果不采用 Hibernate 技术,则需要编写操纵数据库的不同 SQL 代码,这样给省级系统的集成和安全运行带来了相当大的困难。为此,该系统采用面向对象持久化技术,开发了辽河流域防洪调度系统。系统 Web 应用服务器采用 Tomcat 6.0 ;Java 技术运行环境为 Jdk 5.0 ,根据不同的用户数据源,数据库方面分别为 Mysql ,SQL Server 2003 ,SQL Server 2000 ,Oracle9i 等。图 3 为该系

统采用 Hibernate 开发的一个雨情数据

4 小 结

笔者分析了目前 3 层结构防洪调度系统的特点及其应用存在的优缺点,尝试应用面向对象持久化技术和 Java 技术进行基于 Hibernate 的防洪调度系统的研究、设计与开发。应用实例表明,不同的数据库管理系统,只要修改数据库的配置文件,应用系统都能正常运行。对于不同的数据库,不需要修改任何程序,系统也能满足要求,从而大幅度减少了系统的重复开发量。因此,笔者认为该研究成果具有实用价值,对其他类似防洪调度系统的建设与开发具有借鉴意义。

参考文献:

- [1] 丁涛,翟宜峰,周惠成.基于 B/S 结构的信息查询通用数据模型[J].辽宁工程技术大学学报,2003,22(5):646-648.
- [2] 周惠成,彭勇,梁国华.基于 B/S 模式的水库防洪调度系统的设计与开发研究[J].计算机应用研究,2005(6):150-151,186.
- [3] 王国利,林剑艺,周惠成.碧流河水库防洪决策支持系统集成研究[J].大连理工大学学报,2004,44(5):719-722.
- [4] 李根,姜卉芳,穆振侠.基于 WEB 的水库洪水调度系统[J].新疆农业大学学报,2006,29(1):79-82.
- [5] 高昂,卫文学.基于 Hibernate 与 Struts 框架的数据持久化应用研究[J].计算机应用,2005(12):101-104.
- [6] 宋汉增,沈琳.利用 Hibernate 对象持久化服务简化 Java 数据库访问[J].计算机应用,2003(12):135-137.

(收稿日期 2007-11-06 编辑 高建群)