

# 基于 JavaBean 和 Hibernate 数据库应用的构建

迟金成 刘万军<sup>1</sup>(辽宁工程技术大学电子信息工程学院, 软件学院<sup>1</sup>, 葫芦岛 125105)

**摘要** JavaBean 与 Hibernate 的结合使用有助于快速开发软件系统。进行对象-关系的映射(Object/Relation Mapping, 简称 ORM), 可把对象持久化到关系数据库中。文中结合葫芦岛渤海石化起重机械有限责任公司管理信息系统对 JavaBean 和 Hibernate 做了较详细的阐述。

**关键词** JavaBean Hibernate 对象 对象/关系映射(ORM)

**中图法分类号** TP311.132.4; **文献标识码** A

在计算机信息技术飞速发展的今天, 面向对象的开发已成为主流。关系数据库仍然是应用最为广泛的数据库存储, 但关系数据库中存放的是关系数据, 是非面向对象的, 这就需把对象持久化到关系数据库中。因此, 需要进行对象-关系的映射(Object/Relation Mapping, 简称 ORM)。在对象上用 JavaBean 构建, 数据持久层选择 Hibernate 架构, 二者的结合使用可有效地从关系数据操作中解脱出来, 化烦杂为简单。因此, 在项目开发中不管是从效率上, 还是易维护上都是很好的结合。

## 1 JavaBean

JavaBean 实际是一个 Java 类。可以使用 JavaBean 将功能、处理、值和其他任何可以用 Java 代码创造的对象打包成对象。构建一个 JavaBean 必须包含以下几个要素:

- 1) 必须实现 Java.io.Serializable 接口;
- 2) 提供 getter 和 setter 方法访问它的属性;
- 3) 提供无参数的构造器, 此构造器也应该通过调用各个特性的设置方法来设置特性的缺省值。

## 2 Hibernate 框架简介

提出对象/关系映射技术(Object - Relational Mapping, O/R Mapping)的目的主要是为了解决在面向对象的开发平台下(如 Java), 如何使用一种面向对象的方式来操纵和管理关系型数据的问题。利用 O/R Mapping 技术可以实现一个相对独立的对象持久化层(Object Persistence Layer), 利用这个对象持久化层, 开发人员就可以将所要管理的关系型数据封装成标准的对象(或类), 进而使用面向对象的方法来操作这些数据。

Hibernate 框架是一个优秀的、开放源代码的、面向 Java 环境的对象/关系数据库映射工具, 它对 JDBC 进行了轻量级的对象封装, 使得 Java 程序员可以使用面向对象编程思维来操作关系数据库如图 1。它可以让普通的 Java 对象(Plain Old Java Objects, POJOs)变成持久化类(Persistence Object)。一般情况<sup>[1]</sup>下, 一个持久化类和一个表对应, 类的每个实例对应表中的一条记录。

Hibernate 提供了对数据库的种种操作, 如数据查询、删除和获取数据等方法, 因它是基于 SQL 的, 并把复杂的 SQL 语句封装起来, 可以大幅度减少开发时人工使用 SQL 和 JDBC 处理数据的时间和复杂性。

2006 年 11 月 20 日收到

第一作者简介: 迟金成(1977—), 男, 汉, 吉林省辽源市人, 硕士研究生, 研究方向: 网络理论及应用。Email: c7439@21cn.com。

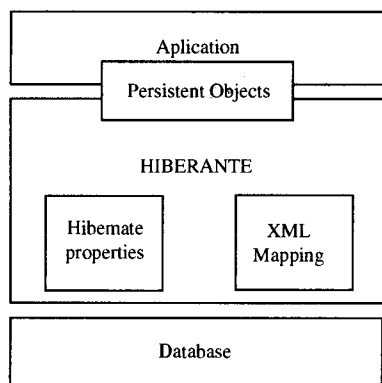


图 1 Hibernate 框架图

### 3 基于 Hibernate 实例

Hibernate 本质上是一个提供数据库服务的中间件,它利用数据库和其他一些配置文件来为应用程序提供数据持久服务。

应用 Hibernate 所做的工作非常简单。首先需要编写 O/R 映射描述文件,完成对象、关系数据库之间的映射,持久对象可以根据映射文件生成;然后编写业务逻辑类,下面以一派车卡为例。

表 1 派车卡

司机		车号	
姓名			
托运单位	货物名称	装货地点	卸货地点
1			
2			

#### 3.1 创建持久化类 VO

例子中第一行为主表 main 和内容表为 content,是一个 One - To - Many 的关系。Hibernate 要持久化的对象可以使用标准 JavaBean,对每个属性值使用有 public 的 getter 和 setter 方法,命名也要符合 bean 方法的命名规则。这里仅给出各个类的属性和类型:

Main:

```

private Long id;
private String driverName ;
private String carNum;
private Date allotDate;
private Set conts = new HashSet();

```

Content:

```

private Long id;
private Long allotId;
private String consignDept ;
private String goodsName ;
private String loadingPlace;
private String unloadingPlace;
private String factualTime ;
private String allTime ;
private SDAllotCarVO sdvo

```

#### 3.2 Hibernate 配置文件

Hibernate 配置文件主要用于配置数据库连接和 Hibernate 运行时所需的各种属性。

以 Oracle 数据库连接为例,有如下相关基本配置属性:

```

<hibernate - configuration >
<session - factory >
<! -- 指定数据库的驱动程序 -- >
<property name = "connection.driver_class" >
oracle.jdbc.driver.OracleDriver
</property >
<! -- 指定连接数据库 URL -- >
<property name = "connection.url" >
jdbc:oracle:thin:@localhost:1521:ora9i
</property >
<! -- 指定连接数据库的用户名 -- >
<property name = "connection.username" >
name
</property >
<! -- 指定连接数据库的口令 -- >
<property name = "connection.password" >
password
</property >
<! -- 指定数据库使用的 SQL 方言 -- >
<property name = "dialect" >
org.hibernate.dialect.Oracle9Dialect
</property >
<! -- 控制在控制台是否输出 SQL 语句,利于开发和调试 -- >
<property name = "hibernate.show_sql" >
true
</property >

```

```
<!-- 关于持久类信息 -->
<mapping resource = "映射文件位置" />
</session - factory>
</hibernate - configuration>
```

以上为 Hibernate 配置的相关基本参数配置,可以根据具体的需要来进行其他属性参数的配置,以实现相应的功能。经葫芦岛渤海石化起重机械有限责任公司管理信息系统的开发过程, Hibernate 到达配置灵活、维护简单。使用方便等优势。

### 3.3 创建对象-关系映射文件 mapping(持久层的配置)

建立 Hibernate 的持久层,首先要将域对象持久化。Hibernate 一般采用易用易维护的 XML 文件来映射(O/R)对象。在这里需要注意的就是各种关系的映射,如 one to one、many to one、collection 的 many to many、one to many、子类映射等。例:关于主表的映射配置:

```
<hibernate - mapping>
<class name = "MainVO" table = "mainTable" dynamic - update =
"true" dynamic - insert = "true">
    <id name = "id" column = "id">
        <generator class = "sequence">
            <param name = "sequence">mainTable_id</param>
        </generator>
    </id>
    <property name = "driverName"/>
    <property name = "carNum"/>
    <property name = "allotDate"/>
    <set name = "conts" table = "ContTable" cascade = "all - delete -
orphan" inverse = "false" order - by = "id">
        <key column = "allotId"/>
        <one - to - many class = "ContVO"/>
    </set>
</class>
</hibernate - mapping>
内容表映射配置:
<hibernate - mapping>
<class name = "ContVO" table = "contTable" dynamic - update =
"true" dynamic - insert = "true">
    <id name = "id" column = "id">
```

```
        <generator class = "sequence">
            <param name = "sequence">contTable_id</param>
        </generator>
    </id>
    <property name = "goodsName"/>
    <property name = "loadingPlace"/>
    <property name = "unloadingPlace"/>
    <property name = "factualTime"/>
    <property name = "allTime"/>
    <many - to - one name = "sdvo" class = "mainVO" cascade =
"none" update = "true" insert = "true" access = "property" column =
"allotId"/>
</class>
</hibernate - mapping>
```

## 4 结论

Hibernate 是一种先进的 O/R 映射工具, JavaBean 实现对象的创建。在项目开发实践中, JavaBean 和 Hibernate 框架的结合完成了对象的创建与对象和关系之间的映射,可以使得开发变得更容易,充分发挥了 JavaBean 和 Hibernate 的优点,基于此框架的开发简洁、结构清晰,有很好的可扩展性和可维护性,非常适于面向对象的设计与开发,从而提高了软件开发效率,增强系统的可维护性和可移植性。

随着人们对此技术研究的深入,一定会使此框架得到进一步的充实和扩充。

## 参 考 文 献

- 1 孙卫琴. 精通 Hibernate: Java 对象持久化详解. 北京: 电子工业出版社, 2005
- 2 夏 昕, 曹晓钢, 唐 勇. 深入浅出 Hibernate. 北京: 电子工业出版社, 2005
- 3 hibernate\_reference.pdf 文档
- 4 <http://www.hibernate.org/>
- 5 <http://www.redsaga.com/mambo/> 满江红

## Based on JavaBean and Hibernate Database Application Constructing

CHI Jin-cheng, LIU Wan-jun<sup>1</sup>

(School of Electronic and Information Engineering,

School of Software<sup>1</sup>, Liaoning Technical University, Huludao 125105, P. R. China)

[**Abstract**] JavaBean and the Hibernate union use is helpful to the fast development software system. Object persistence is put to relational databases, for object - relational mapping( Object/Relation Mapping, called ORM). JavaBean and Hibernate are introduce and have made a more detailed elaboration to Hibernate.

[**Key words**] JavaBean      Hibernate      object      object/relation mapping( ORM)

(上接第 1474 页)

## Forward Security Proxy Signature Scheme of Original Signer's Key Evolving

XIA Feng, WU Li-hua

(Department of Computer Science and Education Technology, Hainan Normal University, Haikou 571158, R. R. China)

[**Abstract**] Forward security proxy signature scheme established mostly supposes that the original signer's secret key is secure, and the proxy signer's secret key evolves. If original signer's secret key is exposed, the proxy signature before is not validity. A new forward security proxy signature scheme is proposed. In the scheme, valid of the original signer's secret key is compartmentalized to some phases, and current secret key generates proxy signature key which is valid only at current phase. If original signer's secret key is exposed, attacker can't get the secret key in the past and can't forge proxy signature key. It will promise that the proxy signature in the past is valid.

[**Key words**]      proxy signature      forward security      Guillou - Quisquater signature