

基于 Hibernate 框架的数据持久层架构设计及应用*

刘德山, 杨彬彬

(辽宁师范大学 计算机与信息技术学院, 辽宁 大连 116029)

摘要: 数据持久层在 Web 应用系统开发中, 主要应用于业务逻辑和数据逻辑的松散耦合, 提高软件可重用性。分析了 Hibernate 框架及 DAO 设计模式的工作原理, 设计了基于 Hibernate 框架采用 DAO 设计模式的数据持久层架构体系, 并通过具体 Web 应用, 阐述了数据持久层架构的具体实现过程。

关键词: 数据持久化; Hibernate 框架; DAO 模式; O/R 映射

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2011)15-0012-03

Design and application of data persistence layer schema based on hibernate framework

Liu Deshan, Yang Binbin

(School of Computer and Information Technology, Liaoning Normal University, Dalian 116029, China)

Abstract: Data persistence is used in business logic and data logic to improve software reusability in the Web application development. Introduced the signification of the data persistence, analyses work principle of Hibernate framework and the DAO design pattern, designed the data persistence layer schema based on Hibernate framework which used DAO design pattern, Finally, the implementation process of the model in a Web project described detailly.

Key words: data persistence; Hibernate framework; DAO design pattern; O/R mapping

在 Web 应用系统中, Web 服务器作为连接客户端和数据库服务器的一个中间层, 既要客户端提出的请求进行业务分析和处理, 又要访问数据库并与数据库进行交互。当业务逻辑代码中包含数据访问代码时, 将给系统的修改和维护带来很大困难, 同时也不利于以后在开发类似系统中进行大粒度的软件重用。因此有必要把功能层划分为业务逻辑层和持久层。业务逻辑层专注于业务逻辑的分析和处理, 持久层专注于与数据库交互进行数据的持久化工作, 并提供一个标准的接口供业务逻辑层调用。通过这样一个层次的划分, 实现了高效、清晰的专业分工和职责划分, 既实现了功能层中业务逻辑和数据逻辑之间的松耦合关系, 又利于系统的修改、维护和软件重用^[1]。

目前的 J2EE 应用系统开发中已经存在不少框架。在系统架构分析时, 如何应用已有框架为应用系统量身定做做一个合适的架构, 对 J2EE 设计开发者提出了挑战。本文基于 Hibernate 框架采用 DAO 设计模式, 设计了数

据持久层架构。并将其应用于问卷调查系统, 实现了业务逻辑和数据逻辑的松散耦合, 提高了系统的开发效率和软件重用性。

1 Hibernate 技术

1.1 Hibernate 框架

Hibernate 是采用 ORM 映射机制进行持久层数据开发的工具, 它是 Java 应用程序和关系数据库中间的桥梁, 负责对 Java 对象和关系型数据之间映射^[2]。其架构体系如图 1 所示。

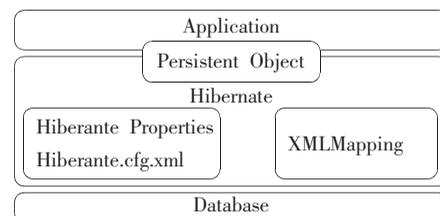


图 1 Hibernate 数据持久化应用结构

Hibernate 内部封装 JDBC 进行访问数据库操作, 其向上层应用对象提供面向对象的数据库访问 API, 使开

欢迎网上投稿 www.pcachina.com 13

《微型机与应用》2011 年第 30 卷第 15 期

* 基金项目: 辽宁省教育厅科学研究项目(2008366)

发者能充分运用面向对象的编程思维操作数据库,而无需关心底层数据库操作。Hibernate 自身通过 hibernate.cfg.xml 和类的映射文件将类和数据库相映射,应用程序通过 Hibernate 及持久化对象类直接访问底层数据库。

1.2 Hibernate 核心接口

利用 Hibernate 进行数据持久化操作,至少会用到下列核心接口:Configuration 接口负责配置启动 Hibernate 并创建 SessionFactory 对象;SessionFactory 接口产生 Session 实例的工厂类,负责初始化 Hibernate 并创建 Session 对象;Session 接口是 Hibernate 进行持久化操作的基础,相当于 JDBC 中的 Connection 对象所起的作用。Session 提供了一系列的持久化操作方法,如保存、更新、删除、查询等;Transaction 接口负责管理事务;Query 和 Criteria 接口负责执行数据库查询^[3]。

2 DAO 设计模式

DAO(Data Access Object)模式称为数据访问对象模式,是 Java EE 核心模式之一,其主要的功能是在业务核心方法和具体数据源之间再增加一层 DAO 接口及其实现类。该模式的本质是向外部提供一个访问数据源的统一接口,对外隐藏操作数据源的实现细节,以此实现业务逻辑层与 DB 的解耦^[4]。这是因为在实际的应用过程中,应用程序所面对的数据源往往是多种多样的,不同数据源的连接方式、数据访问方式会有明显差异,导致了需要访问数据源的组件的代码实现方式与数据源的类型有着密切的关系,组件和数据源之间的这种紧耦合关系也就导致了整个应用系统难以在不同数据源之间进行迁移。使用了 DAO 模式后,即使系统需要进行数据源的迁移,也只需在 DAO 模式内部进行数据源访问代码的修改,而不会涉及上层调用代码,从而提高了软件可维护性。

3 数据持久层架构设计及实现

3.1 数据持久层框架设计

在应用程序和数据库之间构建数据持久层,可降低 J2EE 应用与数据库的耦合度,并简化程序开发。基于 Hibernate 框架采用 DAO 模式设计的数据持久层整体架构如图 2 所示。

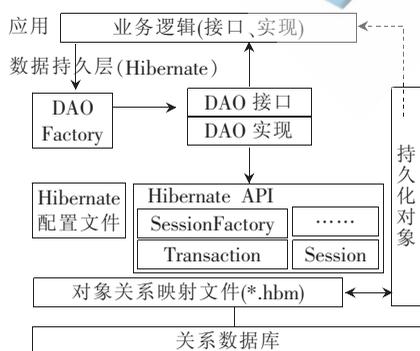


图2 数据持久层架构

业务逻辑是整个框架的业务处理的核心部分,由业务逻辑接口、接口实现类和配置文件等组成。它接收控制层的各类调用请求,控制着系统逻辑的实现和处理,并通过调用数据持久层的 DAO 接口完成数据的存取操作。业务逻辑接口向控制层或其他业务逻辑接口实现类提供各种业务逻辑接口方法,接口实现类则对这些已定义的接口进行具体实现。

数据持久层使用 DAO 来抽象和封装所有对数据源的访问,DAO 管理着数据源的连接以便检索和存储数据,DAO 通过 DAO 工厂管理,DAO 工厂负责生成 DAO 组件实例,并维护 DAO 实例;业务逻辑组件不与 DAO 组件的实现类耦合,只与 DAO 组件的接口耦合;DAO 实现类中封装 Hibernate API 来完成与持久化类相关的业务逻辑操作。

Hibernate 配置文件的主要工作是进行 SessionFactory 配置、关系型数据库“方言”与加载持久化类对应的映射文件等。映射文件用来声明 Hibernate 中持久化类的属性与数据库对应表之间字段的映射关系。

数据持久化层架构的操作流程为:业务逻辑模块调用 DAOFactory,DAOFactory 生成相应的 DAO 接口及实现类,DAO 实现类中封装 Hibernate API 来访问持久化类,通过 O/R 映射文件声明持久化类与关系数据库的映射,最终实现对数据库的操作。

3.2 数据持久层框架的应用——解决方案及实现

本设计的数据持久层架构,作为数据持久层解决方案应用于问卷调查系统,最大限度地降低了系统内部各模块、子系统之间的耦合性,保证了持久层的业务逻辑层相对稳定,基本不需要持久层的调整改变而进行逻辑层的变动。问卷调查系统由问题管理、问卷管理、统计管理、用户管理、学生信息管理等模块构成,其数据持久层的具体组织结构如图 3 所示^[5]。

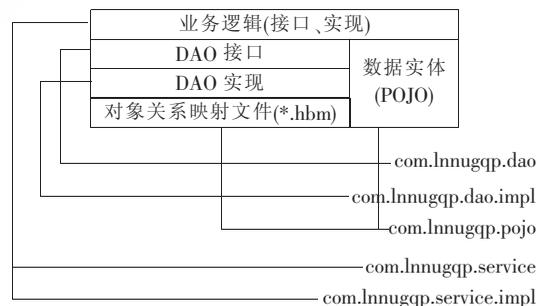


图3 问卷调查系统数据持久层组织结构

4 数据持久层的实现

4.1 持久化类及对象/关系映射文件

4.1.1 持久化类

持久化类的对象 PO(Persistent Object)是一个简单 Java 对象(POJO),一个 PO 代表了与数据库表中某条记录相对应的 Hibernate 实体,PO 的变化在事务提交时反映到实际的数据库表中。Hibernate 中的持久化类符合

《微型机与应用》2011年第30卷第15期

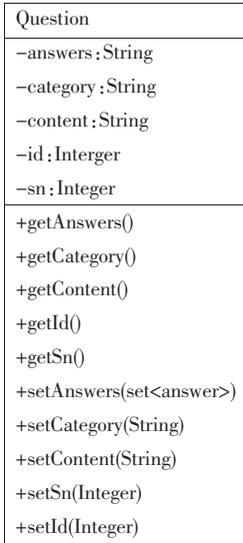


图4 实体类模型图

JavaBean 的规范,包括一些属性和对属性的访问方法。图4是问卷调查系统中 Question 实体类的模型图^[6]。可以看出,除了无意义的主属性只能被获取以外,每一个属性都有被外部对象获取(get)和设置(set)两种方法,通过它们可以获得对象的属性值并实现对象之间的导航。

4.1.2 对象/关系映射文件

映射文件用一个 XML 格式的映射描述文档来声明 Hibernate 中持久化类的属性与数据库对应表之间字段的映射关系,该文件中声明的属性名称

要与对应数据表中的字段名称相同,同时也要与持久化类中所包含的属性名称相同。

4.2 Hibernate 配置

为了用 Hibernate 进行持久化实现,还需要在应用程序的包里放置一个配置文件,以便 Hibernate 能正确地完数据库连接等各种配置和初始化。由于映射文件可以在 XML 格式的配置文件中进行声明,在程序中不必调用 Configuration 类的 addClass 方法来加载映射文件,因此这种格式的配置文件可以提高应用程序的可维护性。

4.3 数据持久层的 DAO 接口及实现类

数据持久化层采用 Hibernate 作为中间件,使用 DAO 设计模式实现对数据库的访问。DAO 模式的实现包括 DAO 接口和 DAO 实现类两部分。DAO 接口负责定义访问特定持久化类所对应的抽象业务逻辑方法;DAO 实现类负责利用 Hibernate API 实现 DAO 接口所定义的抽象方法的具体实现。

问卷调查系统在实现 DAO 模式时,首先创建了 HibernateBasicDao,该类继承 Spring 框架提供的 HibernateDaoSupport 类,HibernateBasicDao 类封装了对数据库持久化的操作,任何它的子类只要使用它的方法就可以实现对数据库的持久化操作。最后使每个 DAO 实现类都继承 HibernateBasicDao 类实现数据库的持久化操作。

图5描述了问题 DAO 接口及其实现类,它封装了对问题的基本操作。其中,delete 方法删除问题信息,getId 方法通过 question 的 Id 获得,getCurrentPage 方法通过当前页数获得 question 的具体类,query 方法更新问题信息等。限于篇幅,其他 DAO 接口及实现类的设计不一一给出。

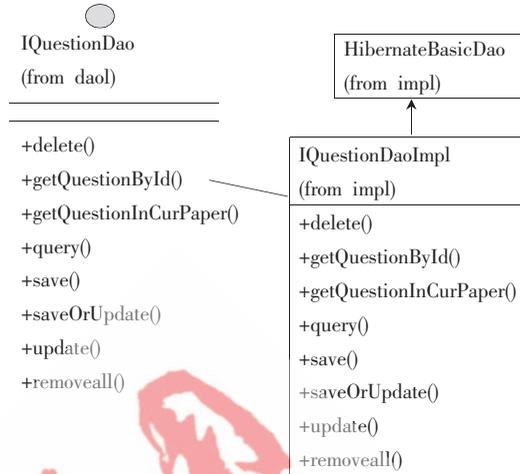


图5 问题 DAO 接口及其实现类

在基于 J2EE 的应用中,系统的数据持久层代表着某个系统中一个相对独立界限明确的逻辑层次,在这个层次中,将数据的使用者和数据实体相关联,负责对数据库存储数据,同时也负责数据的检索、更新和删除,实现数据的读取和持久化操作。本文运用 Hibernate 框架技术设计了一套数据持久层架构体系,并将其作为数据持久层解决方案应用于问卷调查系统。实践结果表明,数据持久层架构的应用,降低了数据持久逻辑与业务逻辑的耦合度,简化了开发过程,优化了数据访问操作,增强了系统扩展性和可维护性。

参考文献

- [1] 郭善飞. 基于 J2EE 的数据持久层组件的设计与实现[D]. 北京:北京邮电大学,2009.
- [2] 张飞,张建. 基于 Spring 与 Hibernate 的数据库访问技术研究[J]. 计算机工程与设计,2009,30(7):1669.
- [3] 汪萌,曲俊华. 基于 Hibernate 技术的持久层解决方案及实现[J]. 计算机系统应用,2010,19(3):155.
- [4] 欧阳宏基. 一种基于 DAO 设计模式与 Hibernate 框架的数据持久化层模型[J]. 微计算机应用,2009,30(3):36-37.
- [5] 严海. 基于 Struts+Spring+Hibernate 框架构建 WEB 应用的设计与实现[D]. 西安:西安电子科技大学,2010.
- [6] 吴京慧. 基于 Hibernate 对象持久化 Web 应用的研究[J]. 计算机应用与软件,2009,26(2):90-91.

(收稿日期:2011-03-14)

作者简介:

刘德山,男,1970年生,副教授,主要研究方向:计算机软件与理论,数据库应用。

杨彬彬,女,1977年生,硕士,讲师,主要研究方向:数据库技术与决策支持系统。