

基于 MVC 模式的网络教学平台设计

匡芳君^{1,2}

(1. 南京理工大学 计算机科学与技术学院, 江苏 南京 210094;
2. 湖南安全技术职业学院 计算机科学与技术系, 湖南 长沙 410151)

摘要:针对目前国内现有网络教学系统的性能、可扩展性和重用性差等方面的不足,提出了基于 MVC 模式的网络教学平台。首先分析了 MVC, Struts, Spring 和 Hibernate 等技术的特点,然后分析了基于 MVC, Struts, Spring 和 Hibernate 等技术的网络教学平台的系统总体结构、软件架构、业务数据流程及系统实现。该平台将 Struts, Spring 和 Hibernate 等开放源代码的软件框架整合应用于网络教学平台,成功地将数据存储、数据显示及业务处理逻辑分离,有利于提高代码的重用性以及软件的开发效率,并有效地屏蔽了后台数据库的数据结构,提高系统的安全性。

关键词: MVC 模式;网络教学平台;Struts 框架;Spring;Hibernate

中图分类号: TP391.41

文献标识码: A

文章编号: 1673-629X(2010)12-0251-04

Design of Web Instruction Platform Based on MVC Pattern

KUANG Fang-jun^{1,2}

(1. Department of Computer Science and Technology, Nanjing University of Science and Technology,
Nanjing 210094, China;

2. Department of Computer Science and Technology,
Hunan Vocational Institute of Safety & Technology, Changsha 410151, China)

Abstract: Web instruction platform based on MVC multitier structure is proposed in order to solve the problem that the performance, scalability and reusability of the domestic existing network teaching system are poor. It firstly analyzes the feature of MVC, Struts, Spring and Hibernate techniques. Then brings forward system total architecture, software framework, business data flow and realization of web instruction platform based on MVC, Struts, Spring and Hibernate techniques. It combines Struts, Spring and Hibernate soft framework into the development of web instruction platform. It successfully divides data storage, data display and logical transaction and improves greatly the reusing attributes and exploitive efficiency of software. It can encapsulate entity bean and improve stability and security.

Key words: MVC pattern; web instruction platform; Struts framework; Spring; Hibernate

0 引言

网络教学平台^[1]是指具有教学功能的计算机软件,在某种程度上能代替或辅助教师实施教学。它与传统课堂教育方式相比,最重要的特点在于突破了时间和空间的限制,不再完全依赖教师和学生之间面对面的教学。但当前网络教学平台多数使用 B/S 模式,在网页中嵌入 ASP/JSP 代码段完成对数据库的存取和数据查询和计算功能,以实现用户身份认证、在线答疑、讨论及在线测试与试题库管理等功能。在此网络教学平台中,只是简单地将表示层、业务层、数据层耦

合在一起,逻辑分工不明确,代码的重用性差,同时,直接利用网页中的 ASP/JSP 代码访问数据库,存在一定的安全隐患。因此,设计和开发一个基于 MVC 的网络教学系统,可以有效地分离表示层、业务逻辑层、数据持久层,从而增强网络教学平台的稳定性、安全性和伸缩性,提高代码的可重用性、软件的开发效率和系统的安全性。

1 MVC 模式

MVC(Model View Controller,模型-视图-控制)模式是一种将表示逻辑和业务逻辑分离的设计模式,旨在分离用户界面显示,用户输入控制和底层的信息应用^[2]。其处理流程是:首先由控制器接收用户请求,并决定应调用哪个模型进行处理,然后模型用业务逻辑处理用户请求并返回数据,最后控制器用相应视图

收稿日期:2010-04-09;修回日期:2010-07-16

基金项目:湖南省教育科研项目(09C1105,09C1106)

作者简介:匡芳君(1976-),女,湖南长沙人,讲师,博士生,CCF 会员,研究方向为软件工程、模式识别、图像处理、信息安全。

返回数据,并通过网页呈现给用户。

基于 B/S 的 Web 应用向软件开发人员提出了一些特有的挑战,最明显的是客户机和服务器的无状态连接,这种无状态行为使得模型很难将更改通知视图。在 Web 上为了获得对应用程序状态的修改,需要修改标准的 MVC 模式,通常称为 MVC Model2,其结构如图 1 所示。

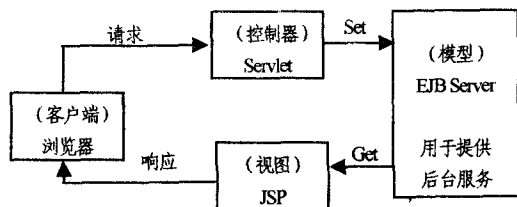


图 1 MVC Model2

2 网络教学平台总体设计

网络教学平台由教师模块、学生模块和教学管理模块三大部分组成,这些模块之间相互联系,相互配合,构成一个完整的网络教学系统。系统总体结构图如图 2 所示。

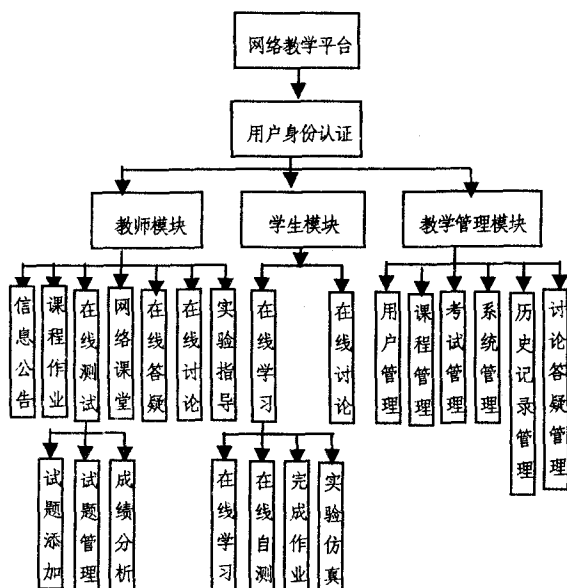


图 2 网络教学平台系统总体结构图

3 网络教学平台关键技术设计

3.1 总体设计思路

本平台利用 Dreamweaver 等工具开发网页中静态部分,用 flash 开发动画,利用 Java, Jsp, JDBC 开发动态、交互性部分,在网站的构建中使用开放源代码的软件框架 Struts^[3], Spring^[4,5] 和 Hibernate^[6]。

3.2 层次结构的划分

网络教学平台层次结构划分为表示层、业务逻辑

层、数据持久层和领域模型层四层,每一层都有明确责任和分工,系统架构如图 3 所示。

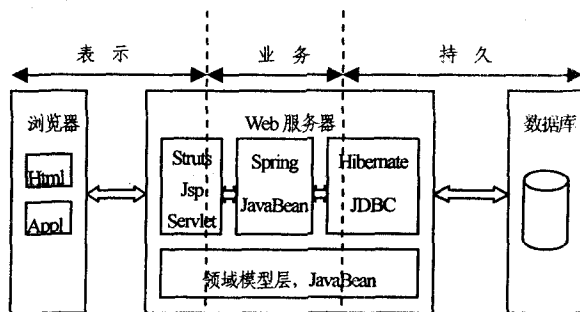


图 3 基于 MVC 模式的网络教学平台系统架构图

业务逻辑层使用 Facade^[7] 模式实现表示层与业务逻辑层之间接口的统一,且可使用 Business Delegate 模式解决表示层与业务逻辑层之间的高度耦合性问题;而数据持久层使用 DAO 模式为业务逻辑层提供数据抽象层接口,实现了数据存储逻辑的分离,数据访问底层实现的分离、资源管理和调度的分离及数据抽象的目的,从而达到了表示层与业务逻辑层、业务逻辑层与数据持久层间的松散耦合和低依赖。

(1) 表示层。

表示层是最靠近用户的一层,负责向用户展示一个交互界面,管理用户的请求,做出相应的响应,对用户请求委派调用业务逻辑和其它上层处理。本层使用 Struts 技术完成与用户的交互,并根据网络教学平台的实际需要,对 Struts 框架进行了扩展,在 BaseAction 中使用了策略设计模式和回调模式,并定义了应用中所有 Action 的公共逻辑,从而保证了应用程序的松散耦合,提高了代码的可重用性。

(2) 数据持久层。

数据持久层是距离用户最远的一个层次,它封装了数据访问细节,为业务逻辑层提供服务,并为 Java 提供持久化的对象——关系映射,把程序中处理的 Java Bean 对象映射到关系数据库的实体。本层使用 Hibernate 技术提供基础性的关系数据库管理,向上则提供基于对象的处理方式,所以程序开发者无需编写具体的访问数据库的 SQL 语句。同时, Hibernate 具有很好的跨数据库能力,且可移植性强,对当前流行的数据库可用相应的 Dialect 进行优化,从而提高了系统的综合利用效率。

(3) 业务逻辑层。

业务逻辑层是处理应用程序的业务逻辑和业务校验,管理事务,管理业务层级对象的依赖,在表示层和数据持久层之间增加了一个灵活的机制,使得它们不能直接联系在一起,只能通过揭示从表示层到业务逻辑层之间的上下文环境来得到业务逻辑层的服务。

和管理程序的执行。系统为学习模块中的业务逻辑层定义了一个接口及接口的实现类,相关组件则基于接口定义,采用这种面向接口而非面向类的编程,可以减少组件间的耦合度,且能极大提升开发过程中测试的灵活性。本层使用 Spring 技术分别向持久层和表示层提供会话、事务支持和业务逻辑服务。

(4)领域模型层。

领域模型层是由系统分析所得到的业务领域中的业务对象集合,它能在不同层之间移动,在 MVC 的各个层次上交换数据信息或表示领域对象。领域模型层不仅能为表示层提供其所需的数据源,而且能为数据持久层提供被持久的对象。系统把对 StudyDAO 接口有显式依赖,但重用度低的且与领域对象关系不太密切的对象封装到 service 层中,而把重用度高且与领域对象密切相关的业务逻辑封装到领域对象模型中,从而避免了领域对象与其他应用组件耦合情况出现,使其重用性大大增强,进而保证了各层间的单向依赖性。

3.3 系统开发关键技术分析

(1)Java applet 技术。

Java applet 是一种 WWW 浏览中运行的 JAVA 小应用程序,Java applet 直接嵌入到 HTML 语言中,通过主页发布到 Internet,并可以创建集音乐、声音和动画于一体的多媒体 WEB 页面。

(2)JSP 技术。

JSP(Java Server Pages)页面是由 HTML 代码和嵌入其中的 Java 脚本(scriptlets)和标记(tags)组成,它可以通过这些标记和脚本来访问服务器端的资源 and 数据。同时,它也可使用 Java 语言编写类 XML 的标记和脚本来封装产生动态网页的处理逻辑,从而实现了网页设计、显示和网页逻辑的分离。在网络教学平台中,JSP 技术主要用于用户身份认证、在线讨论、在线答疑和在线测试等方面的设计与开发。

(3)JavaBeans 技术。

JavaBean 是用 JAVA 语言编写的一种可重用且能跨平台使用的组件,它可被 JSP、Applet、JavaScript 等应用程序调用,在 JSP 中,可将一些繁琐或经常重复使用的操作封装成 JavaBean。使用 JavaBean 可以充分利用组件的可重用性,简化 JSP 代码,使页面代码和逻辑代码互相分离,增加代码的可读性,更便于查看和调试。如网络教学平台中的在线讨论、文件上传、下载等功能的实现。

(4)JDBC 技术。

JDBC(Java Database Connectivity)是一组允许 Java 与 SQL 数据交互的 API 应用程序设计接口,为各种数据库提供统一的访问,是一种 Java 数据库连接技术,

可以实现底层代码与数据库的交互。本平台是通过 JDBC 与数据库相连,来执行查询、追加、删除等操作。

(5)Struts 技术。

Struts 采用了基于 Web 的模型-视图-控制器(Model-View-Controller, MVC)模式,由一组相互协作的类(组件)、Servlet 以及 JSP, taglib 组成,其中:Servlet 用于 HTTP 请求的分配及 JSP 的调用;标记库用于页面的动态生成;实用类库用于 XML 的解析及 Bean 的属性设定。它利用控制器来分离模型和视图,达到一种层间松散耦合的效果,提高系统灵活性、复用性和可维护性。Struts 框架在视图方面提供了 JSP 自定义标签,为设计用户界面提供了方便,并且可把用户的请求封装成 Action,通过控制器统一处理客户的请求。

(6)Hibernate 技术。

Hibernate 是 Java 平台上的一种全功能的、开放源代码的映射框架。传统的使用 Java 构建的动态网站主要使用 JDBC 技术和数据库直接交互。程序员需要编写具体的访问数据库的 Sql 语句,比如查询、删除、更新和插入等,开发效率比较低,容易出错,不能完全使用面向对象技术。Hibernate 框架可以构建 JavaBean 对象和关系数据库实体之间的映射,是构建于 JDBC 技术之上的访问数据库的软件框架,通过 XML 配置文件建立 Java 对象和数据库实体之间的映射关系,自动生成访问数据库的各种语句,自动实现 Java 对象到关系数据库的持久化,且提供事务管理。使用 Hibernate 后,程序员可以像访问一般 Java 对象一样访问数据库。

(7)Spring 技术。

Spring 也是一个软件框架,使用了依赖注入机制,可以对 Java 对象进行动态组合,创建一个可以构造对象的应用环境,然后向这些对象传递它们的协作对象。在本网络教学平台中,Spring 处于业务层,管理应用程序业务逻辑和业务校验,管理事务。控制域模型对象在不同层之间的流动。图 3 中, Spring 处于整个系统的业务层, Struts 把收集到的用户请求转发到 Spring 框架,按照业务规则进行处理。如果需要访问数据库,则由 Spring 调用 Hibernate。

(8)SSH 整合。

SSH 整合即 Struts + Spring + Hibernate 的整合,在该架构中,系统的基本业务流程为:在 Web 表示层,首先通过 JSP 页面对业务数据进行呈现,并对用户的非法操作进行适当的处理,同时,负责发送请求和接收响应,然后 Struts 根据配置文件将接收到的请求委派给相应的 Action,以用于处理页面和业务逻辑的交互,并

对数据进行必要的验证和转换。在业务层, Spring 管理服务层组件, 并负责向 Action 提供调用业务模型组件和 DAO 组件以完成业务逻辑, 并提供事务处理、缓冲池等容器组件以保证数据的完整性和提高系统性能。在数据持久层, Hibernate 则根据模块需要的各种持久化操作, 确定对应的 DAO 操作, 并利用其对象化映射和数据库交互, 分析模块所涉及的表, 确定表之间的关系, 并处理 DAO 组件请求的数据, 返回处理结果。整个业务逻辑层使用整合 SSH 架构^[8-12], 即除了包含 Spring 外, 还包括 Struts 和 Hibernate 的一部分, 如图 3 中的虚线。采用 SSH 整合开发模型, 不仅实现了业务逻辑层与数据持久层的分离, 还实现了视图、控制器与模型的分离, 而且数据库的变化也不会影响前端, 从而大大提高了系统的可重用性。

4 SSH 整合框架在网络教学平台中的应用

下面是以网络教学平台的用户身份认证系统来说明 SSH 框架的实现。

用户身份认证是指用户登录网络教学平台时的身份验证过程。在本网络教学平台中, 用户身份有管理员、教师和学生三种。用户首先从 logon.jsp 页面输入登陆信息, 并通过 struts 调用 staticsQueryAction 的 createOperateLog() 方法获取用户输入信息, 以便判断登录用户类型是管理员、教师还是学生, 再通过不同方法调用数据访问层中的 DAO 从而获取登录用户的具体信息。然后利用 Hibernate 访问具体的数据表来获取持久化对象, 并处理 DAO 中的具体数据, 从而实现从 java 类到数据表之间的映射, staticsQueryAction 将返回的登录用户对象放到 session 会话上, 以便检查用户是否登录, 再通过 struts 返回到 logon.jsp 页面。

在设计用户身份认证系统的查询与统计功能时, 采用 spring 的 ioc 框架定义接口变量和变量所对应的 setter 方法, 而实例化对象的具体操作则由 spring 完成, 即 spring 在运行时读取配置文件来完成对象实例化。在查询时, 先利用 spring 创建 bean, 应用业务注入实现数据持久层的通信, 并配置注入 DWR 所需要的且能供 DWR 框架将 Java 代码编译成 Js 时使用的 DAO; 然后, 设置 hibernate 的属性, 并确定 hibernate 使用的数据库语句; 最后配置 struts 中 action 所需要使用的

的 DAO 和 service。

5 结束语

基于 MVC 模式的网络教学平台运用多种模式的设计, 采用松散耦合的编程体系结构, 分离了业务逻辑层、数据持久层和表示层, 有效地减少了远程调用的次数, 提高了系统响应速度。实践证明, 本平台采用了先进的 MVC 模式开发、分层架构, 简化了 Web 应用程序的开发过程, 提高开发效率, 缩短开发周期, 提高了系统的可扩展性; 同时, 对数据对象的封装存储, 有效地屏蔽了后台数据库的数据结构, 提高了系统的安全性。

参考文献:

- [1] 袁守华. 基于 Web 的课程网络教学网站的开发与应用[J]. 中原工学院学报, 2004, 15(1): 49-53.
- [2] 邱旭东, 刘文浩. 基于 JSP 的 MVC 改进模式研究及应用[J]. 计算机技术与发展, 2006, 16(8): 134-136.
- [3] Davis M, Consultant. Struts, an open-source MVC implementation[EB/OL]. 2001. <http://www.ibm.com/developerworks/java/library/j-struts/index.html>.
- [4] Harrop R. Spring 专业开发指南[M]. 成都: 电子工业出版社, 2006.
- [5] Johnson R, Hoeller J. The Spring Framework Reference Documentation[EB/OL]. 2008. <http://static.springframework.org/spring/docs/2.5.x/reference/>.
- [6] Hibernate Reference Documentation 3.5.1[EB/OL]. 2007. <http://docs.jboss.org/hibernate/stable/core/reference/en/pdf/hibernate-reference.pdf>.
- [7] Alur D, Crupi J, Malks D. Core J2EE Patterns: Best Practices and Design Strategies[M]. American: Prentice Hall, 2003.
- [8] 韩义亭, 张成宇. SSH 架构及其在 Web 开发中的应用[J]. 网络安全技术与应用, 2007(10): 72-74.
- [9] 李腊元, 徐鹏. 基于 MVC 模式的 JSF, Spring 和 Hibernate 整合[J]. 计算机技术与发展, 2008, 18(3): 46-49.
- [10] 廖胜军. 基于 SSH 整合架构的应用研究[D]. 武汉: 武汉理工大学, 2008.
- [11] 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): 1668-1670.
- [12] 李宁. Java Web 开发技术大全—JSP + Servlet + Struts 2 + Hibernate + Spring + AJAX[M]. 北京: 清华大学出版社, 2009.
- [11] Brands S. Untraceable off-line cash in wallets with observers[C]//Advanced in Cryptology CRYPTO'93. [s.l.]: Springer-Verlag, 1993: 302-318.
- [12] 王大星, 杜育松, 沈静. 公正可分的多银行电子现金支付方案[J]. 计算机工程, 2007, 33(16): 126-127.
- [13] 邹候文, 王峰, 唐屹. 椭圆曲线点乘 IP 核的设计与实现[J]. 计算机应用, 2006, 26(9): 2131-2136.

(上接第 224 页)