

# 一种基于双 MVC 架构的远程教学管理系统的设计与实现

## Design and Implementation of Distance Teaching Management System Based on Double MVC Structure

刘运龙 (湖南师范大学继续教育学院 长沙 410012)

黄烟波 (中南大学网络中心 长沙 410012)

黄 海 (湖南师范大学继续教育学院 长沙 410012)

**摘要:**本文介绍了一种基于双 MVC 架构的远程教学管理系统,其详细内容包括系统的主要功能、设计要点及具体实现,最后简要分析了该系统的性能特点。

**关键词:**远程教学 双 MVC 架构

### 1 引言

随着多媒体和网络应用的迅速普及,现代远程教育日益发展。从技术的角度上讲,当前现代远程教育的主流模式是基于 Web 的 B/S 模式,即各地的学员可随时随地通过浏览器点播学校中心服务器上的多媒体教学资源进行学习。现代远程教育以其教学资源的充分共享、教学素材的生动逼真、学习方式的灵活多样等独特优势对传统教育带来了深刻的变革,并在近年来取得了迅速发展。

开展现代远程教育的基础工程是为教师、学员、管理员开发一个网络化、信息化的教学管理平台,对各类教学资源和各种相关信息进行有效管理。管理信息系统的性能直接反映远程教育的现代化水平和影响学校现代远程教育的质量。为此,本文设计并实现了一种基于两个 MVC (Model - View - Controller) 架构的现代远程教学管理信息系统,实际运行时具有良好的性能和独特的优点。

### 2 系统主要功能

对远程教学及其相关活动的管理既是一项以教学为中心的全面管理,又是一项以教学为重点的综合性

管理。对其需求的深入了解和全面分析是系统开发的基础。

本系统是结合湖南师范大学现代远程教育中学教师专升本学历培训的教学实践而开发的。其总体功能是为该校远程教学中的教师、学员、学校管理员、站点管理员开发一个教学管理平台,系统结构上可分为学员操作模块、教师操作模块、学校级管理员操作模块、站点级管理员操作模块四个部分。

#### 2.1 学校管理员操作模块

此模块是系统中的最主要模块,实现了远程教学的一系列相关辅助活动的信息化管理。具体包括教务管理、学籍管理、成绩管理、经费管理、系统管理和各类教学信息发布等各个方面。为保证数据的安全性,管理系统采取了分级管理和分类管理相结合的策略。教务管理员、学籍管理员、成绩管理员只拥有与自己职责相对应的权限,限于在相应的操作子模块上进行查询、增加、修改、删除、统计和打印等管理活动,并且他们的操作代码模块和操作数据库表都是相互独立的。系统管理员拥有最高权限,负责各类用户的管理(如增加用户、删除用户、分配权限),负责整个系统的安全管理和故障管理,维护系统的正常运行。

## 2.2 站点管理员操作模块

教学站点(即校外学习中心)的主要职能是配合学校组织教学和协助学校开展一些管理活动。站点管理员在此模块可对本站点和各类相关数据进行查询、检索、下载、修改和信息集成。具体包括查询本站点学员的学籍信息、成绩信息,发布本站点的集中教学安排,网上报名,修改本站点的相关数据等。

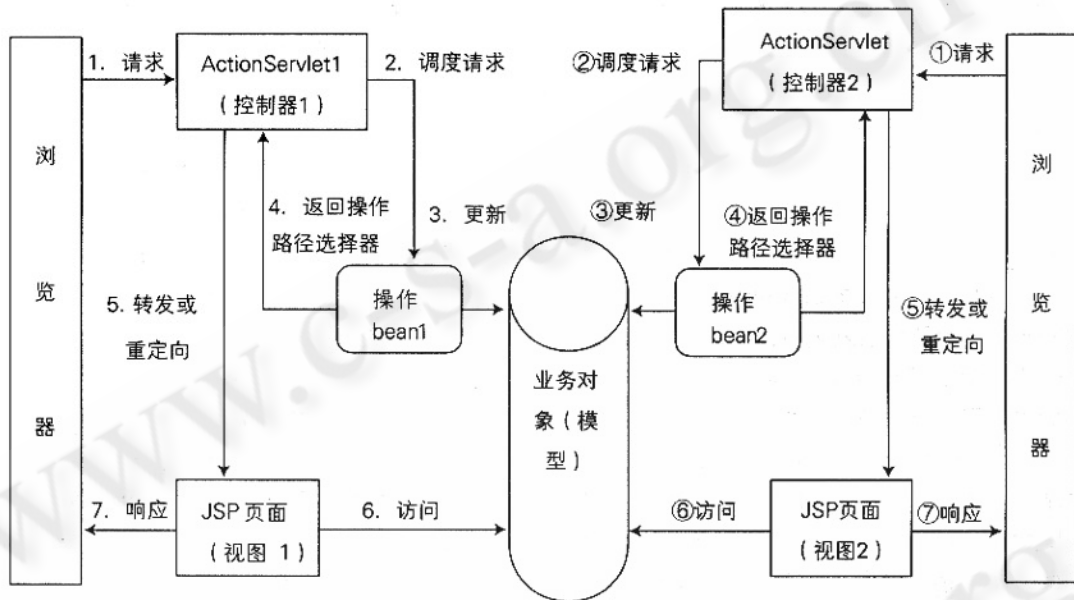


图 1 数据流程图

## 2.3 学员操作模块

此模块的主要功能是为学员提供基于网络的学习途径。经过注册的在籍学员进入系统后可选择多种学习形式,如点播视频课件,参加实时网络答疑,参加网上论坛,检索各种学习资料,浏览各类教学信息,下载部分资料,查询考试成绩等。其中流媒体视频课件点播是最主要的学习形式,学员可以随时随地从教学计划中选择任意一门课程进行学习,并且可以从任意章节开始学习任意遍数。另外,学员对自己学籍档案信息的变化(如工作单位的调整、联系电话、政治面貌的变化等)都可及时进行修改。

## 2.4 教师操作模块

在基于 B/S 模式的远程教学中教师的主要教学活动可分为实时教学和非实时教学。其中实时教学主要是网上实时交互答疑,非实时教学包括课程讲授(先由

第三方录制成流媒体视频课件,再将课件上传或直接拷贝到课件中心服务器上),拟定考试大纲,布置作业练习、模拟试题和参加网上论坛。本系统能方便、快捷地实现上述各种教学。

## 3 系统设计要点及具体实现

本系统的开发模型选择了当前流行的 B/S 三层结构模型。

其中第一层客户端使用 IE 等标准浏览器,第二层 Web 服务器采用 Tomcat5.0,第三层数据库服务器选用 Microsoft SQL Server7.0,介于 Web 服务器和数据库服务器之间的中间件技术采用 JSP (Java Server

Pages) 技术,服务器端操作系统选用 Windows 2000 Advanced Server。

在系统的结构模型上,本系统改进了经典的 MVC 架构,设计并实现了一种同时含有两个 MVC 架构的混合型架构,其具体架构及数据在此架构中的流程如图 1 所示。

从图 1 可看出本系统具有下述结构特点:

① 是一种含有两个 MVC 架构的混合型结构;

② 系统中的两个 MVC 架构是不完全相同的,其主要不同点是两个架构中控制器的设计不完全相同。其中的一个运行步骤少、速度快但不具有事件处理功能(即图中的 ActionServlet1),另一个运行步骤较多、整体速度相对较慢但具有事件处理功能(即图中的 ActionServlet)。事件处理的类型有捕获敏感表单的重复提交、字符国际化和身份验证等。所有不需作事件处

理的操作均由 `ActionServlet1` 控制执行,所有需要作事件处理的操作均由 `ActionServlet` 控制执行。

③ 两个 MVC 架构不是简单混合,而是两者的有机结合。两者共用一个初始化文件 (`web.xml`), 共用一个属性文件 (`actions.properties`), 共用业务对象 (即模型)。

④ 两个 MVC 架构既可独立运行也可并行运行。为便于区分,本文将上图左边不作事件处理的 MVC 架构称作 MVC1, 上图右边能进行事件处理的 MVC 架构称作 MVC2。

### 3.1 模型 (Model) 设计

本系统的模型 (即业务对象) 由后台数据库和一些 `JavaBean` 组成。它们被两个 MVC 架构共用, 其中 MVC1 架构实现对数据库表的查询、修改和删除等操作, MVC2 架构实现对相应数据库表的插入操作。

数据库是远程教学管理信息系统实现的基础。根据数据库设计的一般原则的实际需要, 本系统中设计了一个名为 `edudata` 的数据库, 该数据库含有教师用户表 `Table_teacher`、专业表 `Table_major`、教学计划表 `Table_courses1`、教师课程表 `Table_courses2`、模拟试题表 `Table_test`、课件表 `Table_ware`、答疑纪要表 `Table_answer`、网上自由论坛表 `Table_bbs`、站点教学安排表 `Table_planning`、在籍学员表 `Table_students`、成绩表 `Table_cj` 等十多种数据库基本表, 实现对相关数据的管理。

`JavaBean` 是一个维护一些数据 (或称为属性) 并遵循编码约定的类, 是建立可重用 `Java` 组件的流行技术<sup>[1]</sup>。`JavaBean` 通常有两种类型: 一种 `JavaBean` 其内部代码根据 `bean` 的定义只含有属性、设置方法和获取方法三部分; 另一种类型是广义上的 `bean`, 它在前一类型的基础上封装了一些其他业务操作。本系统主要是第二种类型的应用, 例如系统中的 `LoginDB` 是一个包含连接数据库表并判断是否有数据的 `bean` 类, `ISOtoGb2312` 是一个关于汉字编码转换的 `bean` 类, `DbconnectionPool` 是一个关于建立数据库连接池的 `bean` 类。

### 3.2 视图 (View) 设计

本系统的视图主要为 JSP 页面, 它们分别存放在目录 `jsp1`、`jsp2`、`jsp3`、`jsp4` 中, 其中 `jsp1` 目录存放学员可能访问到的所有 JSP 页面, `jsp2` 目录存放了学校级管理员可能访问到的所有 JSP 页面, `jsp3` 目录存放了教师可能访问到的所有 JSP 页面, `jsp4` 目录存放了站点管理员可

能访问到的所有 JSP 页面。就系统架构而言, 四个目录中的页面既含有属于 MVC1 的 JSP 页面, 又含有属于 MVC2 的 JSP 页面。各目录中与查询、修改、删除数据有关的显示页面均属 MVC1 架构, 与插入数据有关的显示页面属于 MVC2 架构。典型的页面代码流程为: 建立数据库连接, 产生语句和结果集, 循环调用结果集中的数据, 列表显示, 关闭结果集、语句和连接。为了提高页面的可维护性、可扩展性和可重用性, 设计页面格式时使用了节、区域和模板等组件对象, 其中页面区域使用了同一模板文件 (`/jsp/templates/hscf.jsp`), 页面格式包含标题、侧框、内容和页脚 4 个部分。

为了消除 JSP 页面与操作类之间的直接关联, 本系统通过使用逻辑名的技巧来解决。逻辑名与实际操作类和实际 JSP 页面的映射通过位于 `/WEB-INF/classes` 下的属性文件 `actions.properties` 来实现。

### 3.3 控制器 (Controller) 设计

本系统中两个 MVC 架构的控制器由两个 `Servlet` 和一些操作 `bean` 组成。其中 `ActionServlet1` 是 MVC1 框架的控制器, 用来处理所有以 `.do1` 结尾的 HTTP 请求; `ActionServlet` 是 MVC2 框架的控制器, 用来处理所有以 `.do` 结尾的 HTTP 请求。具体来说, MVC1 中 `ActionServlet1` 的服务方法的主要步骤及代码为<sup>[2]</sup>:

```
public class ActionServlet1 extends HttpServlet {
    .....
    public void service (HttpServletRequest req, HttpServletResponse res)
        throws java.io.IOException, ServletException
    {
        Action action = factory.getAction (actionClass,
            getClass().getClassLoader());
        ActionRouter router = action.perform (this, req,
            res);
        router.route (this, req, res);
    }
    ..... }
```

MVC2 中 `ActionServlet` 的服务方法的主要步骤及代码为<sup>[2]</sup>:

```
public class ActionServlet extends HttpServlet {
    .....
    public void service (HttpServletRequest req, HttpS-
```

```

ServletResponse res)
    throws ServletException {
    Action action = getAction(req);
    ActionEvent event = new ActionEvent(action,
        ActionEvent.BEFORE_ACTION, req, res);
    action.fireEvent(event);
    ActionRouter router = performAction(action,
    req, res);
    event.setEventType(ActionEvent.AFTER_ACTION);
    action.fireEvent(event);
    routeAction(router, req, res); }
    .....}

```

系统对模型(业务对象)进行处理、对事件作出反应和转向视图的许多工作是由具体的业务操作来完成的。本系统中的所有业务操作放在 actions1 和目录 actions 中,其中 actions1 目录存放 MVC1 架构的构成类和不需要进行事件处理的各种业务操作类,actions 目录存放 MVC2 架构的构成类和需要进行事件处理的各种业务操作类。例如重复提交添加成绩的表单可能会引起数据的重复存储,因此该操作属于典型的敏感操作,系统中设计了一个名为 AddCjAction 的操作,并放在 MVC2 的操作目录 actions 下。相反重复提交有关查询、修改和删除成绩数据的表单时不会引起数据的变化,因此相关的操作可归为非敏感操作,系统中将它们合并在一个名为 OperateCjAction 的操作中,并放在 MVC1 的操作目录 actions1 下。对系统中的论坛帖子、学员数据、教学资源等其他数据的管理也采取类似的方式进行处理。另外,在 MVC2 中设计了一个事件处理辅助操作类 DealEventAction,专用来辅助处理敏感表单的重复提交。

### 3.4 系统运行过程

在由两个 MVC 架构组成的混合型 MVC 框架中,系统的运行过程是:如果在请求的处理过程中不需要进行事件处理,系统的运行过程将在 MVC1 架构中进行。其具体步骤为:浏览器发送请求至服务器,服务器中的控制器服务方法从操作库中获取操作,然后执行它的 perform 方法。操作的 perform 方法返回一个操作路径选择器,为请求确定路由,导向新的 JSP 页面并显示数据至浏览器。如果在请求的处理过程中需要

进行事件处理,系统的运行过程将在 MVC2 架构中进行。其运行步骤在 MVC1 的基础上增加了事件处理,即在业务操作执行前和执行后都要进行事件处理,以确保对操作的有效控制。

## 4 系统性能分析

采用两个 MVC 架构作为系统的基本结构是本系统最突出的特点。研究表明<sup>[3]</sup>,双 MVC 架构能实现结构、功能和速度三者的优化组合,取得较好的整体性能:

(1) 双 MVC 架构是基于 MVC 架构的,保留了原 MVC 架构所具有的优点;

(2) 与仅含 MVC1 的架构比较,双 MVC 架构增加了能处理事件机制的 MVC2,具有更强的功能;

(3) 与仅含一个 MVC2 的架构比较,双 MVC 架构具有更快的处理速度。实验测试表明,初次运行登录操作在双 MVC 架构中(由 MVC1 处理)比在只含一个 MVC2 的架构中其平均处理时间少 55ms;

(4) 双 MVC 架构具有更强的支持并行访问的能力,更少的异常抛出概率和更强的稳定性。

## 5 结束语

本文设计和实现了一种基于双 MVC 架构的远程教学管理系统,为开展基于 Web 的 B/S 模式的远程教学开发了一个实用平台。该系统在功能上为远程教学中的各参与者(学员、教师、校级管理员、站点级管理员)实现了网络化、信息化管理。在技术上改进了经典的 MVC 架构,提高了系统的性能。该系统已在湖南师范大学现代远程教育中投入使用,运行效果良好。同时,此系统的开发对研究 MIS 系统在互联网环境中的应用及其最新开发技术也是一种探索。

### 参考文献

- 1 Simon Brown 等著, JSP 编程指南(第二版)王军等译,北京电子工业出版社,2002. 22~23。
- 2 David M. Geary 著, JSP 高级开发与应用,贺民译,北京科学出版社,2002. 1。
- 3 刘运龙、黄烟波,一种基于双 Servlet 控制器的 MVC 模式的设计与研究,计算机应用,2005, 25(1): 238~240。