

文章编号:1001-9081(2005)12-2964-02

## 基于 J2EE 平台的 MVC 模式的办公日志系统的实现

冯相忠,王 萍

(浙江海洋学院 信息学院,浙江 舟山 316000)

(fxz@zjou.net.cn)

**摘要:**分析了 J2EE 技术和 MVC 模式的体系结构,在软件开发中,采用 MVC 模式,能简化开发过程、提高软件的性能和可维护性。然后阐述了结合这些技术的办公日志系统的具体实现过程。

**关键词:**J2EE;MVC;办公日志系统

**中图分类号:**TP311.52 **文献标识码:**A

### Implementation of the office log system of MVC design pattern based on J2EE platform

FENG Xiang-zhong, WANG Ping

(Department of Information Technology, Zhejiang Ocean University, Zhoushan Zhejiang 316000, China)

**Abstract:** The J2EE technologies and MVC design pattern was analyzed. Using MVC design pattern in the development of software, the process of developing software can be simplified, and the quality and the maintainability of software can be improved. Those techniques were used to implement a system of the Office Log.

**Key words:** J2EE; MVC; office log system

#### 1 基于 J2EE 的 MVC 设计模式

J2EE 是一种利用 Java 平台来简化诸多与多级企业解决方案的开发、部署和管理相关的复杂问题的体系结构。J2EE 提供了一个企业级的计算模型和运行环境,用于开发和部署多层体系结构的应用。它通过提供企业计算环境所必须的各种服务,使得部署在 J2EE 平台上的多层应用可以实现高可用性、安全性、可扩展性和可靠性。它的优点是:1) 计算平台支持 Java 语言,使得基于 J2EE 标准开发的应用可以跨平台地移植;2) J2EE 提供了企业计算中需要的所有服务,且更加易于使用;3) Java 语言具有安全、严格的特点,使开发者编写的代码非常可靠;4) J2EE 中定义的标准接口,例如 JNDI, JDBC, Java Mail 等,因此可以和许多厂商的产品配合,容易得到广泛的支持;5) J2EE 树立了一个广泛而通用的标准,大大简化了应用开发和移植过程。

MVC(Model-View-Controller) 是一个经典的软件体系结构,它采用了“分治”的思想,将表示和数据相互分离,将系统分成三个大的类:即模型(Model)、视图(View)和控制器(Controller)。其中模型是对象的内在属性,它包含所有的业务逻辑,它以 JavaBean 的形式存在;视图是模型的外在表现形式,也是用户所见到的 JSP 页面,一个模型可以对应一个或多个视图;控制器是模型与视图的联系纽带,解释用户的操作意图,并把它交给模型去执行,根据请求和执行结果来决定下一次显示的视图。

J2EE 技术结合 MVC 设计模式在构建企业级 Web 的应用中,JSP 对应视图,因为整个应用系统主要通过 JSP 与外界进行交互;Servlet 对应于控制器,作为 JSP 与 JavaBean 之间的中间枢纽;JavaBean 对应于模型,主要进行数据业务的处理。基于 J2EE 的 MVC 模式的 Web 应用框架如图 1 所示。

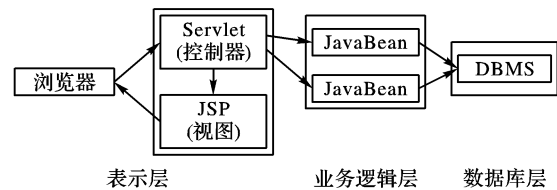


图 1 基于 J2EE 的 MVC 结构

采用图 1 的结构优点是:开发调试比较方便;模式为分层结构,各层之间为松耦合,维护起来方便;可扩展性和伸缩性好。

#### 2 办公日志系统的实现

办公日志系统的开发采用了前面所介绍的基于 J2EE 的 MVC 模式,使用了 J2EE 的 JSP、Servlet、JavaBean 和 JDBC 技术。系统由前台的 JSP 页面和后台的 Java Bean 程序组成。JSP 页面用于显示数据,JavaBean 用于访问数据库及实现复杂业务逻辑和算法。本系统通过 DBADO.java 程序访问数据库,通过 OffLogBean.java 程序实现办公日志系统的各种业务逻辑,如新增公司总结、公司总结查看等。

##### (1) 前台页面显示

前台页面主要用于显示数据,登录页面是一个静态页面 default.html,它由网站标题和登录表单组成。当用户登录成功后进入系统的首页 manage.jsp,根据不同的用户权限,主页面中可实现的功能不同,即页面中的项目不同。而不同用户权限的用户登录所实现的相同功能,由同一 JSP 页面来实现,这样即增加了代码的重用性,也提高了可维护性。当用户以管理员的身份登录时要实现如下的功能和相应的 JSP 页面为:

①查看公司计划总结:total.jsp、searchTotal.jsp 等。

收稿日期:2005-06-30

作者简介:冯相忠(1962-),男,内蒙赤峰人,副教授,硕士,主要研究方向:软件开发、图形图像;王萍(1979-),女,黑龙江齐齐哈尔人,助教,主要研究方向:数据结构、数据处理。

②添加公司计划总结: companyAdd. jsp、companySave. jsp 等。

③选用、编辑和删除公司计划总结: companyEdit. jsp、companyUpdate. jsp、companyDel. jsp、companyState. jsp 等。

④查看、批注员工的个人总结: companyView. jsp、ownReply1. jsp 等。

⑤各部门员工的帐号维护: deptList. jsp、deptNew. jsp、deptUpdate. jsp、userNew. jsp 等。

⑥本人密码修改: passEdit. jsp、passUpdate. jsp 等。

⑦个人办公日志: workTotal. jsp、workAdd. jsp、workSave. jsp 等。

## (2) 数据访问对象 DBADO. java

此 JavaBean 专门用于数据库访问,包括数据库查询、插入、修改、删除等常用的数据库访问操作。首先使用函数 connect() 采用 JDBC 技术建立与数据库的连接,然后调用相关的数据库操作函数,最后调用函数 close(), 关闭数据库连接,释放资源。

其中 connect() 函数代码及含义如下:

```
public boolean connect()
{
    boolean isConnected = false;           //函数返回值
    try
    {
        Class.forName("com.microsoft.jdbc.sqlserver.
            SQLServerDriver").newInstance();
        String url = "jdbc:microsoft:sqlserver://localhost:1433;
            Databasename=offlog";
                                           //offlog 为数据库名
        String user = "sa";                 //用户名
        String password = "sa";            //密码
        conn = DriverManager.getConnection(url, user, password);
                                           //连接数据库

        if (conn != null)
            isConnected = true;
    }
    catch (Exception e)
    {
        System.out.println("connect to database failed");
        e.printStackTrace();
    }
    return isConnected;
}
```

## (3) 业务对象模型 OffLogBean. java

业务对象模型 OffLogBean. java 用于实现办公日志系统的各种业务逻辑,如新增公司总结、公司总结查看等。用户执行的每一个操作,即此 JavaBean 的每一个函数,都要首先对用户权限进行验证,其中添加公司总结 commpanyAdd 函数代码及含义如下:

```
public boolean companyAdd ( HttpServletRequest req,
    HttpServletResponse resp)
{
    boolean isadmin = isAdmin(req, resp);
    //验证是否为管理员或录入员
    if (!isadmin)
        return isadmin;
    //不是管理员或录入员,直接返回 false;
    //是管理员或录入员,进行下面的添加公司的总结操作
    DbADO dbAdo = new DbADO();
    //调用数据访问对象,访问数据库
```

```
Vector allAdmin = null;
allAdmin = new Vector();
int id = 0;
String truename = "";
try
{
    dbAdo.connect();
    String sql = "select * from users where flag = 1";
    //查询所有的管理员
    ResultSet rs = dbAdo.select(sql);
    while (rs.next())
    {
        id = rs.getInt("id");
        truename = rs.getString("truename");
        allAdmin.addElement(new Integer(id));
        allAdmin.addElement(truename);
        System.out.println("admin is: " + id + " " +
            truename);
    }
    //把所有的管理员放在 request 当中,
    //用于前台 JSP 的下拉框显示
    if (allAdmin != null)
        req.setAttribute("allAdmin", allAdmin);
}
catch (Exception e)
{ e.printStackTrace(); }
finally
{ dbAdo.close(); }
return isadmin;
}
```

## (4) 数据库模式的选取

在数据库的选取上,可以选择集中式数据库模式或分布式数据库模式。集中式数据库是集中在一个物理位置的数据库系统,数据库(本地和远程)用户通过网络访问该系统。分布式数据库则是分布于不同的物理位置,但起到单一数据库作用的数据库系统。在电子日志系统中我们采用了集中式数据库模式。在具体选用的数据库上,选用了 MicroSoft 的 SQL Server 2000 数据库,它能够处理大量数据,同时保持数据的完整性并提供许多高级管理功能。它的灵活性、安全性和易用性为数据库编程提供了良好的条件。

## 3 结语

基于 J2EE 的 MVC 设计模式,为构建具有可伸缩性、灵活性、易维护性的企业日志系统提供了良好的机制,建立在 J2EE 基础上电子日志系统,缩短了系统的开发和实施周期,有效维护了系统的复杂性,并使电子日志系统具有良好的可扩展性、集成性,同时减少了电子日志系统的总成本。

### 参考文献:

- [1] 祁耀武,李福太,陈逢春,等. J2EE 平台上 MVC 设计模式在电子政务系统中的应用[J]. 计算机应用研究, 2004, 21(8): 203 - 205.
- [2] 袁冷梅,黄烟波,黄家林,等. J2EE 应用模型中 MVC 软件体系结构的研究与应用[J]. 计算机应用研究, 2003, 20(3): 147 - 149.
- [3] 赛奎春. JSP 工程应用与项目实践[M]. 北京:机械工业出版社, 2005.
- [4] 求是科技. Java 数据库系统开发实例导航[M]. 北京:人民邮电出版社, 2004.