

基于轻量级 J2EE 构架的高校教务管理 系统的设计与实现^①

The design and Implementation of the College's Transaction Administration
Information System Based on Lightweight J2EE Architecture

杨兴春 谯石 董文 李明东 (西华师范大学计算机学院 四川南充 637002)

摘要:J2EE 技术在业界内被一致认为是非常优秀的软件构架,并在许多大型项目中得到了广泛应用,本文分析了基于 J2EE 技术的表现层开源构架 Struts,业务层开源构架 Spring 和数据持久层开源构架 Hibernate 的相关技术特点,并把这种基于 Struts + Spring + Hibernate 的轻量级 J2EE 构架,在高校教务管理系统的开发中得到了实现。

关键词:J2EE Struts Spring Hibernate

1 前言

目前,J2EE 技术的模块化、层次性、平台无关性等诸多优点,已被业界所认同,是电子商务和电子政务的首选解决方案。传统的 J2EE 表现层开发一般采用 Jsp/Servlet 的开发模式,造成显示、业务和数据的高度藕合,业务逻辑层集成通常采用 EJB 组件实现,但 EJB 是一种重量级分布式应用组件,其过分依赖容器,内聚性强,实体 Bean 性能较差,测试和维护困难,且 EJB 容器价格昂贵。基于以上因素,在实际开发与应用中,需要设计一种轻量级的 J2EE 构架,以避免传统的重量级 J2EE 构架的诸多不足。

2 Struts、Spring、Hibernate 简介

J2EE^[1]是由 Sun 公司制定的一个开放式企业级应用规范,它包括层次性的分布式应用模型和一系列的开发技术标准。J2EE 根据功能将应用逻辑分成多个层次,通常包括用户接口层、表示层、业务层和数据持久层,各层支持相应的服务器和组件,组件运行在分布式服务器的容器中,容器为其提供运行环境,负责组件的装卸、事务管理、数据存储和访问控制等。

Struts^[2]是由 Apache 发起的基于 MVC 应用的开源框架,它利用 JavaServlet 和 Jsp 技术构建 Web 应用。MVC 应用分三部分:模型(Model)、视图(View)、控制器(Controller),三部分各自处理自己的任务:视图是人机界面,模型表示业务数据和业务逻辑,控制器接受用户输入并调用相应的模型和视图完成用户的需求^[6]。MVC 开发模式体现了系统的层次性、结构性和功能性,大大减低了系统各部分的藕合度,易于项目开发和维护升级,能充分满足 Web 开发需要。

传统的中间件解决方案 EJB 属重量级组件,其高侵入性与高依赖性使得基于 EJB 的应用受到很大限制。而 2003 年出现的以 J2EE 实用主义为设计理念的 Spring^[3]开源框架,则有效解决了 J2EE 开发中的诸多问题。该框架是一个基于 POJO(普通 java 对象),IOC(反转模式)和 AOP(面向切面编程)技术的轻量级应用框架,由 IOC 容器,一个配置和组织组件的框架,一组内置的事务管理、数据持久化等服务组成。IOC 容器提供无侵入的高扩展性构架,能很方便地将业务对象纳入其中进行管理。Spring 集成了对 Hibernate 的支持,并且基于 Java 语言的 AOP 框架,给 POJO 提供

① 项目编号:2004A099 四川省教育厅重点项目《数据挖掘技术在高等学校决策中的应用研究》

声明式的事务管理,使得事务管理相对简单轻松、简便。

Hibernate^[4]是 Java 对象持久层轻量级封装框架。持久层是应用程序中实现持久数据管理的结构层。目前的对象数据库技术 (ODBMS) 尚未成熟,在面向对象的开发中,通常采用 O/R Mapping(对象/关系映射)技术将关系数据库与对象进行映射,包装关系数据库数据为对象,并提供相应的 API。Hibernate 是 O/R Mapping 技术的 Java 实现,是较彻底的对象/关系映射工具。

3 Struts + Spring + Hibernate 的整合^{[5][7][8]}

综上所述,表现层采用 Struts 框架,业务逻辑层用 Spring 框架取代 EJB 容器,数据持久层以 Hibernate 取代实体 Bean,提供具体的数据库数据处理操作,三者进行有机整合,构成轻量级 J2EE 应用框架。

整个轻量级框架业务流程为:用户接口层利用 JSP + HTML 页面实现交互界面,负责传送页面请求和接收响应,表示层 Struts 收到请求,调用相应的 Action,处于业务层的 Spring IoC 容器负责向 Action 提供业务服务组件 (Service) 和相应的数据访问处理组件 (DAO),并提供事务处理、日志管理等一系列系统服务,处于持久层的 Hibernate 负责对象化映射与数据库交互,具体处理 DAO 组件请求,并返回结果。其整合如图 1 所示。

基于该系统的数据挖掘做准备,是上述基于 Struts + Spring + Hibernate 框架的轻量级 J2EE 实现。

该系统包含教务管理、科研管理、图书资料管理、实验室资源管理、办公自动化、办公用品管理多个子系统,面向学校各部门以及校内外各层次用户。系统 Web 服务器采用 Tomcat5.0,数据库选用 Oracle 9i。根据上述的轻量级 J2EE 构架分层实现本系统,为简便起见,仅介绍用户管理模块的轻量级 J2EE 构架实现,该模块的类图图 2 所示。

该模块分四层:

(1) 用户接口层。仅显示用户登录界面,发送 Struts 请求和接受响应完成交互任务,实现显示和业务逻辑的解耦。本层中,以用户登录为例,Login.jsp 为登录界面,并提供了 Action 请求的入口,登录结果将在 LoginResult.jsp 中显示。

(2) 表示层。配置并部署 Struts 框架,处理业务流程,实现请求 (* .do) 到 Action 的映射,所有的业务逻辑封装在 Action 中,由 Action 调用业务逻辑的服务组件,根据处理结果跳转到 Forward 对象指定的响应。业务流程的部署由 struts - config.xml 文件完成。该文件配置了请求 (LoginAction.do) 对应的 Action 对象 (LoginAction),并设置响应跳转,struts - config.xml 部分配置如下:

.....

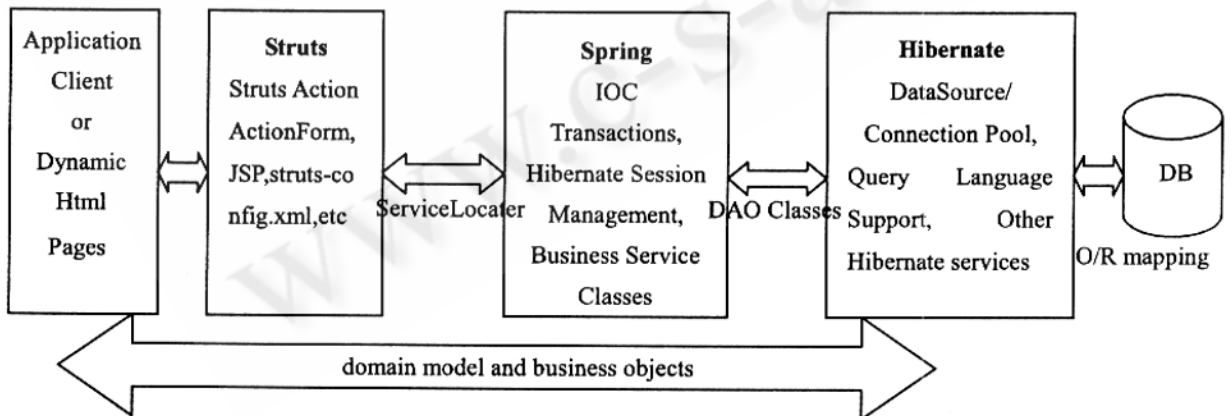


图 1 Struts + Spring + Hibernate 构架图

4 教务系统结构设计

本系统主要完成教务信息的日常处理以及为以后

```
< action path = " LoginAction " type = ? " org.
cwnu. struts. action. LoginAction " >
< forward name = ? " failed " path = " error. jsp "
>
```

```
<forward name = "?" success" path = "LoginRe-
sult.jsp" >
.....
</action >
.....
```

```
.....
<bean id = " transactionManager"
class = " org. springframework. orm. hibernate.
HibernateTransactionManager" >
<property name = " sessionFactory" > <ref lo-
```

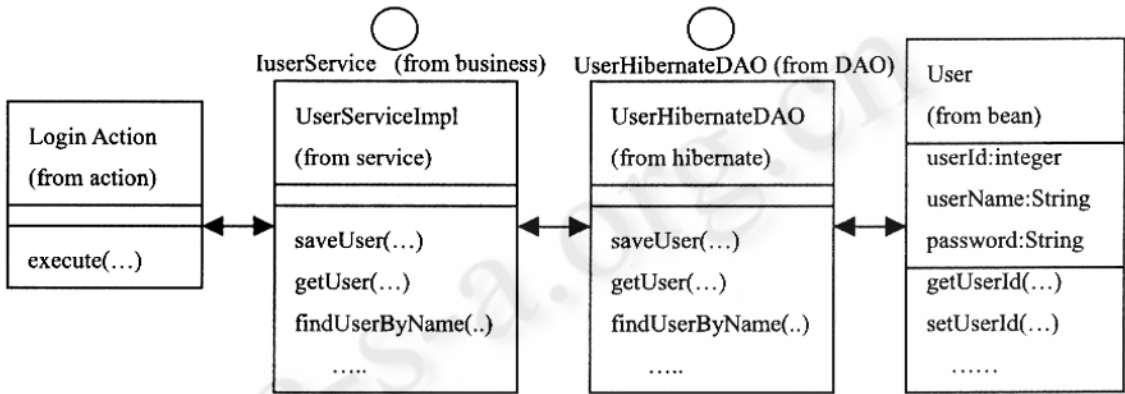


图 2 用户管理模块类图

Action 则根据传来的参数,从 Spring 容器中通过相应的服务 (Service) 接口,调用相应的服务,部分代码如下:

```
public ActionForward execute ( ActionMapping
mapping, ActionForm form,
HttpServletRequest request, HttpServletResponse
response) {
LoginForm loginForm = (LoginForm) form;
String loginUser = loginForm.getUser();
String loginPassword = loginForm.getPassword
();
List userList = getUserService ( ). findUser-
ByName(loginUser);
Iterator it = userList.iterator();
User user = (User) it.next();
.....
```

(3) 业务层。本层基于 Spring 框架,用声明配置的方式(applicationContext.xml)对业务的实现进行封装整合,充分利用 spring 框架的 IOC 与 AOP,配置事务处理,日志处理等,让 Spring 容器来管理此类服务,减轻编程负担。配置文件 applicationContext.xml 的部分配置如下:

```
cal = " sessionFactory" / > </property >
</bean >
<bean id = " userTarget" class = " org. cwnu. cs.
service. userServiceImpl" >
<property name = " userDao" > <ref local = "
userDAO" / > </property >
</bean >
<bean id = " userDao" class = " org. cwnu. cs. hi-
berate. userHibernateDAO" >
<property name = " sessionFactory" > <ref lo-
cal = " sessionFactory" / > </property >
</bean >
<bean id = " daoProxy" class = " org. spring-
framework. transaction. interceptor. TransactionProxy-
FactoryBean" >
<property name = " transactionManager" > <
ref bean = " transactionManager" / > </property >
<property name = " userTarget " > <ref local
= " userTarget" / > </property >
<property name = " transactionAttributes" >
<props >
<prop key = " find * " > PROPAGATION_
```

```

SUPPORTS </prop >
    < prop key = " * " > PROPAGATION_RE-
REQUIRED </prop >
    </props >
</property >
</bean >
.....

```

(4) 数据持久层。本层采用可视化的开发工具 PowerDesigner 设计数据库,然后通过 Hibernate 官方提供的 MiddleGen for Hibernate 和 Hibernate_Extension 工具包,很方便的根据相应的数据库,导出数据库表结构,生成对象和关系的映射(O/R Mapping)及相应的 POJO。扩展 Spring 提供的抽象类 HibernateDaoSupport(该类实现了 HibernateTemplate 与 SessionFactory 实例的关联),这样就可以实现具体的 DAO 操作,以便最终与数据库交互。部分代码如下:

```

import org. springframework. orm. hibernate. support.
HibernateDaoSupport;
.....
public class UserHibernateDAO extends HibernateDao-
Support implements IUserDAO{
    public User saveUser( User user ) {
        try{ this. getHibernateTemplate( ). save( user );
            return user;
        } catch( Exception ex ) {
            ex. printStackTrace( );
            return null;
        }
    }
}
.....

```

5 结束语

在开发大型信息管理系统中,采用 Struts + Spring + Hibernate 的轻量级 J2EE 构架,大大降低了各模块之间的耦合度,提升了软件重用度,又能让开发者充分使用面向对象开发技术,与重量级 J2EE 构架相比,提高了系统的运行性能和系统开发效率以及系统的可维护性。

参考文献

- 1 J2EE 官方网站, Sun Developer Network [EB/OL]. <http://java.sun.com/j2ee/blueprints/index.html>.
- 2 Struts 官方网站, The Apache Software Foundation [EB/OL], <http://struts.apache.org/>.
- 3 Spring 官方网站. Spring Framework [EB/OL]. <http://www.springframework.org>.
- 4 Hibernate 官方网站. Relational Persistence for Java and .NET [EB/OL]. <http://www.hibernate.org>.
- 5 魏海萍等译, Expert one-on-one J2EE Design and Development [M], 电子工业出版社, 2003 年.
- 6 孙卫琴, 精通 Struts: 基于 MVC 的 Java Web 设计与开发 [M], 电子工业出版社, 2004 年.
- 7 席晓峰、吕良双、遼鹏, 使用 J2EE 框架技术构建可重用的 Web 应用 [J], 计算机工程与应用, 2005 (29): 208-214.
- 8 王卫平、王松涛、王名茗, 一种基于 J2EE、Spring 和 hibernate 的轻量级 EAI 构架 [J], 计算机系统应用, 2005 (11): 38-41.