

SSM 框架在 Web 应用开发中的设计与实现

李 洋

(南京邮电大学 通信与信息工程学院,江苏 南京 210000)

摘 要:在 Web 应用的开发过程中,开发框架的选择非常重要。一个好的开发框架能够加速 Web 应用的开发速度,降低开发成本,减少开发人员的工作量,同时能够使 Web 应用具有良好的扩展性和移植性。基于 SpringMVC+Spring+Mybatis (SSM)的框架凭借良好的性能和较快的开发效率,逐渐成为主流的 Web 应用开发框架组合。SpringMVC 是一个基于 MVC 的框架,其主要负责表现层的功能,比如响应请求。Spring 框架主要起到容器的功能,整合了 SpringMVC 和 Mybatis,实现层与层之间的解耦,同时使业务逻辑更加清晰。Mybatis 框架主要负责的是数据持久层,完成和数据库的相关操作。重点以权限管理模块为例,说明了 SSM 框架在开发后台管理系统中的应用,为开发后台管理系统提供了理论指导。

关键词:SpringMVC;Spring;Mybatis;Web 应用;框架

中图分类号:TP39

文献标识码:A

文章编号:1673-629X(2016)12-0190-05

doi:10.3969/j.issn.1673-629X.2016.12.041

Design and Implementation of SSM in Web Application Development

LI Yang

(College of Telecommunications and Information Engineering,Nanjing University of Posts and
Telecommunications,Nanjing 210000,China)

Abstract:In the process of Web application development,the choice of development framework is essential. A good development framework can accelerate the development of Web application,reduce development costs,ease the burden of developer and make the Web application has good expansibility and portability. Framework based on SpringMVC, Spring and Mybatis with better performance and faster development productivity has gradually become the preferred combination of Web application development framework. SpringMVC is a MVC framework,which is in charge of the function of presentation layer like responding to the request. Spring framework plays the part of container,which integrates Mybatis and SpringMVC,and implements the decoupling between layers. Meanwhile,it can make business logic becomes more clear. Mybatis framework is mainly responsible for data persistence layer and complete the operation of database. Taking the rights management module as an example,in order to explain the application of SSM framework in the development of back-stage management system,it provides theoretical guidance for the development of back-stage management system.

Key words:SpringMVC;Spring;Mybatis;Web application;framework

0 引言

目前,国内外有很多的 Web 开发技术,其中应用最广泛的两类技术分别是微软(Microsoft)公司的 .Net 平台和 Sun 及 IBM 等公司提出的 J2EE 平台。J2EE 主要面向使用 Java 程序设计语言进行基于 Web 的企业应用的开发部署^[1]。J2EE 平台相较于 .Net 平台是开放的、跨平台的、可移植的,因此很多企业都选择 J2EE 平台的结构设计和解决方案。传统的 J2EE 以 EJB(Enterprise JavaBean)容器为核心,EJB 组件提供事务管理,处理用户请求,支持打包处理和资源管理等

功能,但是 EJB 与 EJB 容器的高度耦合关系,使其可移植性差^[2],同时 EJB 代码的复杂、维护的困难使其逐步被淘汰,因此轻量级的框架就应运而生。例如,Struts、Spring 等,降低了开发的复杂度,提高了开发时的调试效率,进而提高了 Web 网站的开发速度。

1 MVC 设计模式

MVC(Model-View-Controller,即模型-视图-控制器^[3])是 Web 开发中的一种设计模式。图 1 显示了这三种组件的联系和功能的分工。

收稿日期:2016-03-04

修回日期:2016-06-15

网络出版时间:2016-11-22

基金项目:国家自然科学基金资助项目(61271234)

作者简介:李 洋(1991-),男,硕士研究生,研究方向为卫星通信技术。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20161122.1227.014.html>

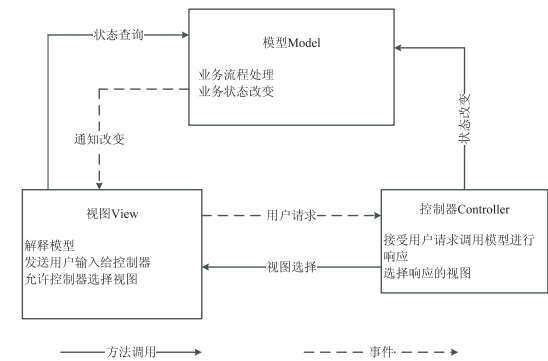


图 1 MVC 组件类型的关系与功能

在这种设计模式中,模型 (Model)、视图 (View)、控制器 (Controller) 分别履行不同的职责。Model 主要实现业务的逻辑处理,业务规则的制定,包括数据库的访问。因此业务模型的设计是 MVC 设计模式中最重要的部分。View 是根据模型数据创建的,用户能看到并且能与之交互。View 通常是采集用户数据的输入,响应用户的请求。Controller 通过读取视图中用户输入的数据,根据用户输入的数据选择合适的模型处理用户请求,并且选择合适的视图来展示模型处理结果。Controller 并不参与数据的处理,而是将数据的处理交给 Model 执行。可见,MVC 设计模式实现的业务处理和视图展示的分,降低了每个模块之间的耦合度,达到了系统的设计目标和整体要求。

2 SpringMVC 框架

SpringMVC 框架是 Spring 基于 MVC 设计模型的用于构建 Web 应用程序的一种实现。在 Web 应用开发过程中, SpringMVC 框架容纳了包括 JSP 在内的大量视图技术,可以灵活配置^[4]。虽然 SpringMVC 与 Struts 功能类似,但是它们的实现原理不同。Struts2 是基于类别的拦截,一个类对应一个 request 上下文。SpringMVC 是基于方法级别的拦截,一个方法对应一个 request 上下文,而方法同时又跟一个 URL 对应。SpringMVC 的工作流程如图 2 所示。

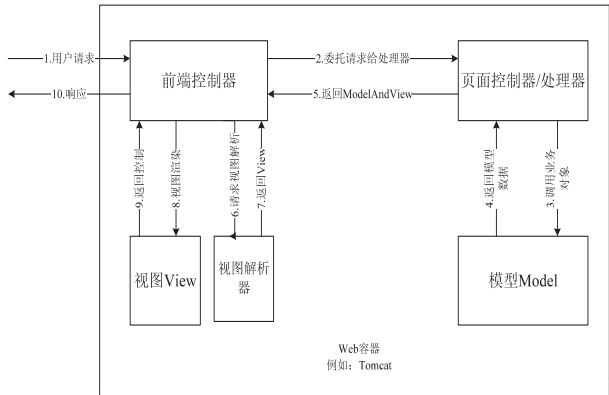


图 2 SpringMVC 工作流程图

SpringMVC 框架的主要构件是前端控制器 (Dis-

patcherServlet)、处理器映射器 (HandlerMapping)、处理器适配器 (HandlerAdapter)、处理器 (Handler)、视图解析器 (ViewResolver)、视图^[5]。其中, DispatcherServlet 从 HttpServlet 继承而来,作用类似于一个 Servlet,主要用于接收用户请求和响应服务,功能如下:

- (1)实现文件上传,本地化解析功能;
- (2)通过处理器映射器,将请求映射到处理器 (返回一个执行链,包括一个处理器、多个处理器拦截器);
- (3)通过处理器映射器实现多种类型的处理器;
- (4)通过视图解析器实现逻辑视图到具体视图的解析;
- (5)实现对具体视图的渲染;
- (6)通过异常处理解析器解析执行过程中遇到的异常。

3 Spring 框架

Spring 是为降低企业的应用开发复杂度而创建的一种开源框架。其主要目的就是实现开发中各层之间的分离,实现允许每层灵活选择何种构件,并且为 J2EE 的开发提供集成框架。Spring 为企业应用的开发提供了一种轻量级的解决方案^[6]。Spring 框架主要提供了面向切面 (AOP) 编程,核心的控制反转 (IoC) 机制,以及能够整合多种持久层的框架和拥有自己的 MVC 框架。Spring 框架希望提供不是专注于某一层的解决方案,而是提供 Java EE 应用开发中各层的解决方案。从图 3 中可以看出, Spring 框架实现了表现层、业务层、持久层的整合,但这并不意味着原来已有的框架能被 Spring 框架取代,而是 Spring 框架能够实现已有框架的无缝整合,为企业级应用开发提供更大的灵活性。

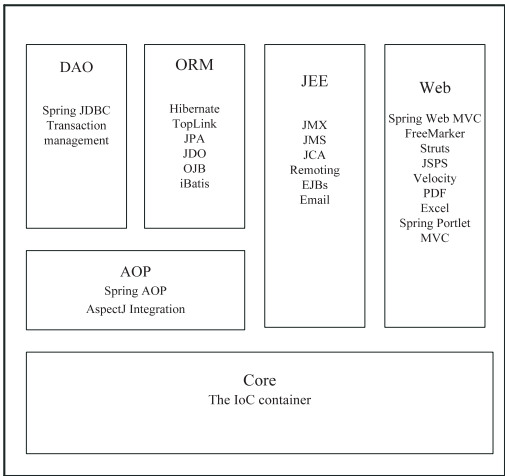


图 3 Spring 框架结构图

总结起来, Spring 具有的优势为:

- (1)提供了一种可有效地组织中间层对象方法。

- (2)提供了一种非侵入设计,降低了代码污染。
 - (3)Spring 通过依赖注入机制降低了业务对象之间的复杂性,实现了对象、组件之间的解耦。
 - (4)通过 AOP 实现日志、事务等功能的集中式管理。
 - (5) Spring 的 ORM 和 DAO (Data Access Objects)^[7]实现了整合多种持久层框架的可能性,简化了开发人员对数据库的操作。
 - (6)Spring 作为整合组件,可以整合其他框架,为企业级应用的开发提供更大的灵活度。
- 相关术语如下所述:

IoC (Inversion of Control,控制反转):面向对象开发中,这是一种设计原则,用来降低代码之间的关联性。IoC 将设计好的类不是在类的内部实现控制,而是由系统去控制,这称为控制反转^[8]。

AOP (Aspect - Oriented Programming,面向切面编程):是由 Gregor Kiczales 提出的一种全新的编程技术,为开发者提供了一种描述横切关注点的机制^[9]。AOP 通过动态代理,实现将多个系统级服务(例如日志、事务等)的行为封装到可重用的模块中,用来实现业务逻辑和系统级服务的松耦合。

非侵入式:侵入式设计表现为客户端程序需要继承框架中的类,而非侵入式设计则表现为客户端实现框架提供的接口,这样就减少了应用程序对框架的依赖,以及框架移植时的负担,提高了应用程序组件的复用性。

4 Mybatis 框架

Mybatis 是 J2EE 应用开发中的一个持久层框架,它通过配置的 XML 文件或者注解实现对象和存储过程,SQL 语句的关联,而不是实现对象与数据库表的关联,并且可以实现动态生成 SQL 语句。MyBatis 框架集合多种操作型关系数据的概念和方法,它是一个强大的数据访问工具和解决的方法^[10]。Mybatis 与 JDBC 相比,简化了相关代码的实现,其通过使用注解方法或者简单的 XML 配置文件实现原始映射,然后通过映射引擎将 JavaBean 或者定义的接口映射成数据库中的记录。Mybatis 的工作原理如图 4 所示。通过 XML 配置文件将 JavaBean、Map 以及本数据类型(如 Integer、String 等)映射成 PreparedStatement 所需的输入参数以 ResultSet 结果集的类型。

Mybatis 让程序将主要精力放在 SQL 语句上,通过 Mybatis 提供的映射方式,自由灵活生成(半自动化,大部分需要程序员编写 SQL)满足需要的 SQL 语句,修改比较容易。Mybatis 可以将 PreparedStatement 中的输入参数自动进行输入映射,将查询结果集灵活映射成

Java 对象。相对于其他的 ORM 框架(如 Hibernate),Mybatis 适用于需求变化多的项目(如互联网项目)。

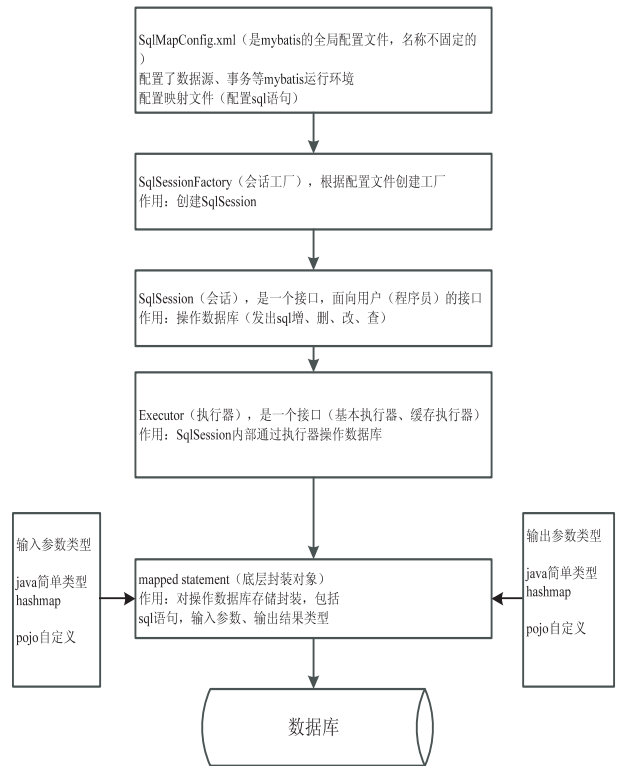


图 4 Mybatis 的工作原理图

5 基于 SSM 的后台管理系统实现

5.1 整合 SSM 框

SSM 是 SpringMVC、Spring、Mybatis 的简写。SSM 是轻量级并且应用较多的一种组合框架,好多中小企业的门户网站、政府网站、金融项目都是基于 SSM 架构开发的。由于 Spring 是一个业务层的框架,能够很好地整合表现层的 SpringMVC 框架和持久层的 Mybatis 框架。整合以后, Spring 就会充当整个应用的工厂。首先整合 Spring 和 Mybatis, Spring 整合 Mybatis 最简单的理解就是“把 Mybatis 数据源的配置、事务的管理、SqlSessionFactory 的创建以及数据映射器接口 Mapper 的创建交由 Spring 管理”,所以 Mybatis 的配置文件 mybatis-config.xml 中不需要再配置数据源及事务,在业务层 service 实现时不需要手动地获取 SqlSession 以及对应的数据映射器接口 Mapper,通过 Spring 的注入即可。Spring 整合 Mybatis 的配置文件部分代码如下:

```
<!--引入配置文件-->
<bean id="propertyConfigurer"
class="org.springframework.beans.factory.config.Property-
PlaceholderConfigurer">
<property name="location" value="classpath:jdbc.properties" />
</bean>
<!--Spring 和 MyBatis 整合-->
```

```
<bean id="sqlSessionFactory" class="org.mybatis.spring.
SqlSessionFactoryBean">
<property name="dataSource" ref="dataSource" />
<property name="mapperLocations" value="classpath:com/
cn/iLake/mapping/*.xml"></property>
</bean>
```

第二步整合 SpringMVC 和 Spring。由前述可知 SpringMVC 框架是 Spring 的基于 MVC 设计模型的实现,因此单独配置一个 SpringMVC 的配置文件,其主要完成自动扫描控制器、视图模式、注解的启动等功能的配置,然后在 web.xml 文件中配置整合即可。SpringMVC 的配置文件部分代码如下:

```
<mvc:annotation-driven />
<!--自动扫描该包-->
<context:component-scan base-package="com.cn.iLake" />
<!--配置视图解析器-->
<bean id="viewResolver"
class="org.springframework.web.servlet.view.freemarker.
FreeMarkerViewResolver">
</bean>
<!--静态资源访问-->
<mvc:resources mapping="/resources/*" location="/re-
sources/" />
```

5.2 系统模型

经过网站需求的分析,网站的后台管理功能设计结构如图 5 所示。

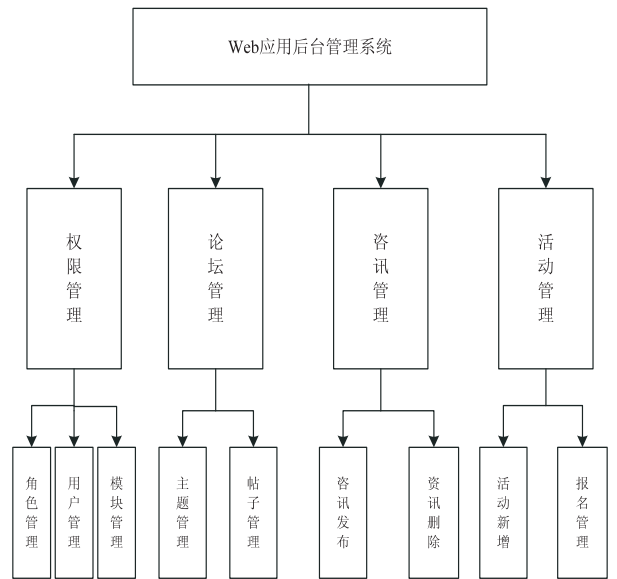


图 5 后台管理功能结构图

5.3 体系架构

J2EE 规范了服务器端应用系统开发的框架,其核心思想就是组件与分层。其分层主要包括三个方面:表示层、业务逻辑层、数据层。在实际项目开发过程中,由于项目的需求与设计,通常需要扩展三层划分的体系结构,一般是拆分原来的三层体系划分变为四层

体系划分,也就是在业务逻辑层与数据层之间增加一个服务层。其用于持久层的数据操作,表示数据库的持久内存存储对象^[11]。

开发中四层体系结构如图 6 所示。

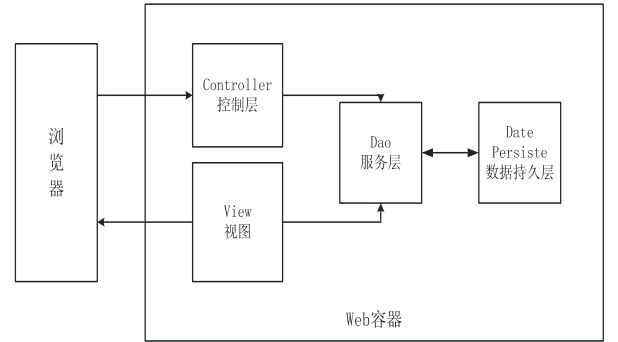


图 6 J2EE 四层开发架构图

5.4 系统实现

在后台管理系统结构图中,网站后台管理总共有四种功能:权限管理、知识管理、资讯管理、活动管理。所谓的管理就是实现数据库的增删改查,因为知识、活动、资讯管理都是以权限管理为基础的,因此下面主要讲述权限管理功能的实现。

5.4.1 数据持久层

MyBatis 应用程序首先创建 SqlSessionFactory 实例,其作用相当于一个数据库连接池,然后 SqlSessionFactory 通过配置的 XML 文件来获取一个 SqlSession。执行 SQL 语句所必须的方法,都包含在 SqlSession 中,所以通过 SqlSession 完成对数据库的操作,用完之后需要释放数据库连接。

5.4.2 服务层

数据访问采用 DAO (数据访问对象) 模式^[12],DAO 是数据库访问接口,可以从任何一种配置好的数据源处获取数据^[13]。服务层 (DAO) 的使用可以减少业务逻辑代码和数据库访问代码之间的耦合度,有利于划分和分配任务。这里用权限管理里面的角色管理举例,首先定义接口类 RoleDao,其定义如下:

```
public interface RoleDao {
//根据 id 查询角色信息
public Role findRoleById(int id) throws Exception;
//修改角色
public void modifyRole(int roleId) throws Exception;
//添加角色
public void insertRole(Role role) throws Exception;
//删除角色
public void deleteRole(int id) throws Exception;
```

然后定义其接口实现类 RoleDaoImpl,在接口实现类中实现的 RoleDao 接口中声明的方法。根据 id 查询角色信息的方法实现如下:

```
public Role findRoleById(int id) throws Exception {
```



```
SqlSession sqlSession=sqlSessionFactory.openSession();
Role role=sqlSession.selectOne("test.findRoleById", id);
sqlSession.close();
return role;
}
```

最后在 Mybatis 的 mapper 配置文件中添加如下配置即可。

```
<mapper namespace="test">
<select id="findRoleById" parameterType="int" resultType
="com.cn.iLake.dmo.Role">
SELECT * FROM ROLE WHERE id=#{value}
</select>
</mapper>
```

5.4.3 控制层

在增删改一个用户信息时,需要一个 Form 表单来临时存储用户信息表中各个字段的信息,因此需要一个表单类 RoleInfoForm。它其实就是一个简单的 JavaBean,并且每个字段都定义了 Getter() 和 Setter() 方法,然后直接就可以在 Controller 的函数参数列表中直接写这个 JavaBean 对象, Spring 会自动进行封装。Controller 中代码如下:

```
@RequestMapping(value="/roleInfo", method = RequestMethod.POST)

public String queryRole(RoleForm role, HttpSession session,
Model model)

throws UnsupportedEncodingException,
Exception {
}
```

已经在 web.xml 的文件中配置了请求地址, DispatcherServlet 会根据请求地址把该次请求分派到所写的 Controller 类中的方法上。通过注解 @Controller 注册所写的 Controller 类, @RequestMapping 注册请求的路径。至此,控制层的功能已经实现。

5.4.4 表示层

通过 DispatcherServlet 可以将页面的链接地址、提交的表单请求转发到注册过的 Controller 类中的方法上,实现整个系统页面之间的跳转。由于在 SpringMVC 的配置文件中已经配置好了视图解析器,其解析的文件是 ftl 文件,所以需要开发的是 ftl 文件,而不是

JSP 文件。

6 结束语

文中结合 SpringMVC、Spring、Mybatis 框架的各自功能和特点,很好地体现了 MVC 分层开发的设计理念,因而有利于应用程序的维护和功能扩展,降低每个模块之间的耦合度,达到了系统的设计目标和整体要求。

参考文献:

- [1] 冯 博,朱穗晖. 程序设计教程[M]. 北京:清华大学出版社,2005.
- [2] Armstrong E, Ball S, Bodoff S, et al. The J2EE 1.4 tutorial [M]. [s.l.]:Sun Micro Systems Inc.,2005:2-25.
- [3] 郭继成. Struts 与 Hibernate 实用教程[M]. 北京:电子工业出版社,2006.
- [4] Andrew H. Object-oriented analysis and design[M]. Beijing: China Machine Press,2003:17-56.
- [5] Linaje M, Preciado J, Sanchez F F. Engineering rich internet application user interfaces over web-models[J]. IEEE Internet Computing,2007,11(6):53-59.
- [6] 李 刚. 轻量级 JavaEE 企业应用实战[M]. 第4版. 北京:电子工业出版社,2015.
- [7] 黎永良,崔杜武. MVC 设计模式的改进与应用[J]. 计算机工程,2005,31(9):96-97.
- [8] 魏学松,张育平. IOC 框架的研究与设计[J]. 计算机技术与发展,2006,16(3):213-216.
- [9] Elrad T, Aksit M, Kiczales G, et al. Discussing aspects of AOP [J]. Communications of the ACM,2001,44(10):33-38.
- [10] 张 宇,王映辉,张翔南. 基于 Spring 的 MVC 框架设计与实现[J]. 计算机工程,2010,36(4):59-62.
- [11] 刘中兵. 开发者突击:Java Web 主流框架整合开发:J2EE+ Struts+ Hibernate+ Spring[M]. 北京:电子工业出版社,2008.
- [12] 贾 昆,甘仞初,高慧颖. 数据访问对象模式在企业应用集成中的应用[J]. 计算机工程与设计,2006,27(3):373-375.
- [13] 王 坤. 基于 J2EE 平台 Spring MVC 框架开发的 MIS 系统设计与实现[D]. 上海:华东师范大学,2008.