快速的开源J2EE应用程序开发中的Spring Web MVC框架：案例学习

摘要

如今，开发Web应用程序是非常有挑战性的，我们需要准确的，经济的，有效率的开发应用程序

我们对增加生产率而且减少系统复杂度非常关注。这一直就是在一场改变程序员在开发Java企业版应用程序的运动中的一个潜在的主题。我们关注的焦点是怎样在没有EJB（企业版Java Bean）的情况下开发出还算满意的J2EE应用来。一个最好的选择就是Spring框架，它提供更少的服务但是比EJB有更小的侵入性。在此基础上，紧接着就是在web应用软件的开发和实现领域更高效和更少复杂的需求。在这篇文章当中，我们简短的描述一下Spring的体系架构，并展示一个使用Spring MVC框架的学习例子

关键字：MVC，Spring，XML

1介绍

如今，Web是一个非常复杂的问题。随着公司和企业的期望在不断增长，Web应用程序越来越复杂，性能越来越重要。复杂度随着通信设备的增多而增加。业务驱使应用程序要使用网络和其他各种通信设备。随着网络上的数据负载不断增加，我们不得不关注体系结构。我们来讨论一下，在快速的应用程序开发过程中如何在保持应用的模型视图体系结构的情况下使用spring web mvc框架快速的工作。

Spring框架有一系列的特性，我们来简短的讨论一下这些特性。

1 控制反转：控制反转或者说IoC是一种用来接通在一个应用程序中的各种服务和组件的技术。IoC是一个软件的设计模式和各种编程技术的集合，在这种模式下，系统的控制流被反转了，这和传统的软件中的模块的交互模式是有区别的。在IoC作用下，不是应用程序调用框架，而是框架调用程序指定的组件。IoC也可以解释为”被依赖的子资源或者说依赖在运行期间被注入到依赖方“，这也称为依赖注入。org.springframework.beans.factory.BeanFactory实际上可以代表Spring的IoC容器，它负责维护和管理beans。BeanFactory接口是Spring框架中的IoC容器中的核心接口。一个bean就是一个被Spring IoC容器实例化和管理的简单的对象。这些bean和bean之间的依赖用被容器使用的配置元数据来表示。

2

构造函数依赖注入：我们可以使用java的构造函数来转载bean的值。一个java类的定义中有单个属性的构造函数。Details.xml文件提供传给构造函数的值。现在，另一个java对象使用BeanFactory的方法来载入这个xml文件。使用xml文件和java文件中的构造函数来载入值。这被构造函数用来传值。

3

Setter依赖注入：我们定义的每一个bean都有getter和setter方法。我们也可以用set方法来设置bean中的属性值。setter方法重写了重bean中初始载入的值。

4

接口：我们可以在spring中定义一个接口类。为了做到这一点，我们要在java程序中引入接口。这样我们就可以使用接口中定义的方法来使用spring和xml。

5

继承：在java编程中，一个java类可以获得另一个java类中的属性。有三种形式。1.抽象：被声明为抽象的bean可以在spring中被继承 2.父子：我们可以定义像父子那样的层次关系。3.父-子-孙 关系：这样我们可以定义3个或更多类的层次关系。

6

自动装配：自动装配被用来映射xml文件和java文件中的属性和值。有四种形式来集成它。根据名字，根据类型，构造函数，自动检测byName是默认的定义。

7

Bean的作用域：所有spring中定义的bean都有四种范围，分别是原型，会话，请求，单例，全局会话。这被用来控制bean的访问。

8

引用bean：在xml文件中定义的的bean可以作为值被分配到另一个bean中。这被用来从一个bean中读值和被分配到另一个bean中。

二 spring的主要组件

在spring中我们也遵守MVC原则。spring被设计用在桌面和网络应用中。Spring由三个核心的基础组件组成。1.控制器：处理导航逻辑，并且在业务逻辑上与服务层交互。2.模型：在控制器和包含渲染视图所需数据的视图间的契约。3.视图：从模型中获取数据来渲染对请求的响应。Spring MVC中的核心组件如下

1.DispatcherServlet:这是spring的前端控制器。Web.xml接收请求并且把请求交给DispatcherServlet处理，这是与请求进行交互的第一个控制器。它也是Servlet的实现，它控制应用程序的整个流程并且引导应用程序的流程。

2.Controller：这是用户创建的用来处理请求的组件。它封装了导航逻辑。控制器把服务委托给服务对象。

3.View：视图负责渲染输出。根据结果和视图设备，通信设备，对于不同类型的输出可以展示出不同的视图。

4. ModelAndView:ModelAndView是spring框架的核心部分。它实现应用程序的业务逻辑它由控制器创建。他把请求和视图联系起来。它存储业务逻辑和模型数据。控制器会调用它，它再执行。执行时，它会返回数据和视图的名称

5.ViewResolver:视图怎样被展示依赖于从ModelAndView接收到的结果。它被用来映射逻辑视图名和实际的视图实现。这部分实现输出媒介是什么，怎样展示。

6.HandlerMapping：被DispatcherServlet用来映射收到的请求和单个控制器的策略接口。它需要识别请求，调用相应的处理器来提供服务。处理器会调用控制器。

三。spring体系结构

Spring框架提供一个完全符合MVC的模块来构建web应用。spirng提供插入式的体系结构它可以被配置来使用多种视图技术比如JSP，Velocity，Tiles，iText等。Spring MVC分隔controller, model object, dispatcher Servlet 和 handler object.对象和控制器的清晰分割可以使他们更容易的定制。图一展示了执行流程。

下图02显示了Spring模型的顺序图。 调度器Servlet是应用程序的入口。。 Dispatch Servlet一收到服务的请求，它将决定handler。

所有处理程序都会对Servlet进行映射。 处理者将会起作用，并将调用相应的控制器并

将请求参数传递给它。

现在控制器开始起作用，它包含业务逻辑和一个与控制器相关联的ModelAndView。执行后它将返回ModelAndView给Dispatch Servlet。 此ModelAndView包含数据和视图

名称。

Dispatcher Servlet从控制器获取ModelAndView。 它包含数据和视图名称。 Servlet会调用

视图解析器。 视图解析器将识别要显示数据的视图的名称。 最后它

将把数据以适当的格式呈现给用户。

四 spring和xml

Xml广泛应用于spring框架。 它简化了开发过程，节省了时间。 xml用于

存储在执行应用程序期间使用的数据。

web.xml是应用程序中的入口点。 它会告诉你进一步的导航路径。 它加载应用程序

上下文类并声明调度器Servlet的xml文件的名称。

web.xml:- <?xml version="1.0" encoding="UTF-8"?>

<web-app version="1.0" xmlns="http//java.sun.com/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemalLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app-2\_5.xsd">

<listener><listener-Class>org.springframework.web.context.ContextLoaderListener</listener-Class></listener>

<servlet>

<servlet-name>dispatcher</servlet-name><servlet-

class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>dispatcher</servlet-name>

<url-pattern>/send/\*</url-pattern>

</servlet-mapping>

<welcome-file-list>

<welcome-file>index.jsp</welcome-file>

</welcome-file-list>

</web-app>

上述web.xml .xml将在服务器上与传入请求进行交互。 该文件定义了作为调度器的Servlet的名字，ApplicationContext和index.jsp作为欢迎页面。 定义为\*.\*的url模式表示

所有类型的传入请求。

ApplicationContext.xml： - ApplicationContext构建在BeanFactory之上。 它能很容易的

与Springs AOP特性，消息资源处理，事件传播集成。 BeanFactory提供

配置框架和基本功能。 ApplicationContext增加了应用程序的功能。

当在J2EE环境中构建应用程序时，必须使用ApplicationContext。

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:p="http://www.springframework.org/schema/p"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:tx="http://www.springframework.org/schema/tx"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-2.5.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-beans-2.5.xsd

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx-2.5.xsd">

<bean id="superClass" class="packagename.SuperClass" />

<bean id ="subClass" class=" packagename.SubClass">

</bean>

<property name="superClass" ref="superClass"/>

</beans>

上面的文件加载SuperClass.java和SubClass.java的bean。也定义了引用类

Dispatcher-servlet.xml：-

Spring的Web MVC框架是一个请求驱动的Web MVC框架，它是围绕一个servlet设计的

向控制器发送请求，并提供处理应用程序的许多功能。 DispatcherServlet

与Spring ApplicationContext完全集成，并允许您使用spring的特性。

Dispatcher Servlet是应用程序的中央控制单元。它用于定义视图

解析器，bean，handler及其应用程序的映射。

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:p="http://www.springframework.org/schema/p"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:tx="http://www.springframework.org/schema/tx"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-2.5.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-beans-2.5.xsd

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx-2.5.xsd">

<bean id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix">

<value>/WEB-INF/views/</value></property>

<property name="suffix"><value>.jsp</value></property>

</bean>

<bean id="urlMapping" class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">

<property name="mappings">

<props><prop key ="/\*">dispatchController</prop>

</props>

</property>

</bean>

<bean id="dispatchController" class=" packagename.DispatchController"></bean>

</beans>

五，它是怎样工作的 很久也来，工业级的系统都要求快速的开发。有许多工具可以加速开发。但是，随着技术和需求一直在增长，这就需要工具和一些能能够处理日益增长的应用系统的体系架构。工具和体系架构是不同的。Spring是能够支持大型应用系统的MVC架构。在这种技术和系统架构下，一旦搭建好，就能在不更改已有代码的情况下非常容易的扩张系统的功能。它使用xml文件来给应用系统增加新的映射，请求，java bean等

六

架构的优点

让我们来看一看spring web MVC框架能给一个项目带来的好处。

Spring有效地组织了您的中间层对象，EJB不影响它。配置管理服务

可用于任何架构层和任何运行时环境。

Spring Web MVC框架使用MVC设计模式，对于快速开发Web应用程序是一个强大，灵活和设计良好的框架。

Spring避免了大量单例模式的使用。这是一个降低程序可测试性和对象方向的主要问题。

明确分离角色：Spring MVC很好地分离了组成应用程序的不同组件。所有组件，如控制器，命令对象和评估者的每个组件扮演

不同的角色。

适配器控制器：如果您的应用程序不需要HTML表单，您可以编写一个更简单的版本的不需要组成一个控制器的所有组件的Spring控制器。 Spring提供几个控制器，每一个都有不同的作用

Spring通过在应用程序中使用一个一致的方式处理配置来避免使用各种自定义属性文件格式的需要

Spring通过面向接口编程，而不是面向实现编程提供了良好的编程实践。

使用它构建的应用程序依赖其中的少量API。 Spring应用程序中的大多数业务对象都没有

依赖Spring。

使用Spring构建的应用程序非常容易进行单元测试。

Spring可以把EJB作为一个实现选项，而不是由应用程序架构决定。

您可以选择将业务接口实现为POJO或本地EJB，而不会影响调用代码。

Spring提供了适用于许多应用程序的替代EJB。 它可以使用AOP来提供

声明式事务管理，而不使用EJB容器。

Spring为数据访问提供了一致的框架，无论是使用JDBC还是使用O / R映射，Hibernate。 它

在JDBC，JMS，JavaMail，JNDI和许多API等领域提供了一致和简单的编程模型

这使它成为理想的架构。

这是使用POJO构建应用程序的框架。 它也降低了开发人员的复杂性。

在使用JDBC时，它解决了连接泄漏的问题，我们只需要编写必要的SQL，也可以解决

从数据库返回的错误问题。

七。结论

Spring WEB mvc框架是为RAD中的应用程序提供环境的框架

环境。 在这个框架下，我们可以依靠应用程序的一致性，性能和可靠性。

由于这是一个开源环境，所以建议开发人员在开发大型Web应用时继续使用这项技术

。

引用文献

[1] Shu-qiang Huang, Huan-ming Zhang, ” Research on Improved MVC Design Pattern Based on Struts and XSL” , in Information Science

and Engineering ISISE 08 International Symposium on, 2008, vol. 1 PP. 451 – 455

[2] Juanjuan Yan; Bo Chen; Xiu-e Gao, "Le Wang; Research of Structure Integration Based on Struts and Hibernate" , in 2009 WRI World

Congress on Computer Science and Information Engineering,2009, vol. 7,PP. 530-534

[3] Wojciechowski, J.; Sakowicz, B.; Dura, K.; Napieralski, A.,"MVC model, struts framework and file upload issues in web applications

based on J2EE platform", in Proceedings of the International Conference on Modern Problems of Radio Engineering, Telecommunications

and Computer Science 2004, 2004, ,PP 342-345

[4] Erxiang Chen; Minghui Liu, "Research and Design on Library Management System Based on Struts and Hibernate Framework", in WASE

International Conference on Information Engineering ICIE 09, 2009, Vol. 2,PP. 310-313

[5] Yonglei Tao; "Component- vs. application-level MVC architecture", in Frontiers in Education 2002 FIE 2002. 32nd Annual,2002, Vol 1,PP.

T2G-7 - T2G-10

[6] Meiyu Fang, "Design and Implement of a Web Examination System Using Struts and EJB" ,Seventh International Conference on in Web-

based Learning 2008, ,, 2008,pp. 25-28

[7] Wang Ning; Li Liming; Wang Yanzhang; Wang Yi-bing; Wang Jing,"Research on the Web Information System Development Platform

Based on MVC Design Pattern",in IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2008 ,

Vol 3,pp. 203-206

大学生考勤管理系统

管理学生

管理教师

多维度统计考勤

修改密码

找回密码

管理自己班级的考勤

登陆

退出

登陆

退出

修改密码

找回密码

查询自己的考勤信息

管理员模块

教师模块

学生模块