

struts1和struts2和springMVC的区别和介绍

MVC是web开发常用的模式，M即模型层（Model）：主要由javabean来实现。V即视图层（View）：主要由jsp、velocity、freemarker等。C即控制层（Controller）：主要由servlet、struts、springmvc来实现。

struts1

Struts1 工作流程

- 1 ActionServlet接收由浏览器发出的请求后，检索和请求匹配的ActionMapping对象。如果不存在，返回请求无效的信息。
- 2 检索和请求匹配的ActionForm对象。如果不存在，就创建一个。把提交的表单中的数据保存在ActionForm对象中
- 3 根据配置信息决定是否进行校验。如果是，就进行校验。如果通过校验，进行下一步
- 4 ActionServlet根据ActionMapping对象包含的信息决定把请求转发给哪个Action。如果这个Action的对象不存在，就创建一个，然后调用它的execute方法
- 5 Action的execute方法返回一个ActionForward对象。ActionServlet把请求转发给ActionForward指定的JSP。
- 6 ActionForward对象指定的JSP生成动态网页，返回给浏览器

struts2

struts2是webwork升级版，跟struts1没有关系。struts1和servlet耦合度高，各层之间耦合度高，单元测试困难，表示层技术单一，而struts2没有和servlet耦合，更容易测试，支持更多表示层技术。

struts2工作流程

- 1) 客户端初始化一个指向Servlet容器（例如Tomcat）的请求。
- 2) 这个请求经过一系列的过滤器（Filter）。
- 3) 接着FilterDispatcher被调用，FilterDispatcher询问ActionMapper来决定这个请求是否需要调用某个Action。
- 4) 如果ActionMapper决定需要调用某个Action，FilterDispatcher把请求的处理交给ActionProxy。
- 5) ActionProxy通过Configuration Manager询问框架的配置文件，找到需要调用的Action类。
- 6) ActionProxy创建一个ActionInvocation的实例。

7) ActionInvocation实例使用命名模式来调用, 在调用Action的过程前后, 涉及到相关拦截器(Interceptor)的调用。

8) 一旦Action执行完毕, ActionInvocation负责根据struts.xml中的配置找到对应的返回结果。

Struts2的核心就是拦截器。Struts.xml中所有的package都要extends="struts-default"。同理与所有的Java类都要extends自Object一样。struts-default.xml里面就是要做以上事情。

springmvc

1. [spring](#) mvc将所有的请求都提交给DispatcherServlet,它会委托应用系统的其他模块负责对请求进行真正的处理工作。

2.DispatcherServlet查询一个或多个HandlerMapping,找到处理请求的Controller.

3.DispatcherServlet将请求提交到目标Controller

4.Controller进行业务逻辑处理后, 会返回一个ModelAndView

5.Dispathcher查询一个或多个ViewResolver视图解析器,找到ModelAndView对象指定的视图对象

6.视图对象负责渲染返回给客户端。

struts1与struts2的比较

1、struts1要求Action类继承一个抽象基类, 而不是接口, struts2继承ActionSupport类或者实现Action接口。

2、struts1是单例模式的, Action资源必须是线程安全的或同步的, 会有线程安全问题, struts2为每一个请求都实例化一个对象, 不存在线程安全问题。

3、struts1依赖servlet API 因为execute方法的参数有HttpServletRequest和HttpServletResponse对象, 而struts2不用依赖servlet。

4、struts1依赖servlet API, 所以[测试](#)要依赖web容器, 测试难, struts2不依赖于容器, 允许Action脱离容器单独被测试。

5、ActionForm, 普通的JavaBean不能用作ActionForm, 因此, 开发者必须创建大量的ActionForm类封装用户请求参数, Struts 2直接使用Action属性来封装用户请求属性, 避免了开发者需要大量开发ActionForm类的烦琐。Struts 2提供了ModelDriven模式, 可以让开发者使用单独的Model对象来封装用户请求参数, 但该Model对象无需继承任何Struts 2基类, 是一个POJO, 从而降低了代码污染。

6、struts1 支持jsp作为表现层技术, struts2支持velocity、freemarker等表现层技术。

7、Struts1 整合了JSTL, 因此使用JSTL EL。这种EL有基本对象图遍历, 但是对集合和索引属性的支持很弱。Struts2可以使用JSTL, 但是也支持一个更强大和灵活的表达式语言OGNL

8、Struts1支持每一个模块有单独的Request Processors（生命周期），但是模块中的所有Action必须共享相同的生命周期。Struts2支持通过拦截器堆栈（Interceptor Stacks）为每一个Action创建不同的生命周期

struts2与springmvc比较

- 1、spring mvc的入口是servlet，而struts2是filter
- 2、struts2是类级别的拦截，一个类对应一个request上下文，springmvc是方法级别的拦截，一个方法对应一个request上下文，而方法同时又跟一个url对应
- 3、spring会稍微比struts快。spring mvc是基于方法的设计，而struts是基于类，每次发一次请求都会实例一个action，每个action都会被注入属性，而spring基于方法，粒度更细，但要小心把握像在servlet控制数据一样。spring3 mvc是方法级别的拦截，拦截到方法后根据参数上的注解，把request数据注入进去，在spring3 mvc中，一个方法对应一个request上下文。而struts2框架是类级别的拦截，每次来了请求就创建一个Action，然后调用setter getter方法把request中的数据注入；struts2实际上是通过setter getter方法与request打交道的；struts2中，一个Action对象对应一个request上下文。
- 4、spring3mvc的验证也是一个亮点，支持JSR303，处理ajax的请求更是方便，只需一个注解@ResponseBody，然后直接返回响应文本即可。