

<http://blog.csdn.net/chenleixing/article/details/44570681>

- 1、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，所以说从**架构**本身上SpringMVC就容易实现restful url，而struts2的架构实现起来要费劲，因为Struts2中Action的一个方法可以对应一个url，而其类属性却被所有方法共享，这也就无法用注解或其他方式标识其所属方法了。
- 2、由上边原因，SpringMVC的**方法之间基本上独立的，独享request response数据**，请求数据通过参数获取，处理结果通过ModelMap交回给框架，方法之间不共享变量，而Struts2搞的就比较乱，虽然方法之间也是独立的，但其**所有Action变量是共享的**，这不会影响程序运行，却给我们编码 读程序时带来麻烦，每次来了请求就创建一个Action，**一个Action对象对应一个request上下文**。
- 3、由于Struts2需要针对每个request进行封装，把request，session等servlet生命周期的变量封装成一个Map，供给每个Action使用，并保证线程安全，所以在原则上，是**比较耗费内存的**。
- 4、拦截器实现机制上，Struts2有以自己的**interceptor机制**，SpringMVC用的是**独立的AOP方式**，这样导致Struts2的配置文件量还是比SpringMVC大。
- 5、SpringMVC的**入口是servlet，而Struts2是filter**（这里要指出，filter和servlet是不同的。以前认为filter是servlet的一种特殊），这就导致了二者的机制不同，这里就牵涉到servlet和filter的区别了。
- 6、SpringMVC**集成了Ajax，使用非常方便**，只需一个注解@ResponseBody就可以实现，然后直接返回响应文本即可，而Struts2**拦截器集成了Ajax，在Action中处理时一般必须安装插件或者自己写代码集成进去**，使用起来也相对不方便。

7、SpringMVC验证支持JSR303，处理起来相对更加灵活方便，而Struts2验证比较繁琐，感觉太烦乱。

8、Spring MVC和Spring是无缝的。从这个项目的管理和安全上也比Struts2高（当然Struts2也可以通过不同的目录结构和相关配置做到SpringMVC一样的效果，但是需要xml配置的地方不少）。

9、设计思想上，Struts2更加符合OOP的编程思想，SpringMVC就比较谨慎，在servlet上扩展。

10、SpringMVC开发效率和性能高于Struts2。

11、SpringMVC可以认为已经100%零配置。