



链滴

springmvc + shiro 配置 (一、结构及配置文件)

作者: [CM](#)

原文链接: <https://ld246.com/article/1486375067276>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

springmvc + shiro 配置

最近自己做的项目框架升级之后，一直没有时间来整理一个完整的配置笔记，大都是一些小问题，这天项目得空，刚好整理一下，顺便也修正一下配置文件中的冗余配置和不规范。

整个框架涉及的部分分为以下几个部分：

- SpringMVC配置
 - SpringMVC+Hibernate配置
 - SpringMVC+Ibatis配置
 - * SpringMVC+Shiro配置
-

为什么配置了两个持久层

SpringMVC的优缺点在此处就不再阐述，而持久层配置了两个的原因在于两者的优势和不足可以相弥补。

****如此搭配也是初次尝试，在性能等方面的考虑和框架的冗余上必定考虑不足，如有明知的同学，请言指出，万分感谢。****

在简单的增删改查上面，Hibernate做持久层无疑有着优势，利用IDE的功能可以快速的导出Entity和hbm.xml文件，而对这些简单的增删改查，hibernate所提供的find,save,update等方法，是十分便利的。然而，当处理复杂的表关系的时候，Hibernate的应用明显就要复杂很多，对于基础开发人员来说，备足够的sql编写能力，但对于Hibernate的多表联查等复杂操作，及获取返回对象等就显得力有不逮。当团队里面需要快速开发的时候，就需要简化这些学习和摸索的时间，故用Ibatis来弥补这一缺点。Ibatis在此场景里面最大的优势就是sql语句了，所有的操作都可以通过编写SQL语句来实现，这正好补Hibernate的不足。废话不多说，直接上代码。

spring的版本用的4.0+的，之前有文章记录

SpringContext配置文件

```
classpath*:jdbc.properties
```

```
com.web.app.wx.common.entity
```

...

```
org.hibernate.dialect.MySQL5Dialect
false
update
UTF-8
true
${jdbc.url}
${jdbc.driverClassName}
true
5
50
120
100
120
2
true
```

`${jdbc.driverClassName}`

`${jdbc.url}`

`${jdbc.username}`

`${jdbc.password}`

classpath:sqlConfig.xml

GET
HEAD
POST

Servlet配置文件

Spring-Shiro配置文件

shiro 安全框架配置

```
/admin/login = anon  
/admin/logout = anon  
/resources/** = anon  
/menu/** = roles["ROLE_SERVICE,ROLE_USER"]  
/clips/** = roles["ROLE_SERVICE,ROLE_USER"]  
/** = authc
```

文章篇幅有些长，我后续再写几篇来对此进行描述