## Springmvc的工作原理（最好会画图）：见笔记本

## Springmvc的工作机制

<http://blog.csdn.net/zhongyoubing/article/details/51346482>

在容器初始化时，会建立所有url和controller的对应关系，保存到Map集合中，tomcat启动时，会通知spring初始化容器（加载bean的定义信息和初始化所有单例bean），然后然后sringmvc会遍历容器中的bean，获取每一个controller中的所有方法访问url，然后将url和controller保存到一个map中。

这样就可以根据request快速定位到controller,因为最终处理request的是controller中的方法,Map中只保留了url和controller中的对应关系,所以要根据request的url进一步确认controller中的method,这一步工作的原理就是拼接controller的url(controller上@RequestMapping的值)和方法的url(method上@RequestMapping的值),与request的url进行匹配,找到匹配的那个方法。

确定处理请求的method后,接下来的任务就是参数绑定,把request中参数绑定到方法的形式参数上,这一步是整个请求处理过程中最复杂的一个步骤。springmvc提供了两种request参数与方法形参的绑定方法:

① 通过注解进行绑定,@RequestParam

② 通过参数名称进行绑定.

使用注解进行绑定,我们只要在方法参数前面声明@RequestParam(“a”),就可以将request中参数a的值绑定到方法的该参数上.使用参数名称进行绑定的前提是必须要获取方法中参数的名称,**Java**反射只提供了获取方法的参数的类型,并没有提供获取参数名称的方法.springmvc解决这个问题的方法是用asm框架读取字节码文件,来获取方法的参数名称.asm框架是一个字节码操作框架,关于asm更多介绍可以参考它的官网.个人建议,使用注解来完成参数绑定,这样就可以省去asm框架的读取字节码的操作.

## 源码分析

分三个部分来分析：

其一,ApplicationContext初始化时建立所有url和controller类的对应关系(用Map保存);

其二,根据请求url找到对应的controller,并从controller中找到处理请求的方法;

其三,request参数绑定到方法的形参,执行方法处理请求,并返回结果视图.

关键的类：DefaultAnnotationHandlerMapping