**Spring web MVC**框架提供了MVC(模型 - 视图 - 控制器)架构和用于开发灵活和松散耦合的Web应用程序的组件。 **MVC**模式导致应用程序的不同方面(输入逻辑，业务逻辑和UI逻辑)分离，同时提供这些元素之间的松散耦合。

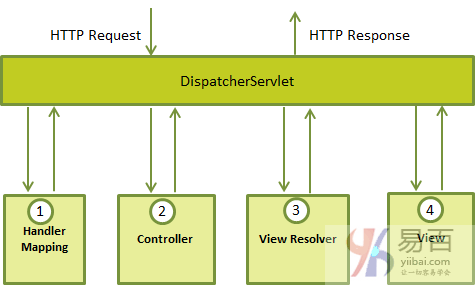
**模型(Model)**封装了应用程序数据，通常它们将由POJO类组成。

**视图(View)**负责渲染模型数据，一般来说它生成客户端浏览器可以解释HTML输出。

**控制器(Controller)**负责处理用户请求并构建适当的模型，并将其传递给视图进行渲染。

**DispatcherServlet组件类**

**Spring Web**模型 - 视图 - 控制器(MVC)框架是围绕DispatcherServlet设计的，它处理所有的HTTP请求和响应。 **Spring Web MVC DispatcherServlet**的请求处理工作流如下图所示：



以下是对应于到DispatcherServlet的传入HTTP请求的事件顺序：

在接收到HTTP请求后，DispatcherServlet会查询HandlerMapping以调用相应的Controller。

Controller接受请求并根据使用的GET或POST方法调用相应的服务方法。 服务方法将基于定义的业务逻辑设置模型数据，并将视图名称返回给DispatcherServlet。

DispatcherServlet将从ViewResolver获取请求的定义视图。

当视图完成，DispatcherServlet将模型数据传递到最终的视图，并在浏览器上呈现。

所有上述组件，即: HandlerMapping，Controller和ViewResolver是WebApplicationContext的一部分，它是普通ApplicationContext的扩展，带有Web应用程序所需的一些额外功能。

一般在配置Servlet时候，  
<description>描述信息</description>  
<display-name>显示名称</display-name>  
<servlet-name>Servlet的名称</servlet-name>  
<servlet-class>Servlet所引用的类名</servlet-class>