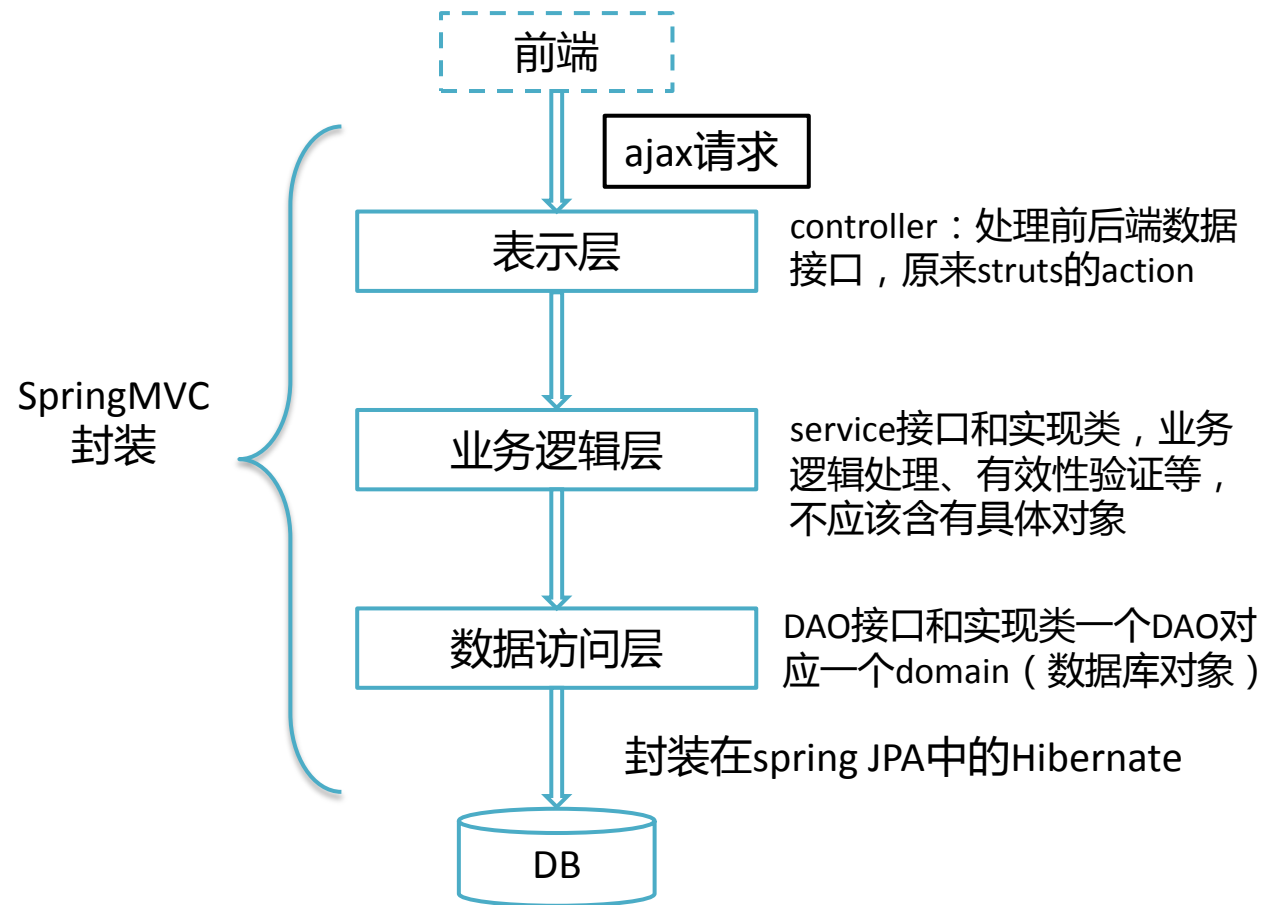


# Java程序结构与SpringMVC

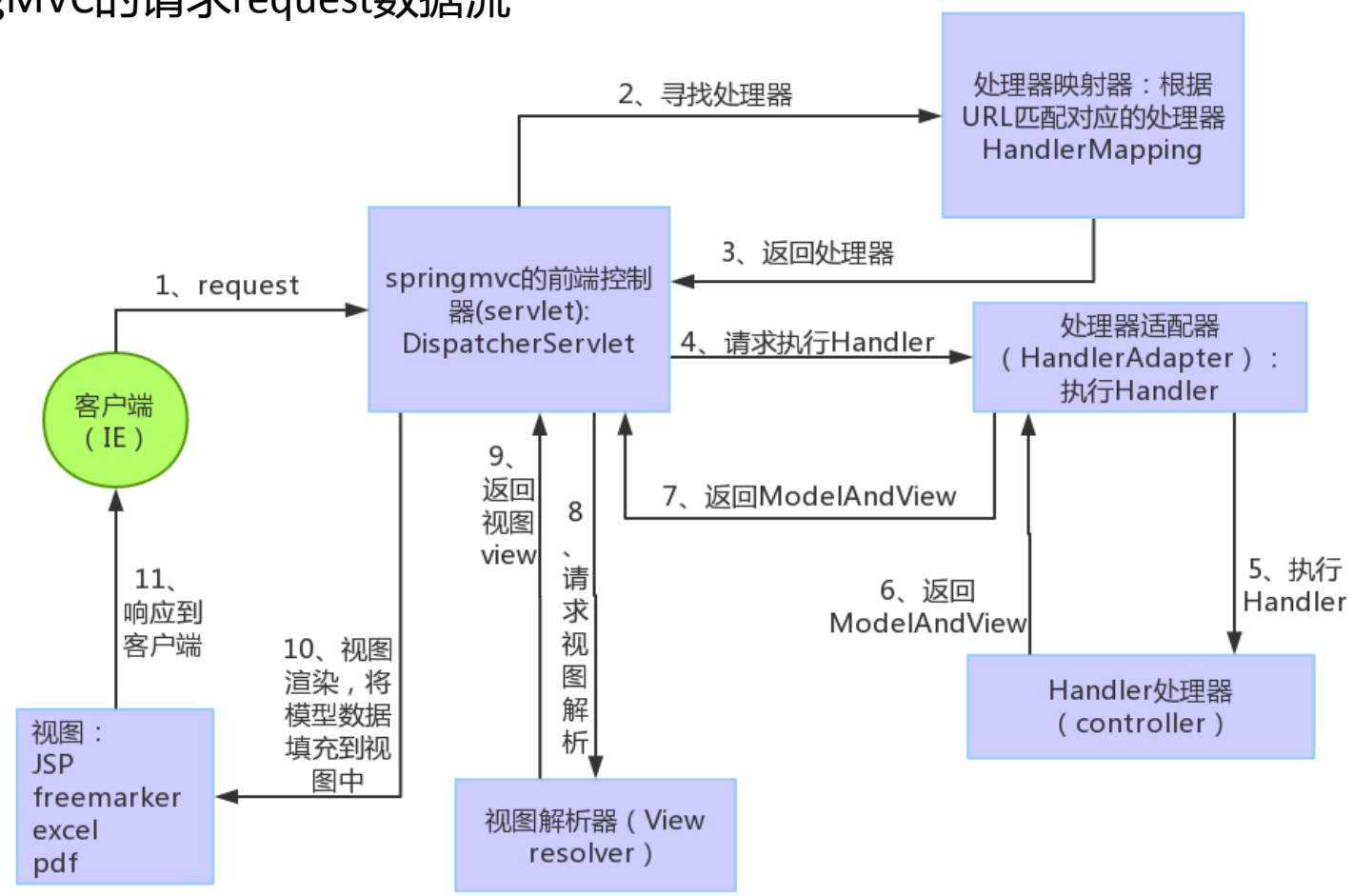


```
xxx/controller/InspectionController.java
@Controller
@RequestMapping("/inspections")
public class InspectionController {...}
```

```
xxx/service/InspectContentService.java
@Service
public class InspectContentService {...}
```

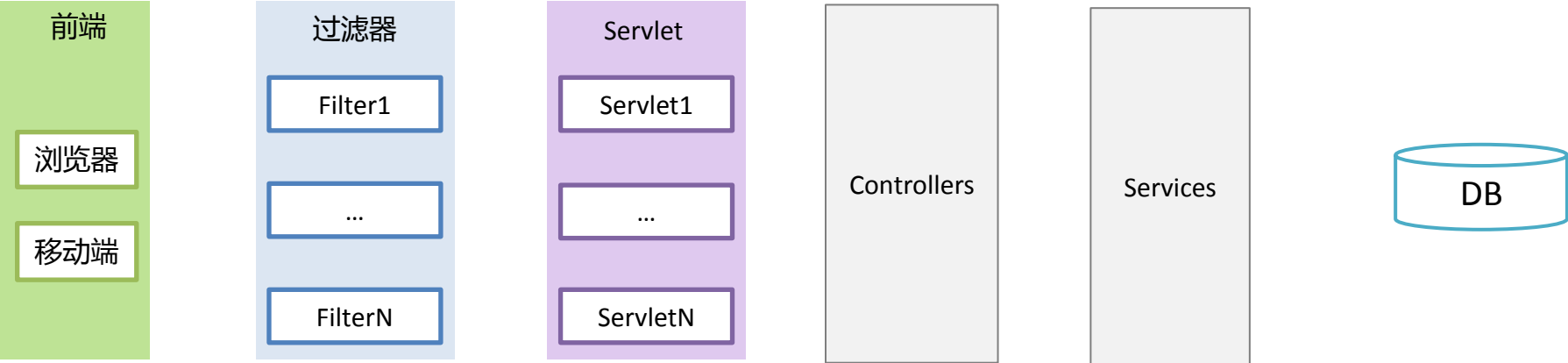
```
xxx/domain/InspectContent.java
@Entity
public class InspectContent {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
```

# 基于springMVC的请求request数据流

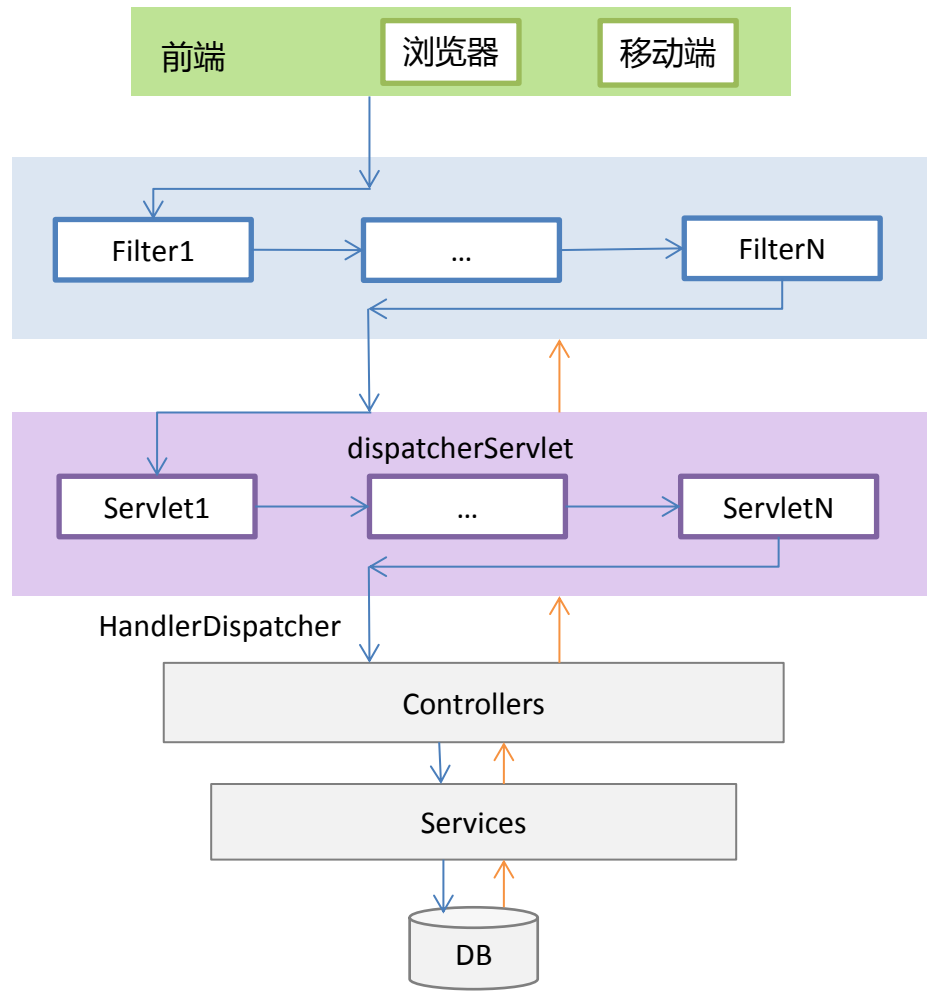


ModelAndView : 封装模型数据信息+视图信息

# 基于springMVC的请求的前后端分离



# 基于springMVC的请求的前后端分离



# Java项目的启动过程

Spring Boot封装

启动JVM进程

Starting xxxApplication on PC-xxx with PID 3256

生成spring 上下文

Refreshing org.springframework.boot.context.embedded.AnnotationConfigEmbeddedWebApplicationContext

启动web容器  
(tomcat服务)

Starting service Tomcat  
Starting Servlet Engine: Apache Tomcat/8.5.11  
Initializing Spring embedded WebApplicationContext

加载web.xml  
解析标签、按顺序  
实例化

CharacterEncodingFilter

编码过滤器

FilterRegistrationBean

WebAppRootListener

获得物理路径

ContextLoaderListener

启动Spring容器，实现了  
ServletContextListener接口

DispatcherServlet

配置DispatcherServlet

ServletRegistration  
Bean注入servlet

Spring嵌入的Hibernate

解析  
application.properties

配置之后才能运行controller

tomcat启动

Tomcat started on port(s): 8080 (http)

web.xml载入过程  
classloader执行过程