

一套吃透 Spring 系列

郝老师



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



目录

Contents

- ◆ Spring整合web环境
- ◆ web层MVC框架思想与设计思路



Spring整合web环境

- Javaweb三大组件及环境特点
- Spring整合web环境的思路及实现
- Spring的web开发组件spring-web

Spring整合web环境

- Javaweb三大组件及环境特点

在Java语言范畴内，web层框架都是基于Javaweb基础组件完成的，所以有必要复习一下Javaweb组件的特点

组件	作用	特点
Servlet	服务端小程序，负责接收客户端请求并作出响应的	单例对象，默认第一次访问创建，可以通过配置指定服务器启动就创建，Servlet创建完毕会执行初始化init方法。每个Servlet有一个service方法，每次访问都会执行service方法，但是缺点是一个业务功能就需要配置一个Servlet
Filter	过滤器，负责对客户端请求进行过滤操作的	单例对象，服务器启动时就创建，对象创建完毕执行init方法，对客户端的请求进行过滤，符合要求的放行，不符合要求的直接响应客户端，执行过滤的核心方法doFilter
Listener	监听器，负责对域对象的创建和属性变化进行监听的	根据类型和作用不同，又可分为监听域对象创建销毁和域对象属性内容变化的，根据监听的域不同，又可以分为监听Request域的，监听Session域的，监听ServletContext域的



Spring整合web环境

- Javaweb三大组件及环境特点
- Spring整合web环境的思路及实现
- Spring的web开发组件spring-web

Spring整合web环境

- Spring整合web环境的思路及实现

在进行Java开发时要遵循三层架构+MVC，Spring操作最核心的就是Spring容器，web层需要注入Service，service层需要注入Dao（Mapper），web层使用Servlet技术充当的话，需要在Servlet中获得Spring容器

```
AnnotationConfigApplicationContext applicationContext =  
    new AnnotationConfigApplicationContext(ApplicationConfig.class);  
AccountService accountService = (AccountService) applicationContext.getBean("accountService");  
accountService.transferMoney("tom", "lucy", 100);
```

web层代码如果都去编写创建AnnotationConfigApplicationContext的代码，那么配置类重复被加载了，Spring容器也重复被创建了，不能每次想从容器中获得一个Bean都得先创建一次容器，这样肯定是不允许。

所以，我们现在的诉求很简单，如下：

- ApplicationContext创建一次，配置类加载一次;
- 最好web服务器启动时，就执行第1步操作，后续直接从容器中获取Bean使用即可;
- ApplicationContext的引用需要在web层任何位置都可以获取到。

Spring整合web环境

- Spring整合web环境的思路及实现

针对以上诉求我们给出解决思路，如下：

- 在ServletContextListener的contextInitialized方法中执行ApplicationContext的创建。或在Servlet的init方法中执行ApplicationContext的创建，并给Servlet的load-on-startup属性一个数字值，确保服务器启动Servlet就创建;
- 将创建好的ApplicationContext存储到ServletContext域中，这样整个web层任何位置就都可以获取到了

Spring整合web环境

- Spring整合web环境的思路及实现

自定义代码完成上述ContextLoaderListener和WebApplicationContextUtils（见代码）



Spring整合web环境

- Javaweb三大组件及环境特点
- Spring整合web环境的思路及实现
- Spring的web开发组件spring-web

Spring整合web环境

- Spring的web开发组件spring-web

到此，就将一开始的诉求都解决了，当然我们能想到的Spring 框架自然也会想到，Spring其实已经为我们定义好了一个ContextLoaderListener，使用方式跟我们上面自己定义的大体一样，但是功能要比我们强百倍，所以，遵循Spring "拿来主义" 的精神，我们直接使用Spring提供的就可以了，开发如下：

先导入Spring-web的坐标：

```
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-web</artifactId>
  <version>5.3.7</version>
</dependency>
```

Spring整合web环境

- Spring的web开发组件spring-web

在web.xml中去配置ContextLoaderListener，并指定配置文件的位置

```
<context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>classpath:applicationContext.xml</param-value>
</context-param>
<listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>
```

在Servlet中直接使用

```
@WebServlet("/accountServlet")
public class AccountServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        ServletContext servletContext = request.getServletContext();
        ApplicationContext applicationContext =
        WebApplicationContextUtils.getWebApplicationContext(servletContext);
        AccountService accountService = applicationContext.getBean(AccountService.class);
        accountService.transferMoney("tom", "lucy", 500);}}}
```

Spring整合web环境

- Spring的web开发组件spring-web

如果核心配置类使用的是注解形式的，那么Spring容器是AnnotationConfigWebApplicationContext，如下配置方式

```
public class MyAnnotationConfigWebApplicationContext extends
AnnotationConfigWebApplicationContext {
    public MyAnnotationConfigWebApplicationContext() {
        //注册核心配置类
        super.register(ApplicationConfig.class);
    }
}
```

```
<context-param>
    <param-name>contextClass</param-name>
    <param-value>com.itheima.web.MyAnnotationConfigWebApplicationContext</param-value>
</context-param>
<listener>
    <listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>
</listener>
```



Spring整合web环境

- Javaweb三大组件及环境特点
- Spring整合web环境的思路及实现
- Spring的web开发组件spring-web



目录

Contents

- ◆ Spring整合web环境
- ◆ web层MVC框架思想与设计思路



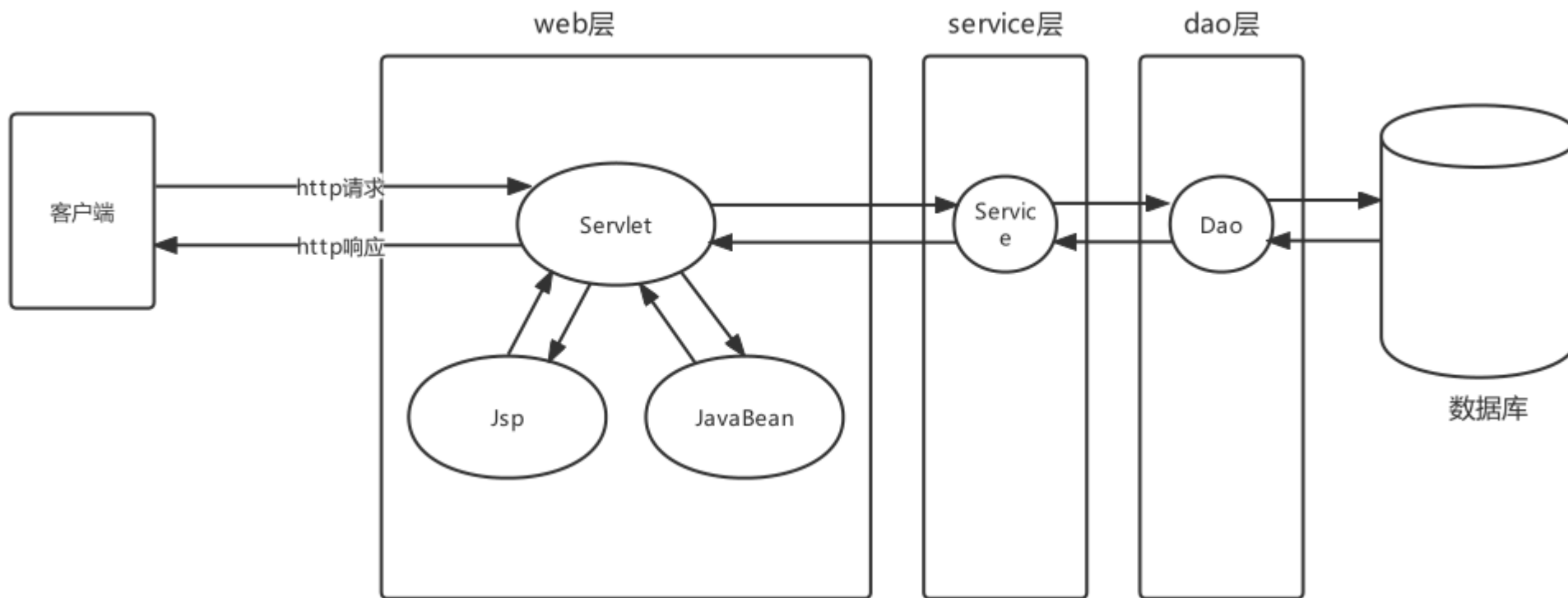
web层MVC框架思想与设计思路

- MVC框架思想及其设计思路

web层MVC框架思想与设计思路

- MVC框架思想及其设计思路

Java程序员在开发一般都是MVC+三层架构，MVC是web开发模式，传统的Javaweb技术栈实现的MVC如下



web层MVC框架思想与设计思路

- MVC框架思想及其设计思路

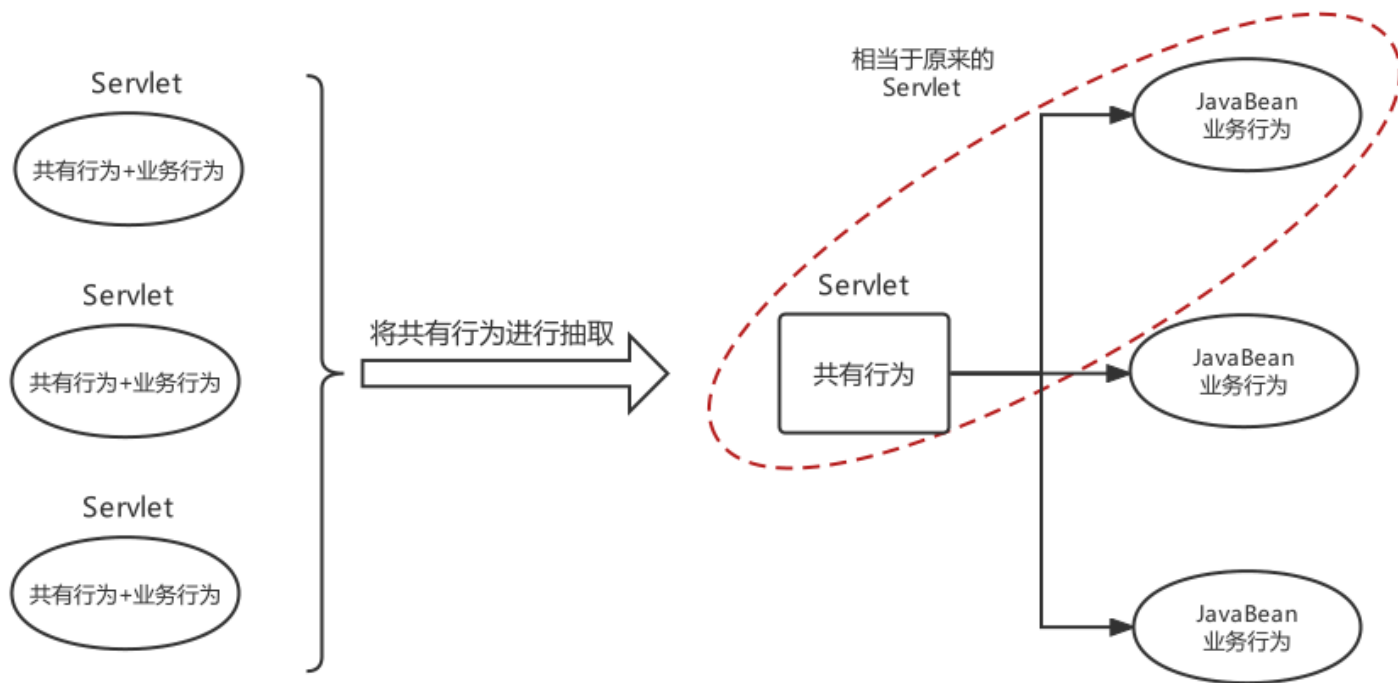
原始Javaweb开发中，Servlet充当Controller的角色，Jsp充当View角色，JavaBean充当模型角色，后期Ajax异步流行后，在加上现在前后端分离开发模式成熟后，View就被原始Html+Vue替代。原始Javaweb开发中，Service充当Controller有很多弊端，显而易见的有如下几个：

Servlet作为Controller的问题	解决思路 and 方案
每个业务功能请求都对应一个Servlet	根据业务模块去划分Controller
每个Servlet的业务操作太繁琐	将通用的行为，功能进行抽取封装
Servlet获得Spring容器的组件只能通过客户端代码去获取，不能优雅的整合	通过Spring的扩展点，去封装一个框架，从原有的Servlet完全接手过来web层的业务

web层MVC框架思想与设计思路

- MVC框架思想及其设计思路

负责共有行为的Servlet称之为前端控制器，负责业务行为的JavaBean称之为控制器Controller



分析前端控制器基本功能如下:

- 1、具备可以映射到业务Bean的能力
- 2、具备可以解析请求参数、封装实体等共有功能
- 3、具备响应视图及响应其他数据的功能



web层MVC框架思想与设计思路

- MVC框架思想及其设计思路



目录

Contents

- ◆ Spring整合web环境
- ◆ web层MVC框架思想与设计思路



传智教育旗下高端IT教育品牌