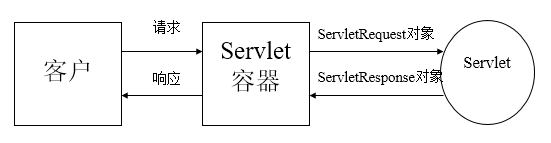
# Servlet

**概念：**所谓Servlet，首先是一个Java类，是一种运行在支持Java应用服务器上的Web组件，**它与普通Java类的区别在于它是一个专门处理请求和响应的Java类**。

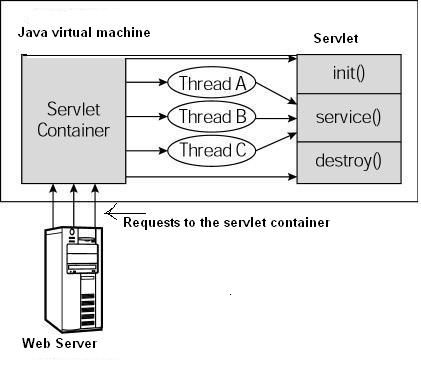
Servlet是成熟的J2EE（Java 2 Enterprise Edition）组件技术之一。Servlet属于目前主流的MVC（Model－View－Controller）架构中的控制器层（Controller）组件，主要功能是接受前端用户的请求，调用后端的逻辑处理程序，最终给客户返回响应。

#### Servlet容器

Servlet容器（Tomcat），其作用是负责处理客户请求，当客户请求来到时，Servlet容器获取请求，然后调用某个Servlet，并把Servlet的执行结果返回给客户



#### Servlet生命周期



Servlet的生命周期就是从servlet出现到销毁的全过程，主要分为以下几个阶段

**JVM加载类——实例化（为对象分配空间）——初始化（执行init(…)方法）——请求处理（服务阶段）——销毁（调用destroy()方法）**

**一个Servlet类执行方法的顺序：**

**构造函数——init(ServletConfig config)——init()——doGet(…)、doPost(…)等——destroy()**

服务器启动时(web.xml中配置load-on-startup=1，默认为0)获第一次请求该servlet时，会初始化servlet对象（即执行初始化方法init(ServletConfig conf)，该servlet对象处理请求它的所有请求（即service(ServletRuquest request， ServletResponse response)，最后服务器关闭时，才会销毁这个servlet对象，执行destroy（）方法

#### Init方法

Init方法被设计成只调用一次，第一次创建servlet的时候被调用，在后续每次用户请求时不再调用。

#### Service方法

Service方法是执行实际任务的主要方法。Servlet容器（即Web服务器）调用service方法来处理来自客户端（浏览器）的请求，并把格式化的相应写回给给客户端

服务器每接收到一个servlet请求时，服务器会产生一个新的线程并调用service方法检查HTTP类型（GET，POST、PUT等），并在适当的时候调用doGet、doPost等方法

#### Destroy方法

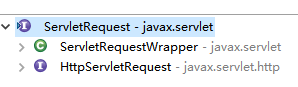
该方法只会被调用一次，在servlet生命周期结束时被调用，调用之后，servlet对象被标记为垃圾

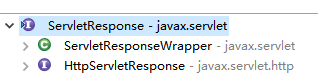
### 源码分析

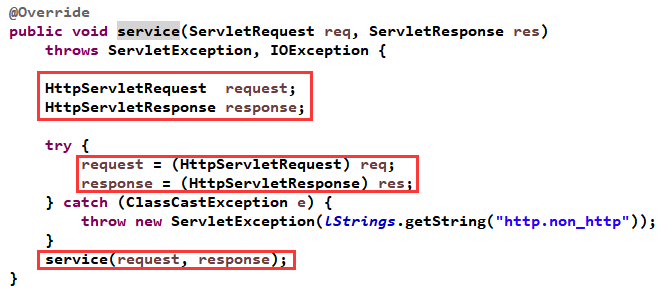
#### HttpServlet类

**1.重写了父类的service方法，把ServletRequest 、ServletResponse转为**

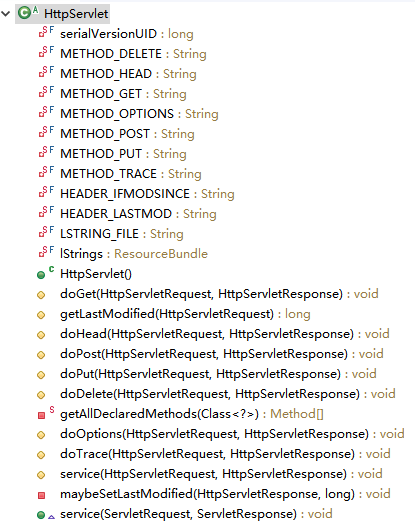
**HttpServletRequest 和HttpServletResponse**







**2.接着调用重载的service方法，并根据请求方法名（get、post等）来执行不同的方法**



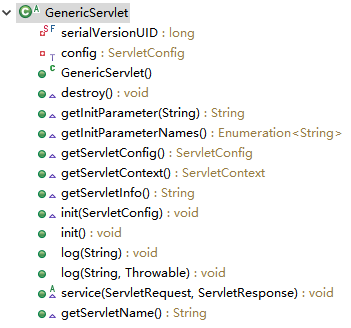
**3.HttpServlet继承了GenericServlet**



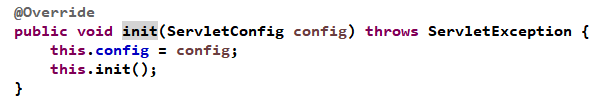
#### GenericServlet类

**1.实现了Servlet和ServletConfig接口**

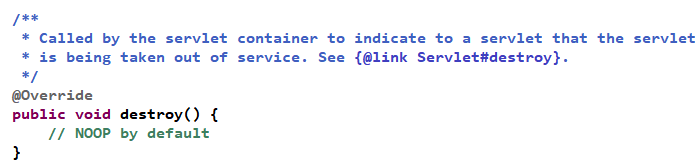




**2.init方法**



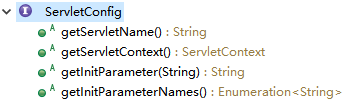
**3.destroy方法**



#### Servlet接口



#### ServletConfig接口



### Servlet基础

#### 保存会话数据的两种技术——cookie、session

**Cookie——客户端技术，存储在客户机上的一个小文本文件**

**读：**用request对象获取客户端的cookie——request.getCookies()

**写：**用response对象写——Cookie c=new Cookie("good","washer");

response.addCookie(c);

**有效期：**如果不设置cookie的过期时间则表示其生命周期为会话期间

设置方法：cookie的setMaxAge(秒为单位)

**Session——服务器端技术，服务器为每一个用户创建一个其独享的session对象，并存储在服务器**

**什么时候会被创建？**JSP的本质是servlet，只有在servlet中调用了request.getSession()或者request.getSession(true)，服务器才会产生session。访问JSP会自动创建session的原因是JSP解析成Java文件时会自动获取session

有效期：session的过期时间是从session不活动的时候开始计算，默认的失效时间

是30分钟，可以用session.invalidate()方法来人为消亡

#### 转发与重定向

转发：request.getRequestDispatcher("url").forward(request, response);

继续把当前的请求对象传递给转发后的页面，因此前后可以共用同一个请求。转发后地址栏不变。转发的范围不能超出当前的Web应用

重定向：response.sendRedirect("url");

结束了本次用户的请求，发送新的请求给下一个页面。重定向后地址栏显示的是新的地址。可以重定向到网络上任意可达的位置

### MVC

概念：MVC是一种 使用MVC（Model View Controller模型-视图-控制器）设计创建Web应用程序的模式

Model（模型）表示应用程序核心（比如数据库记录列表）

View（视图）显示数据（如数据库记录）

Controller（控制器）处理输入（写入数据库记录）

#### 经典的MVC框架

Model（模型）-JavaBean实现，用于封装业务数据

View（视图）-JSP实现，用于显示数据

Controller（控制器）-Servlet实现，用于控制模型和视图