**深入理解Java Servlet与Web容器之间的关系**

2010-05-27 15:05 Jacob.G [51CTO](http://developer.51cto.com/java/) 字号：[**T**](javascript:setfont(12);) | [**T**](javascript:setfont(16);)



理解Java Servlet与Web容器之间的关系，有利于理解Java Web应用的运行方式。进而让Java开发者能够设计和开发出更合理的Java Web应用程序。

AD：[51CTO 网+首届APP创新评选大赛火热启动——超百万资源等你拿！](http://mdsa.51cto.com/act/2015/applications/)

【51CTO特稿】自从计算机软件开发进入网络时代，就开始涉及到通讯问题。在客户/服务器（也叫C/S应用）时期，每个软件都有自己的客户端和服务器端软件。并且客户端和服务器端之间的通讯协议差别也很大。后来随着互联网的发展，基于浏览器/服务器的应用逐渐成为主流，通讯协议也统一到HTTP协议。但是，在HTTP协议之上，如何处理来自客户端的请求信息，以及如何对请求进行回应，则经历了很长时间也没有统一下来。目前，对于这个问题的解决方案主要有两种，一个是CGI，另一个是Servlet。

**CGI(Common Gateway Interface)，通用网关接口**

通用网关接口，简称CGI，是一种根据请求信息动态产生回应内容的技术。通过CGI，Web 服务器可以将根据请求不同启动不同的外部程序，并将请求内容转发给该程序，在程序执行结束后，将执行结果作为回应返回给客户端。也就是说，对于每个请求，都要产生一个新的进程进行处理。因为每个进程都会占有很多服务器的资源和时间，这就导致服务器无法同时处理很多的并发请求。另外CGI程序都是与操作系统平台相关的，虽然在互联网爆发的初期，CGI为开发互联网应用做出了很大的贡献，但是随着技术的发展，开始逐渐衰落。

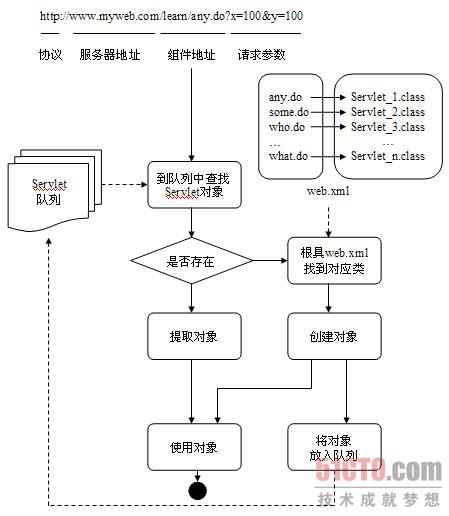
**Servlet**

Servlet最初是在1995年由James Gosling 提出的，因为使用该技术需要复杂的Web服务器支持，所以当时并没有得到重视，也就放弃了。后来随着Web应用复杂度的提升，并要求提供更高的并发处理能力，Servlet被重新捡起，并在Java平台上得到实现，现在提起Servlet，指的都是Java Servlet。Java Servlet要求必须运行在Web服务器当中，与Web服务器之间属于分工和互补关系。确切的说，在实际运行的时候Java Servlet与Web服务器会融为一体，如同一个程序一样运行在同一个Java虚拟机（JVM）当中。与CGI不同的是，Servlet对每个请求都是单独启动一个线程，而不是进程。这种处理方式大幅度地降低了系统里的进程数量，提高了系统的并发处理能力。另外因为Java Servlet是运行在虚拟机之上的，也就解决了跨平台问题。如果没有Servlet的出现，也就没有互联网的今天。  
在Servlet出现之后，随着使用范围的扩大，人们发现了它的一个很大的一个弊端。那就是为了能够输出HTML格式内容，需要编写大量重复代码，造成不必要的重复劳动。为了解决这个问题，基于Servlet技术产生了JavaServet Pages技术，也就是JSP。Servlet和JSP两者分工协作，Servlet侧重于解决运算和业务逻辑问题，JSP则侧重于解决展示问题。Servlet与JSP一起为Web应用开发带来了巨大的贡献，后来出现的众多Java Web应用开发框架都是基于这两种技术的，更确切的说，都是基于Servlet技术的。

**Java Servlet与Web容器之间的关系**

Java是一种动态加载和运行的语言。也就是说当应用程序持有一个类的地址（CLASSPATH）和名称（包名和类名）的情况下，可以在程序运行期间任何时候加载这个类，并创建和使用该类的对象。Servlet就是基于这个机制与Web容器融合在一起的。目前已知的所有支持Java Servlet的Web容器都是采用Java开发的。当Web容器接收到来自客户端的请求信息之后，会根据URL中的Web元件地址信息到Servlet队列中查找对应的Servlet对象，如果找到则直接使用，如果没有找到则加载对应的类，并创建对象。也就是说，Servlet对象是在第一次被使用的时候才创建的，并且一旦创建就会被反复使用，不再创建新的对象。所有创建出的Servlet对象会在Web服务器停止运行的时候统一进行垃圾回收。

为了解决客户端请求地址与Java Servlet之间对应关系问题，Web容器需要一个用来描述这种对应关系的文件，一般是web.xml文件。如果一个Web应用程序中存在很多个Servlet，那么web.xml会变得非常庞大。在Servlet 3.0规范推出之后，允许在Servlet代码中使用声明式语法来代替web.xml中的描述信息，这才让web.xml瘦身下来。下图是这个过程的一个示意图。



在这个图中，我们仅仅是概要的，采用以比较容易理解的方式描述了Web容器与Servlet之间的关系，以及当接受到请求之后的处理流程。在实际的Web容器中，会比这要复杂很多。

【编辑推荐】

1. [Java Web应用开发中的一些概念](http://developer.51cto.com/art/201005/201999.htm)
2. [Tomcat 7 应用实测：声明式Servlet 3.0](http://developer.51cto.com/art/201005/202144.htm)
3. [探秘Servlet 3.0中的Web安全改进](http://developer.51cto.com/art/201005/199299.htm)
4. [简化Web应用开发 Servlet 3.0特性详解](http://developer.51cto.com/art/201004/197524.htm)
5. [Servlet 3.0的异步处理](http://developer.51cto.com/art/201001/178237.htm)

【责任编辑：[关德亮](mailto:) TEL：（010）68476606】