由架构图重点分析易混淆web服务器

http://qzs.qzone.qq.com/ac/b.gif

学过Asp.net，利用Asp.net做项目，在IIS发布网站。

学过JSP，得知JSP最终转化成Servlet，并且使用Tomcat部署过java web项目。

在学JSP，Servlet时，web服务器，web应用服务器，Servlet容器，Tomcat，Jboss等等，这些在视频中经常被提到，但是却不知道他们之间的关系，难道是所属关系或等价关系？

不知道你是否能清晰说出这几个专业名词之间的联系？

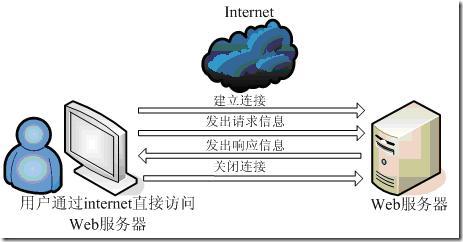
若是你对web服务器了解，可以略过此文章，或者提出宝贵的意见。若是对web服务器概念模糊，建议，可以停下来看看此文章，互相探讨。

一个事物的认识，首先应该是表面概念理解，知道是怎么回事。尤其对于J2EE的常用规范，虽然现在不使用，但是必须知道每个规范的全称以及大概的用途。下次遇到话，知道从何地寻找。类似于索引Index

在说所谓的Jboss，Tomcat ，WebLogic等服务器之前，首先给大家说一下经常提到的Web服务器以及应用程序服务器两者之间的概念。

**Web服务器：**是指驻留于因特网上某种类型计算机的程序。其实就是向客户端发送响应的程序。

由此可以看出：web服务器是一个被动的程序，只有当客户端发送请求时，服务器才能发送响应。

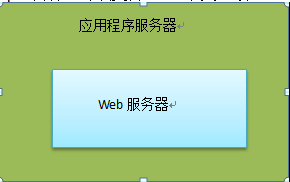


**应用程序服务器：**通过很多协议来为应用程序提供(serves)商业逻辑(business logic)。

**两者区别：**web服务器主要是发送给客户端html浏览。而应用程序服务器主要是为客户端提供商业逻辑（其实对于我们三层来说，就是业务逻辑层）。

应用程序服务器一般是通过组件比如EJB把应用程序的接口把商业逻辑提供给客户端程序，这样，客户端程序通过api，调用存放在服务器端的业务逻辑。并且应用程序服务器一般还是可以管理自己的资源，安全，集成配置等功能。

有很多应用程序服务器，其中可以包括web服务器。从这个概念来说，web服务器是程序服务器一个子集。比如WebLogic既可支持jsp，同样也可以支持EJB（下文继续）



虽然是可以用子集的形式描述，但是后来WebService的出现，WebService传递的是基于xml形式，web服务器与应用程序传递的效果是一样的，所以呢。web服务器与应用服务器基本上不太区分了。

**常见的Web服务器**

目前最流行的两大web服务器是：IIS（Internet information services）和Apache

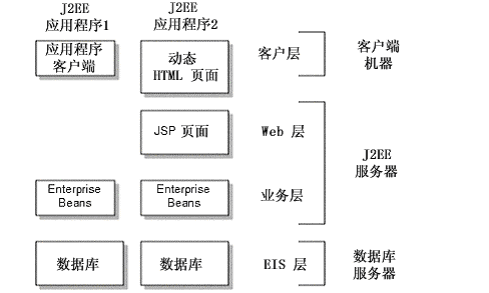
很多著名的网站都是建立在ⅡS的平台上。ⅡS提供了一个图形界面的管理工具，称为Internet服务管理器，可用于监视配置和控制Internet服务。

学过.net的，大家都知道，发布网站都是在IIS上。至于具体的步骤在此不演示，电脑中基本上都自带，直接安装IIS组件即可。

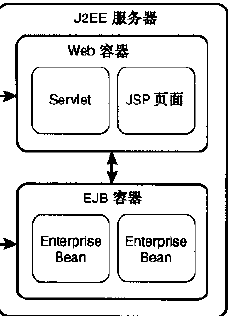
另一个就是Apache（Apche http server）：是Apache软件基金会的一个开放源码的网页服务器，可以在大多数计算机操作系统中运行，由于其多平台和安全性被广泛使用。正是因为它的开源以及移植性，才会一直很流行。它只能运行静态的html，至于动态的jsp，他是无法运行的。

**常见的J2EE的Web服务器以及应用程序服务器**

在介绍基于J2EE服务器时，给大家看一下J2EE的架构图，其中架构图是从我学习[J2EE规范总结博客](http://blog.csdn.net/llhhyy1989/article/details/7002159)中截取的。



同样，我们再看一下J2EE服务器的架构图：



大家心中的J2EE服务器与这两幅图结合，不知道是否对J2EE服务器有了新的认识，不知道是否对常见的J2EE服务器区别有了新的认识。

那我们根据这幅图说一下常见的J2EE中的服务器。

**最熟悉以及最常用的就是Tomcat**

最基本的特点：Tomcat是Web容器而不是应用程序服务器。Tomcat只支持JSP，Servlet，不支持EJB。

背景：Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，它是Apache 软件基金会（Apache Software Foundation）的Jakarta 项目中的一个核心项目，由Apache、sun 和其他一些公司及个人共同开发而成。由于有了Sun 的参与和支持，Servlet和JSP 规范总是能在Tomcat 中得到体现。

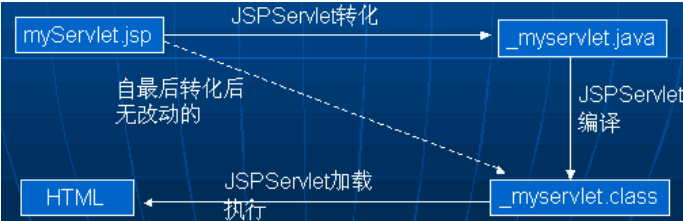
因为Tomcat 技术先进、性能稳定，而且免费，因而深受Java 爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的Web 服务器。

Tomcat 很受广大程序员的喜欢，因为它运行时占用的系统资源小，扩展性好，支持负载平衡与邮件服务等开发应用系统常用的功能。

那我们项目中通常只用tomcat，而tomcat只是一个web容器，只支持JSP，Servlet。我们再次看J2EE的服务器架构体发现，tomcat只是上面一部分。而什么还是可以作为J2EE的服务器呢？

主要是因为我们项目中没有使用的EJB，而是简单的业务逻辑。

那我们再看一下JSP的执行图，进一步领会为啥只用Tomcat，可以不用Apache？



那Apache和Tomcat的关系？

Apache仅仅支持静态网页，对于支持动态网页就会显得无能为力；Tomcat则既能为动态网页服务，同时也能为静态网页提供支持。但是Apache和Tomcat可以联合使用。Apache负责接受所有来自客户端的HTTP请求，然后将Servlets和JSP的请求转发给Tomcat来处理。Tomcat完成处理后，将响应传回给Apache，最后Apache将响应返回给客户端。

**其他常用的应用服务器**

**JBOSS**  
背景：一个基于J2EE的开放源代码的应用服务器。 因为JBoss代码遵循LGPL许可，你可以在任何商业应用中免费使用它，而不用支付费用。JBoss支持EJB 1.1和EJB 2.0 EJB3.0的规范，它是一个管理EJB的容器和服务器。  
特点：免费的，需要的内存和硬盘小，核心的服务是提供EJB，但不支持JSP和Servlet。  
因此：Jboss和tomcat可以配合使用。tomcat支持JSP和Servlet，JBoss支持EJB。请看J2EE服务器图，两者结合，很好的选择。  
但是，经过考证，Jboss的版本不同，支持web容器的程度不同。  
早起的Jboss版本，不支持JSP和Servlet。所以需要和web容器配合使用。  
Jboss4.0版本，支持J2EE中几个常用的规范，像JMail，JDBC，JTA等。并且也很好支持web容器。在Jboss中，嵌套了web容器。所以根据J2EE服务器图而言，这就是为啥能够作为J2EE服务器的原因了。  
**WebLogic**  
背景：WebLogic是美国bea公司出品的一个application server确切的说是一个基于J2EE架构的中间件，webserver是用来构建网站的必要软件用来解析发布网页等功能，它是用纯java开发的。  
特点：它可以支持J2EE中很多规范，比如JDBC，EJB，XML，JMS，JSP等。BEA WebLogic Server是专门为企业电子商务应用系统开发的。企业电子商务应用系统需要快速开发，并要求服务器端组件具有良好的灵活性和安全性，同时还要支持关键任务所必需的扩展、性能、和高可用性。BEA WebLogic Server简化了可移植及可扩展的应用系统的开发，并为其它应用 系统和系统提供了丰富的互操作性。（不是免费的哦）  
**Web Sphere**  
背景：WebSphere是 IBM 的集成软件平台。它包含了编写、运行和监视全天候的工业强度的随需应变 Web 应用程序和跨平台、跨产品解决方案所需要的整个中间件基础设施，如服务器、服务和工具。WebSphere 提供了可靠、灵活和健壮的集成软件。  
旗下的产品：web sphere application server是我们经常说的另一个应用程序服务器。  
特点：支持网络服务和SOAP。 非常详细的和全方位的管理以及运行的监控。（不是免费的哦）