

# SERVIDORES DE APLICACIONES JAVA EE Y MICROSOFT IIS

Desarrollo web en entorno cliente.  
Franklin Cashabamba Casabamba.

## Java EE:

Como consecuencia del éxito del lenguaje de programación Java, el término servidor de aplicaciones usualmente hace referencia a un servidor de aplicaciones Java EE. Entre los servidores de aplicación Java EE privativos más conocidos se encuentran WebLogic de Oracle (antes BEA Systems) y WebSphere de IBM. EAServer de Sybase Inc. es también conocido por ofrecer soporte a otros lenguajes diferentes a Java, como PowerBuilder. Entre los servidores de aplicaciones libres se encuentran JOnAS del consorcio ObjectWeb, JBoss AS de JBoss (división de Red Hat), Geronimo de Apache, TomEE de Apache, Resin Java Application Server de Caucho Technology, Blazix de Desiderata Software, Enhydra Server de Enhydra.org y GlassFish de Oracle. Mucha gente confunde Tomcat como un servidor de aplicaciones; sin embargo, es solamente un contenedor de servlets.

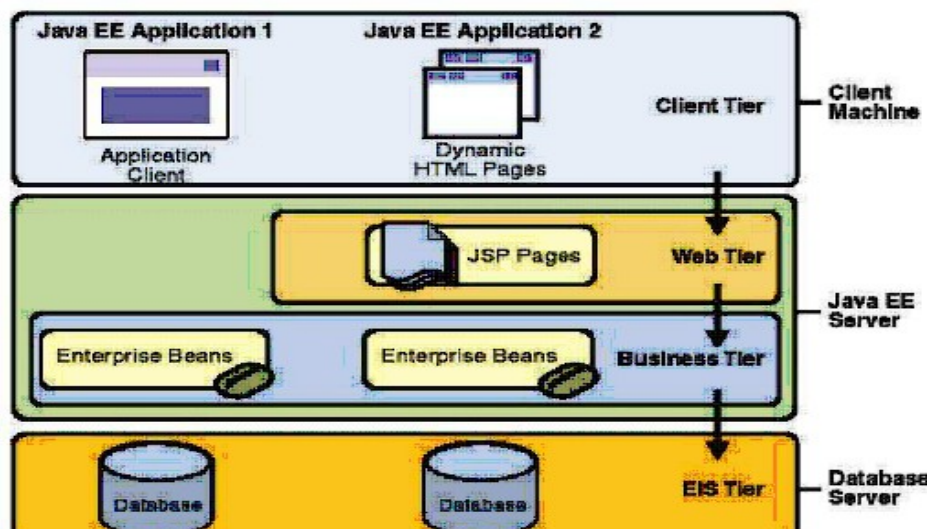
### **Servidores de aplicación Java EE**

La plataforma de Java EE es un conjunto de especificaciones en el API que te permiten construir aplicaciones web. Un contenedor web se encarga de implementar todas las especificaciones del API, además de proveer servicios, para el manejo y ejecución de componentes web, tales como servlets, JSPs, filters, listeners, etc.

Java EE provee estándares que permiten a un servidor de aplicaciones servir como "contenedor" de los componentes que conforman dichas aplicaciones. Estos componentes, escritos en lenguaje Java, usualmente se conocen como Servlets, Java Server Pages (JSPs) y Enterprise JavaBeans (EJBs) y permiten implementar diferentes capas de la aplicación, como la interfaz de usuario, la lógica de negocio, la gestión de sesiones de usuario o el acceso a bases de datos remotas.

La portabilidad de Java también ha permitido que los servidores de aplicación Java EE se encuentren disponibles sobre una gran variedad de plataformas, como Unix, Microsoft Windows y GNU/Linux.

### **Esquema general de servidores Java EE**



## **Arquitectura de Servidores Java EE**

Java EE provee un perfil web y modelo de aplicación estandarizado que permite definir una arquitectura para la implementación de aplicaciones multinivel. Un perfil web permite a los desarrolladores crear aplicaciones web con un conjunto de tecnologías bien definidas. En las aplicaciones multinivel, la funcionalidad de la aplicación es separada dentro de diversas áreas conocidas como niveles.

### **Nivel de Cliente**

Este nivel es el más alto en la arquitectura de Java EE, este el punto en el cual un cliente de nuestra aplicación puede hacer solicitudes al servidor Java EE, el cual comúnmente se encuentra ubicado dentro de otra máquina, en otra región a varios kilómetros del mismo, cuando el cliente envía una solicitud al servidor, este la recibe, procesa y devuelve una respuesta en base a la solicitud del mismo.

### **Nivel Web**

Este nivel es el encargado de manejar las interacciones entre el nivel del cliente y el nivel posterior a este (nivel de negocio), la forma de trabajo de este nivel se puede resumir en los siguientes 5 pasos aun que pueden existir dependiendo de los requerimientos de cada sistema web:

- Recoger los datos ingresados por el cliente.
- Administrar el flujo de las páginas (y la navegación) del cliente.
- Mantener la información y estado de los datos de la sesión por cliente.
- Obtener los resultados de la capa de negocio, a partir de los componentes.
- Presentación de los datos obtenidos de la capa de negocio hacia el cliente.

Una aplicación web puede ser definida como aplicación empresarial de tipo Java EE, sí utiliza componentes y servicios de Java EE, en el nivel web de la aplicación. Una aplicación web puede ser compuesta de las siguientes tres capas.

### **Capa Web**

Esta capa se crea del nivel Web arriba mencionado, la cual utiliza los componentes de Java EE tales como servlets y JSP. Esta capa se encuentra comunicada con la siguiente capa (capa de servicio), pero no debe existir un acoplamiento directo entre estas dos capas.

### **Capa de Servicio**

La capa de servicio crea el Nivel De Negocio mediante el uso de componentes Java EE tales como EJBs. Esta capa se encuentra comunicada con la capa de acceso a datos pero al igual que con la capa anterior no debe existir un acoplamiento directo, ya que la modificación de la capa de acceso a datos no debe afectar la modificación de esta capa. De hecho la capa de servicio no debe conocer nada sobre la capa de acceso a datos y sobre la capa web.

## **Capa de Acceso a Datos**

Esta capa consiste en el Nivel De Datos el cual utiliza componentes de Java EE tales como JDBC y JPA, esta capa no debe de contener ninguna lógica de trabajo, y debe de abstraer el mecanismo de persistencia de la capa de servicio.

Es importante saber que el flujo de este tipo de arquitectura es de arriba hacia abajo (del nivel del cliente hacia el nivel de datos), en otras palabras el nivel del cliente puede llamar al nivel de datos pero no al revés.

Como podemos darnos cuenta la arquitectura de Java EE se encuentra bien definida gracias a Oracle y a la JCP los cuales proveen componentes perfectamente estandarizados y mediante el uso de estos, nosotros podemos construir aplicaciones empresariales exitosamente. Entonces, ¿Porque existen los frameworks para el desarrollo web en Java?

La respuesta es porque al momento de generar aplicaciones web, necesitaríamos que todos los integrantes del equipo de desarrollo conozcan en un nivel muy avanzado los componentes de la arquitectura, lo cual podría resultar muy difícil de conseguir en los mismos, y otra respuesta podría ser, que en aplicaciones muy grandes el manejo y empleo de los componentes es muy repetitivo y ocasionando redundancia dentro del sistema. Encontrado la solución a estos problemas en los diversos frameworks basado en la JVM (Struts, Spring Web MVC, JSF).

## **Microsoft IIS**

Internet Information Services o IIS1 es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Originalmente era parte del Option Pack para Windows NT. Luego fue integrado en otros sistemas operativos de Microsoft destinados a ofrecer servicios, como Windows 2000 o Windows Server 2003. Windows XP Profesional incluye una versión limitada de IIS. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS.2

Este servicio convierte a un PC en un servidor web para Internet o una intranet, es decir que en los ordenadores que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas. Por ejemplo, Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP3 o Perl.4

## Arquitectura y Componentes

Los componentes de IIS están divididos en dos grandes grupos: un grupo de componentes que se encuentra en el modo Kernel y un grupo de componentes que están incorporados en procesos de modo usuario.

### Componentes en modo Kernel

La distribución de componentes en modo Kernel muestra ahora dos componentes adicionales, SSL y Windows Authentication. La razón de esto es la de disminuir el impacto en performance por la gran cantidad de context switches entre HTTP.sys y Isass.exe durante la ejecución de tareas de decodificación y autenticación windows. El resultado es que ahora HTTP.sys realiza decodificación SSL y que el Windows Authentication puede manejarse a través del cache de HTTP.sys.

Aparte de lo mencionado anteriormente, no mucho ha cambiado a nivel de HTTP.sys. Otro par de características que podemos mencionar son la inclusión de contadores de desempeño (performance counters) propios de HTTP.sys, así como integración con el comando NetSh.

### Componentes en modo Usuario

Lo primero que se debe mencionar a nivel de componentes en modo usuario es el hecho de que FTP Service y NNTP Service ya no están alojados en el proceso inetinfo.exe. FTP Service vive ahora en el proceso llamado svchost.exe y NNTP Service ya no es soportado en IIS 7.X/8.X.

El componente Application Host Helper Service (AppHostSvc) se encarga de realizar tareas de respaldo de la configuración de IIS presente en el archivo ApplicationHost.config.

El componente W3ADM fue reemplazado por Windows Process Activation Service (WAS). Este nuevo servicio es el responsable de iniciar y manejar los worker process así como de reciclarlos de forma reactiva. Adicionalmente, a través de WAS es posible trabajar con peticiones utilizando protocolos distintos a HTTP, entonces, a pesar que IIS cumple con el rol de servidor web, WAS nos permite no estar atados al protocolo HTTP.

inetinfo.exe continúa existiendo, más que todo para soportar compatibilidad con aplicaciones legacy. En lugar de la metabase, ahora existe un emulador de la metabase que se parece y se comunica tal como la hacía la metabase original, sólo que en su lugar lo que hace es traducir información para el nuevo modelo de configuración de IIS presente en el archivo ApplicationHost.config.

Isass.exe aún es utilizado para decodificación SSL y Windows Authentication, con el agregado de que una parte importante de su implementación reside en HTTP.sys como mencionamos previamente.

Del lado de los worker process, ahora vemos que tenemos los managed modules y native modules. Los managed modules son componentes .NET que pueden acoplarse al modo integrado de pipeline. No sudéis que más adelante hablaremos de los pipelines.

Los native modules son componentes desarrollados en C++ y que pueden acoplarse a los worker process. De hecho, esta es la forma como se han desarrollado los componentes

nativos desde IIS 6 hasta IIS 8, a través de las native APIs. Esto facilita también todo lo relacionado con soporte de aplicaciones legacy.

### Esquema general

