



JEE 6

Víctor Custodio, 2015

Introducción a JEE

- Es un conjunto de especificaciones
- Destinadas a desarrollar aplicaciones empresariales:
 - Distribuidas
 - Multi-capa y Multi-nivel
 - Escritas en Java
 - Ejecución en un servidor de aplicaciones



Introducción a JEE

- Es un conjunto de especificaciones

Introducción a JEE

- ¿Qué quiere decir que sea un conjunto de especificaciones?
 - Cualquier fabricante puede coger las especificaciones y desarrollar un producto final que las cumpla.
 - Por ejemplo Glassfish, Apache tomcat, Jboss...

Introducción a JEE

- ¿Qué es una especificación?
 - Un documento que detalla cada una de las tecnologías de la plataforma Java EE.
 - Un conjunto de reglas que dictan como debe desarrollarse ese producto (Contenedor web, servidor de aplicaciones...) de tal forma que se pueda garantizar que una aplicación desarrollada siguiendo las especificaciones de Java EE (con tecnologías de la plataforma JEE) pueda desplegarse y ejecutarse

Introducción a JEE

- Destinadas a desarrollar aplicaciones empresariales:
 - Ejecución en un servidor de aplicaciones

Servidores Web y Servidores de Aplicaciones

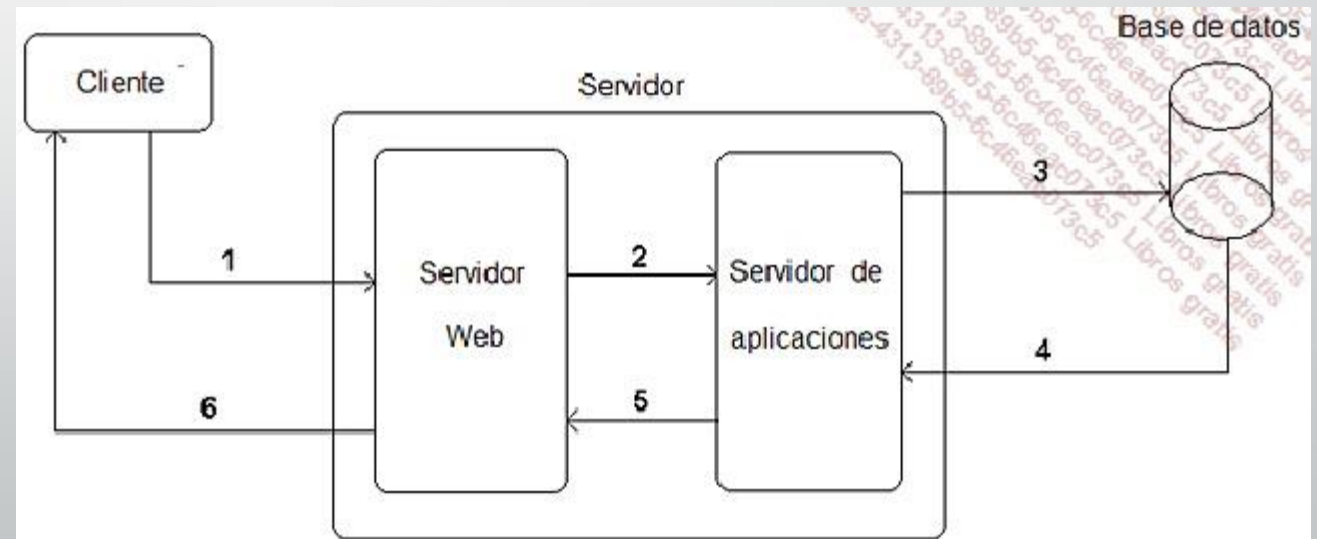
- **¿Un servidor de aplicaciones es un servidor Web? NO.**
- **Un servidor Web**
 - No es más que un simple servidor de archivos.:
 - Los clientes se dirigen a éste mediante el protocolo HTTP para obtener un recurso.
 - Cuando el servidor Web recibe una petición HTTP, extrae de la petición el nombre del recurso solicitado, lo busca en el disco y "lo envuelve" dentro de una respuesta HTTP para transmitirlo al cliente.
 - Éste es el único trabajo que puede realizar un servidor web, el servidor http no realiza ningún tratamiento en el recurso antes de transmitirlo al cliente

Servidores Web y Servidores de aplicaciones

- **Un servidor de aplicaciones:**
 - Los recursos que le son confiados no son simples archivos estáticos, sino que contienen el código que se va a encargarse de ejecutar en nombre de los clientes que realicen la petición.
 - Cuando un servidor de aplicaciones recibe una solicitud HTTP, éste también analiza la petición para determinar qué recurso se le ha solicitado. Generalmente, la petición concierne código ejecutable alojado en el servidor. Contrariamente a lo que haría un servidor Web en la misma situación, no transfiere al cliente el código sino que lo ejecuta y es el resultado de la ejecución de este código lo que se reenvía al cliente.

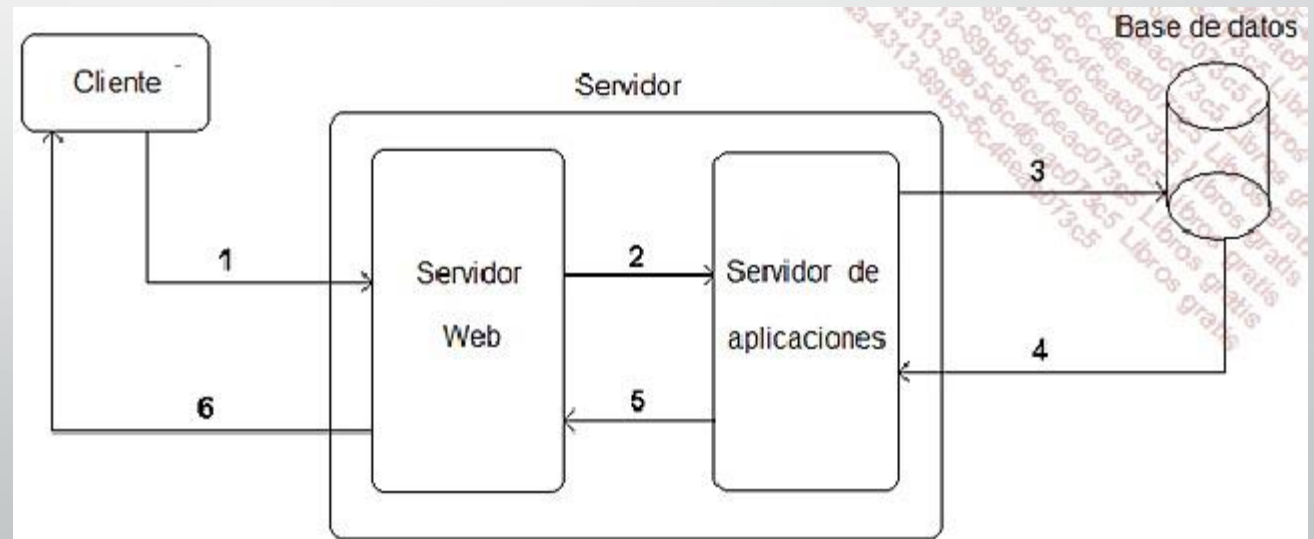
Servidores Web y Servidores de aplicaciones

- **Un servidor de aplicaciones** generalmente toma también las funciones de servidor Web:
 - Cuando el servidor recibe una petición HTTP proveniente del exterior, es la parte del servidor Web la que recibe esta petición y la analiza. Si concierne a un recurso estático, el servidor Web realiza su función
 - Si la petición concierne a un recurso dinámico (código), el servidor Web no sabe tratar esta petición, por lo que la transfiere a la parte correspondiente al servidor de aplicaciones del servidor



Servidores Web y Servidores de aplicaciones

- Éste realiza su función ejecutando el código correspondiente y generando una respuesta HTTP.
- Si así lo requiriera, el servidor de aplicaciones puede contactar con otro servidor o una base de datos para poder construir la respuesta.
- Esta respuesta HTTP se transmite al servidor Web que a su vez se encarga de reenviarla al cliente



Introducción a JEE

- Destinadas a desarrollar aplicaciones empresariales:
 - Escritas en Java

Tecnologías de la plataforma Java EE

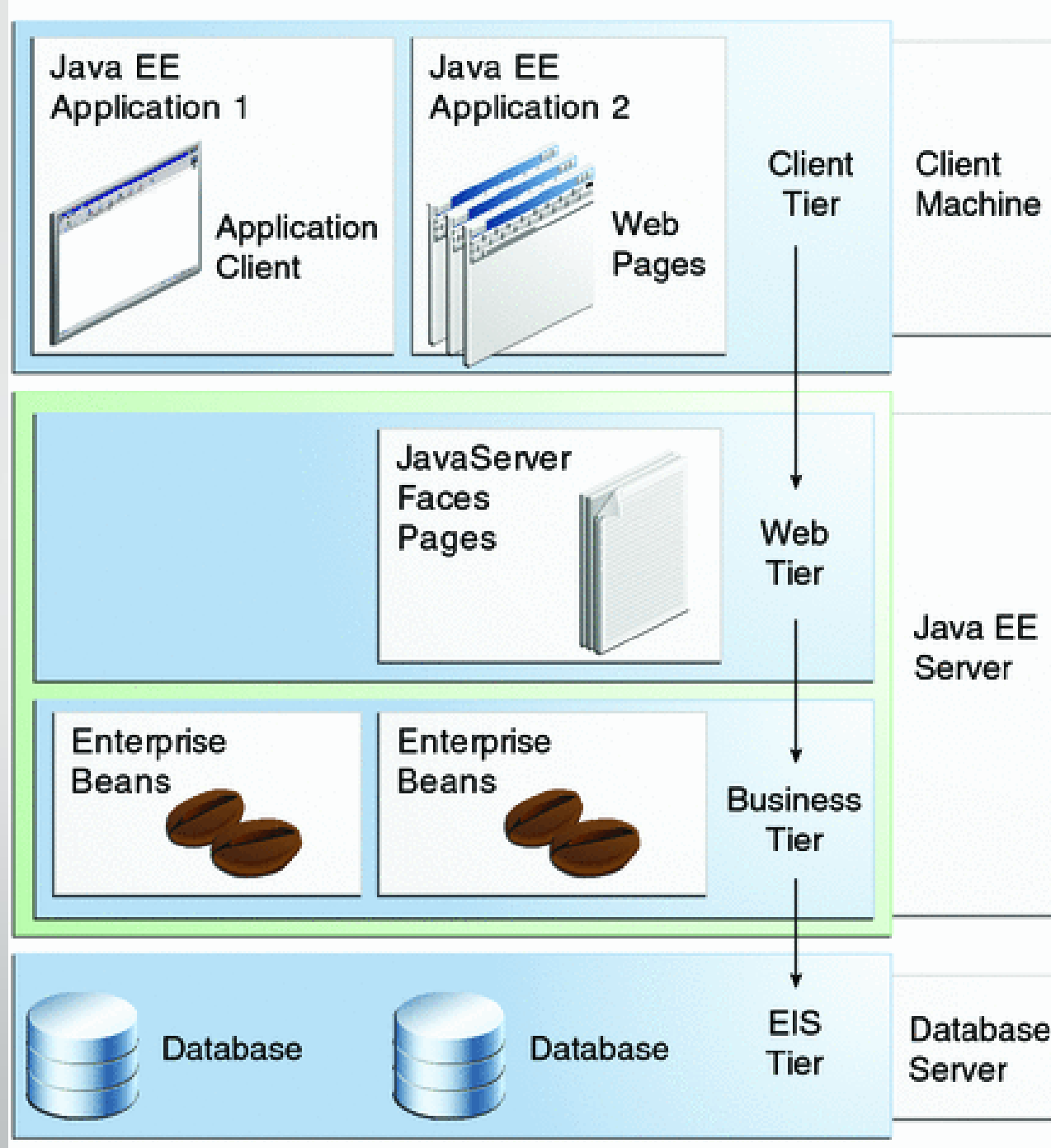
- Enterprise JavaBeans (EJB).
- Java Servlet
- JavaServer Page (JSP)
- JavaServer Pages Standard Tag Library (JSTL).
- JavaServer Faces (JSF)
- Java Message Service (JMS).
- **Java Transaction API (JTA).**
- JavaMail API y JavaBeans Activation Framework (JAF).
- Tecnologías XML (JAXP, JAX-RPC, JAX-WS, JAXB, SAAJ, JAXR)
- **JPA**, JDBC API
- Java Naming and Directory Interface (JNDI)
- Java Authentication and Authorization Service (JAAS)

Introducción a JEE

- Destinadas a desarrollar aplicaciones empresariales:
 - Multi-capa o Multi-nivel

Aplicación multinivel

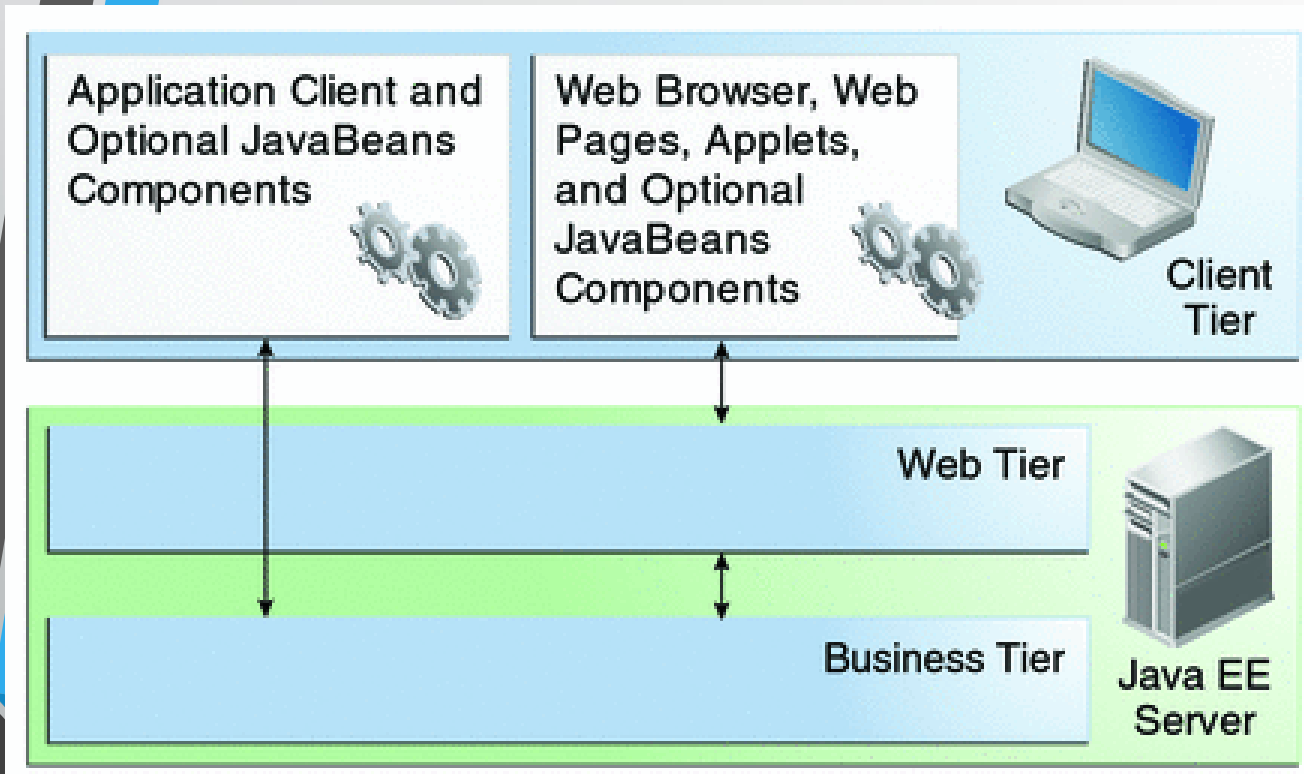
- Podemos separar el desarrollo de la aplicación en diferentes capas según su función
- Aplicaciones JEE suelen ser multinivel y distribuirse en tres localizaciones,
 - ordenadores clientes,
 - el sistema donde se ejecuta el servidor de aplicaciones, y
 - el sistema donde reside la base de datos



Aplicación multinivel

- El nivel del cliente (**Client-tier**) que tiene una capa destinada a mostrar la interfaz gráfica de usuario. Las aplicaciones Java EE pueden ser una aplicación Java Swing normal, o una aplicación Web renderizada en un navegador. Esta capa se ejecuta en el ordenador cliente.
- En el nivel del servidor(**JEE server**) está el nivel de la lógica de negocio (**Business-tier**) y el de la lógica de presentación (**Web-tier**). Estas capas se ejecutan en el servidor de aplicaciones.
- El nivel de los datos (**EIS-tier** (Enterprise information system)) es el destinado a la gestión de los datos. Este nivel puede separarse a su vez en uno o más niveles.

Client- tier



- Los clientes pueden ser varios: navegador web, paginas web, applets, aplicaciones JSE..
- El cliente se comunica con la capa de negocio que se ejecutan en el servidor Java EE, ya sea directamente o, como en el caso de un cliente que se ejecuta en un navegador, pasando a través de páginas web o servlets que se ejecutan en la capa web.

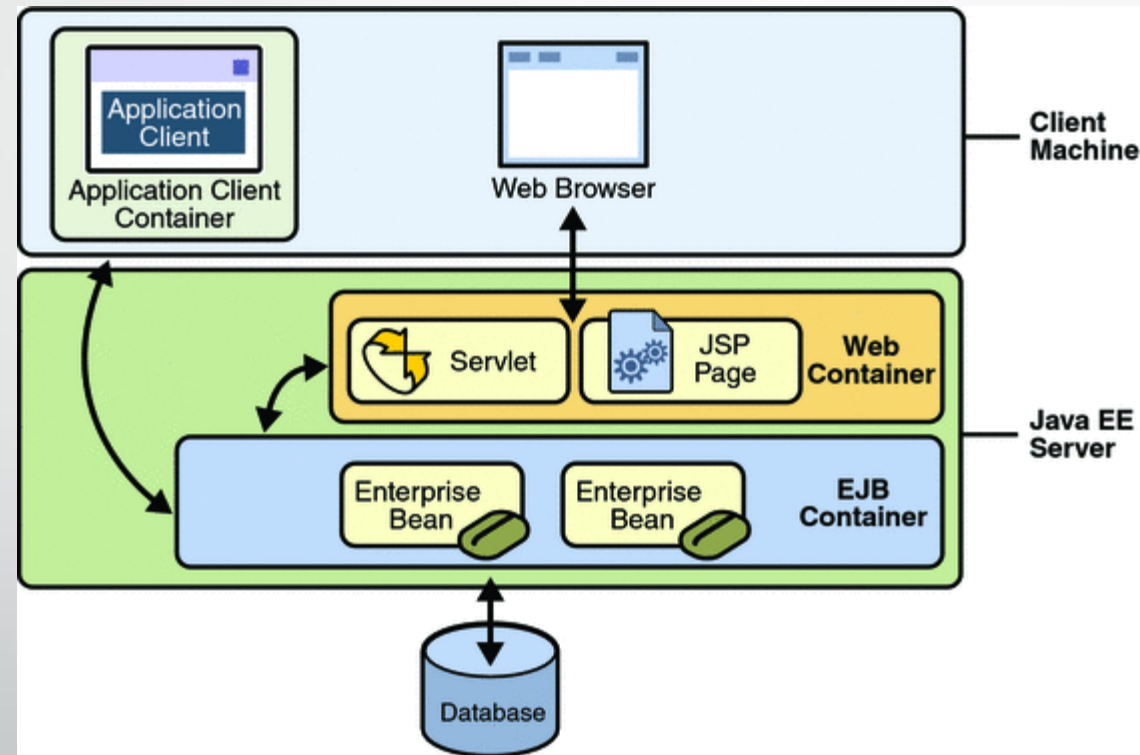
Server-Side

- Componentes de un servidor JEE:
 - Un componente no es más que una unidad de software, puede ser un componente web como una página JSP o un servlet, o un componente EJB, etc.
 - Estos componentes se ejecutan dentro de sus correspondiente **contenedor** dentro del servidor de aplicaciones.

Server-Side

- Contenedores en el Servidor de Aplicaciones o servidor JEE
 - El contenedor es un entorno de ejecución que gestiona los componentes
 - Por eso, los componentes deben de cumplir el contrato que establece el contenedor. Ese contrato no es más que un conjunto de métodos que debe implementar el componente y que permite al contenedor interactuar con él.
- Existen dos tipos de contenedores dentro de un servidor de aplicaciones:
 - **Contenedor WEB** encargado de gestionar los componentes servlets y páginas JSP.
 - **Contenedor EJBs** encargado de gestionar los componentes EJBs.

Server-Side. Contenedores



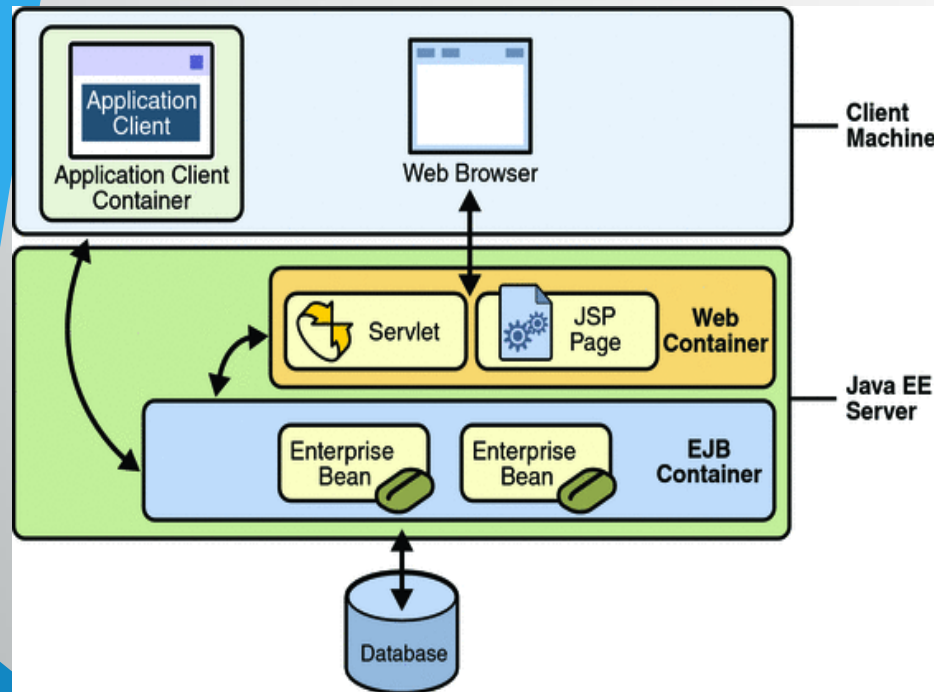
Server-Side. Contenedor Web

- Componentes Web Java EE son :

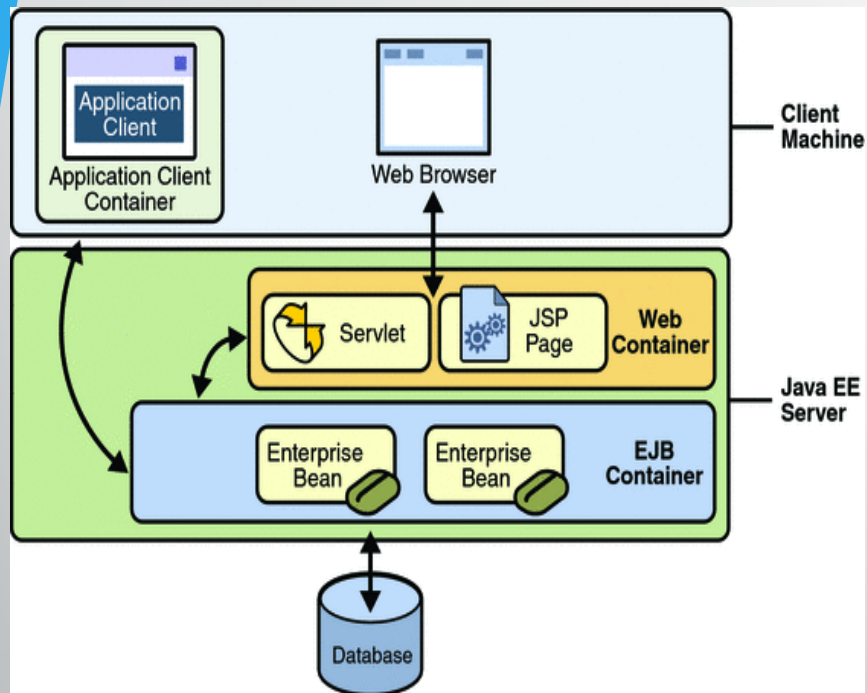
Servlets: clases de lenguaje de programación Java que procesan dinámicamente peticiones y construyen respuestas.

O páginas web creadas con la tecnología JavaServer Faces y / o la tecnología JSP (páginas JSP).

- Las páginas **JSP** son documentos basados en texto que se ejecutan como servlets pero permiten un enfoque más natural para la creación de contenido estático.
- La tecnología **JavaServerFaces** proporciona un marco de componentes de interfaz de usuario para las aplicaciones web.



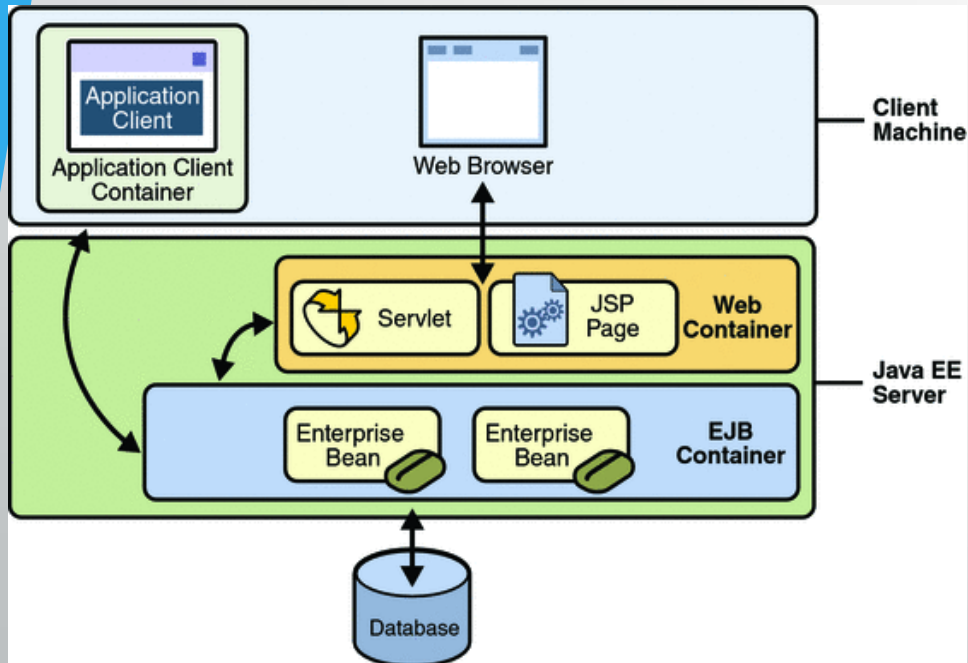
Server-Side. Contenedor Web



- Existen elementos en los componentes Web que no entran dentro de las especificaciones JEE:
 - Páginas HTML estáticas y applets.
 - Clases de utilidad de servidor
 - JavaBeans (Pojos)

Server-Side. Contenedor EJB

- Código de la Empresa:



- Lógica que resuelve o satisface las necesidades de un dominio de negocio en particular, como la banca, el comercio minorista, o de las finanzas,
- Los responsables son los beans de negocio (EJB).
 - En ocasiones estos beans de negocios también pueden ser usados en el web container

Contenedores

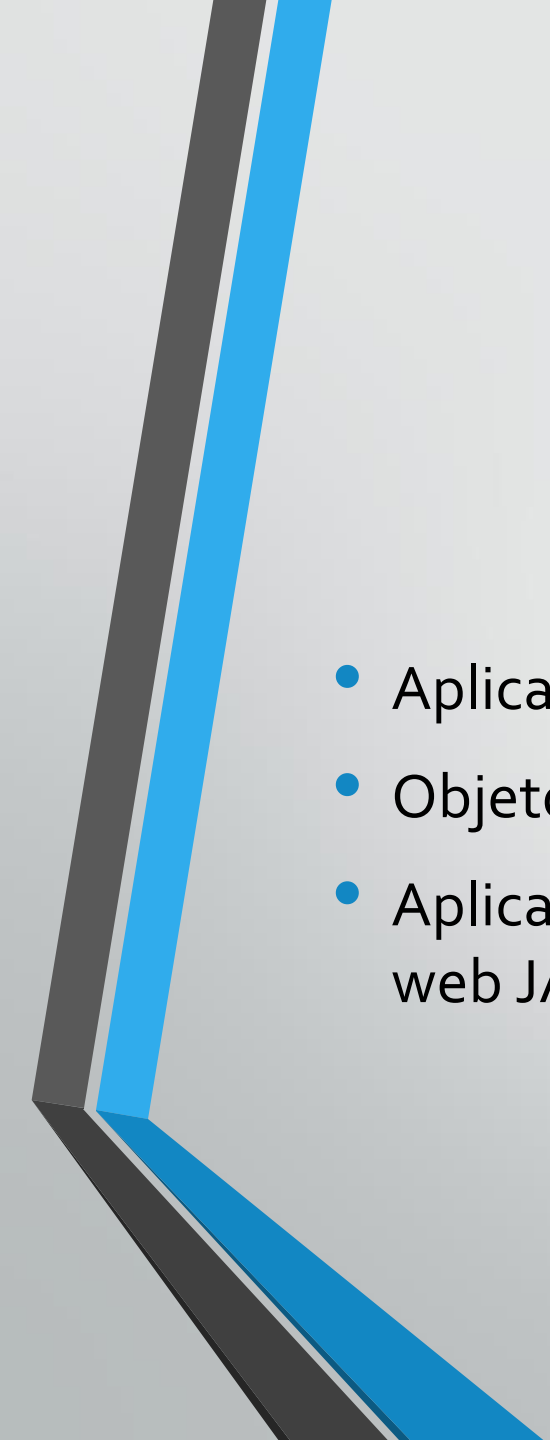
- Cada contenedor proporciona una serie de servicios que el componente puede utilizar.
 - Es el encargado de gestionar el ciclo de vida de los componentes, realizar la reserva de recursos, etc.
 - Algunos de estos servicios son servicios declarativos, esto quiere decir que algunos servicios se declaran en vez de programarse. La declaración se realiza mediante descriptores de despliegue o anotaciones
 - El descriptor de despliegue no es más que un archivo XML que describe como se deben desplegar esos componentes en el contenedor del servidor de aplicaciones.

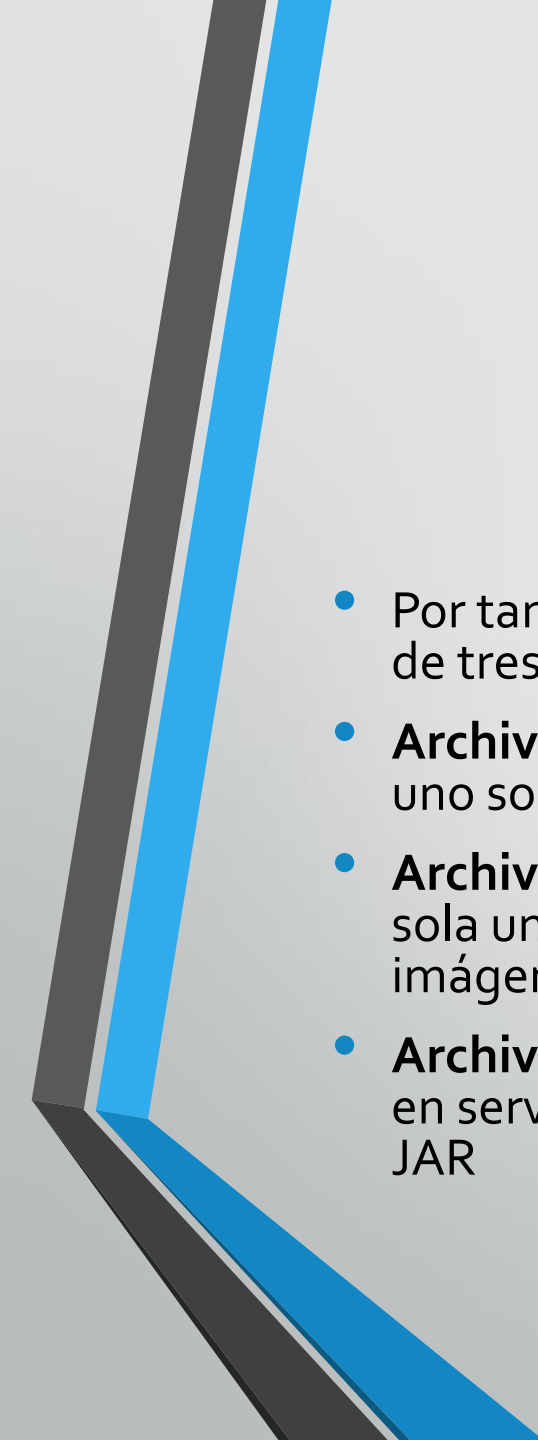
Enterprise Information System Tier – (EIS tier)

- Se encarga de software EIS e incluye sistemas de infraestructura de la empresa, tales como el sistemas de bases de datos y otros sistemas de información heredados.
- Por ejemplo, los componentes de aplicaciones Java EE pueden necesitar acceso a los sistemas de información empresarial para la conectividad de base de datos.



Tipos de aplicaciones JEE

- 
- Aplicaciones Web JAVA. (Contenedor Web)
 - Objetos distribuidos EJBs. (Contenedor EJB)
 - Aplicaciones empresariales que engloba a las dos anteriores, aplicaciones web JAVA y objetos distribuidos EJBs. (Ambos contenedores)

- 
- Por tanto los módulos (empaquetados) que forman una aplicación empresarial pueden ser de tres tipos:
 - **Archivos JAR** (Java Archive): Los archivos JAR permiten agrupar distintos archivos .java en uno solo. Es el empleado para empaquetar componentes EJBs.
 - **Archivos WAR** (Web Application Archive): Los archivos WAR permiten empaquetar en una sola unidad aplicaciones web completas (servlets, páginas JSPs, contenido estático como imágenes y otros recursos Web).
 - **Archivos EAR** (Enterprise Application Archive): Los archivos EAR son archivos desplegables en servidores de aplicaciones JEE. Contienen archivos WAR y EJBs empaquetados en ficheros JAR



Estructura de una aplicación Web Java

Modulo Web (WAR)

Estructura de una aplicación web JEE

- La normativa JEE también describe cómo debe organizarse una aplicación para que pueda ser soportada por cualquier servidor de aplicaciones compatible.
- En la arquitectura Java EE, un módulo web es la unidad desplegable y utilizable más pequeño de los recursos de la web . Un módulo web Java EE corresponde a una aplicación web tal como se define en la especificación de Java Servlet y se despliega en un contenedor Web

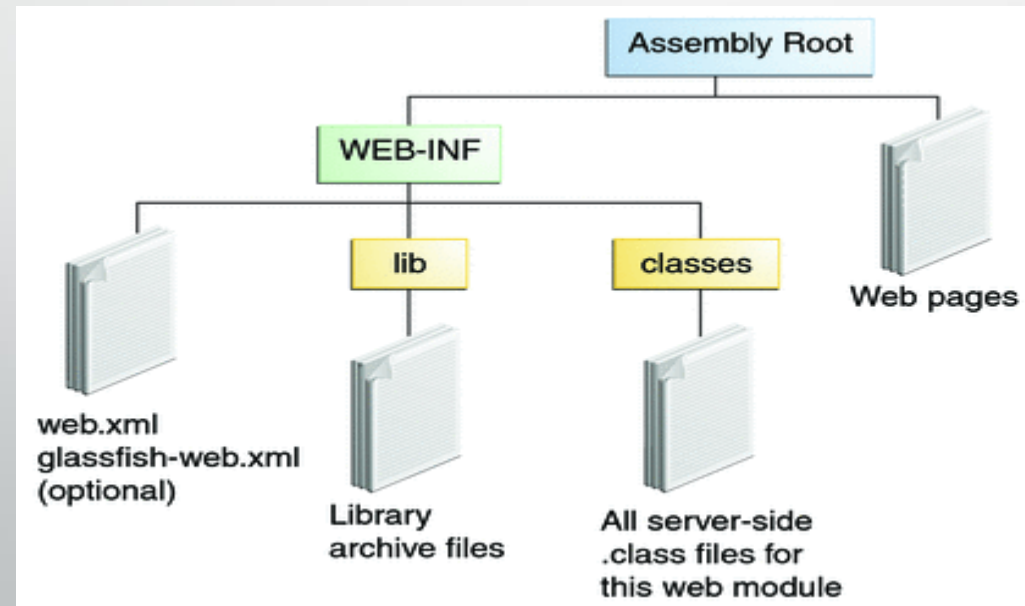
Estructura de una aplicación web JEE

- Un modulo web generalmente se compone de los elementos siguientes:
 - De recursos estáticos: páginas HTML, imágenes, sonidos, hojas de estilo...
 - De recursos dinámicos: servlets, JSP, Java Bean.
 - De librerías de clases utilizadas por los diferentes componentes dinámicos.
 - De descriptor de despliegue que permite definir los parámetros de funcionamiento de la aplicación en el servidor, los enlaces entre las URL y los recursos dinámicos de la aplicación, las páginas por defecto y de error de la aplicación, la seguridad de la aplicación, etc.

**En la actualidad este descriptor de despliegue se sustituye por anotaciones y solo se usa si utilizamos JSF y algunas configuraciones de seguridad*

Estructura de una aplicación web JEE

- Los archivos que contienen estos elementos tienen que organizarse en una forma de árbol concreta para ser fácilmente accesibles por el servidor de aplicaciones. Este árbol básico se presenta en la figura siguiente:



Estructura de una aplicación web JEE

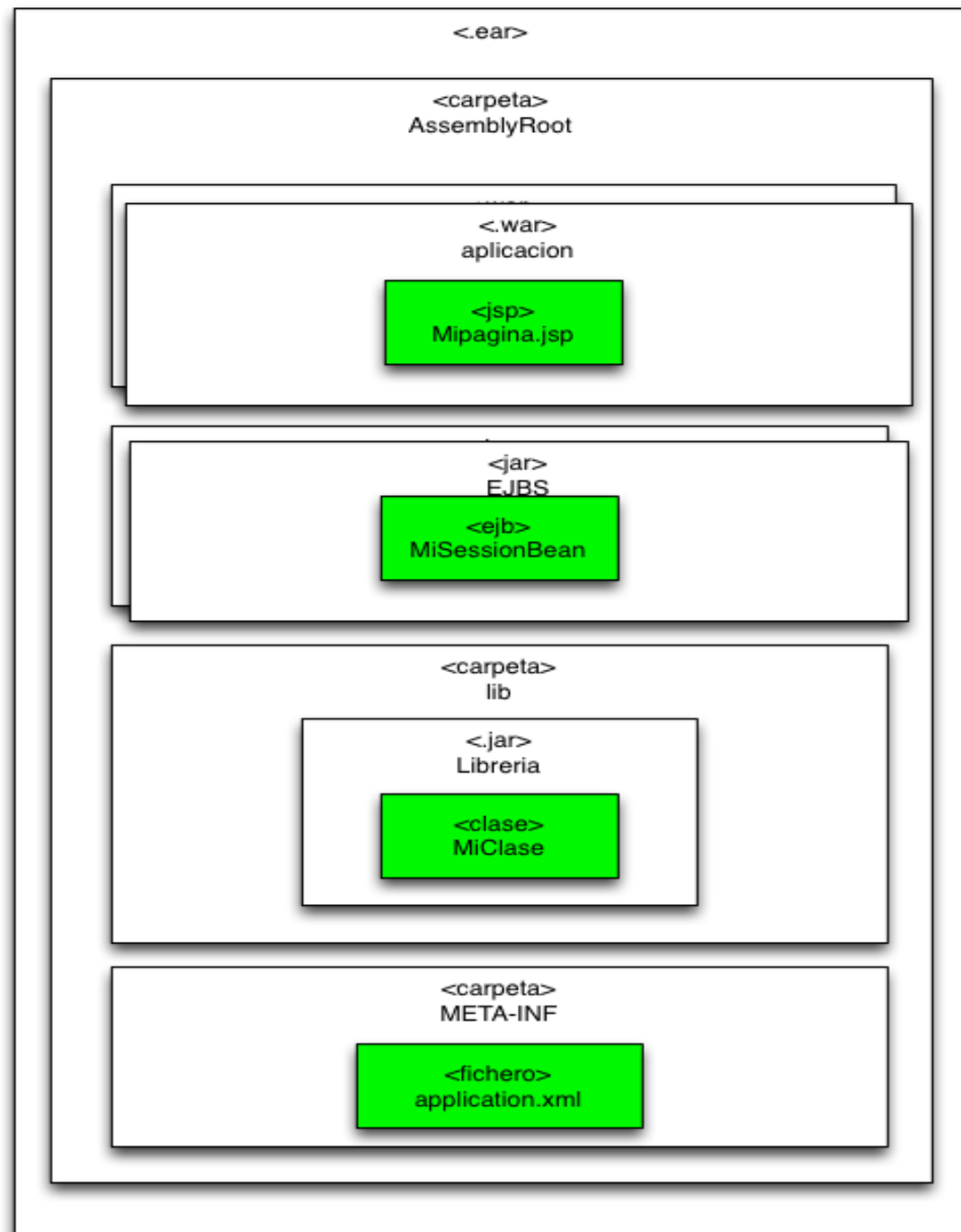
- El directorio de nivel superior de un módulo web es la raíz de la aplicación.
- En la raíz están las páginas XHTML, clases y archivos del cliente , y recursos web estáticos , como imágenes . Además contiene un subdirectorio llamado WEB-INF , que puede contener los siguientes archivos y directorios:
 - **clases**: un directorio que contiene clases del lado del servidor : servlets , archivos de clase enterprise bean , clases de utilidad , y componentes JavaBeans
 - **lib**: Un directorio que contiene los archivos JAR que contienen los granos de la empresa, y los archivos JAR de bibliotecas llamados por clases del lado del servidor
 - **Descriptores de implementación** , tales como web.xml (el descriptor de despliegue de aplicaciones web) y ejb-jar.xml (un descriptor de despliegue EJB)



Estructura de un Java Enterprise Archive (EAR)

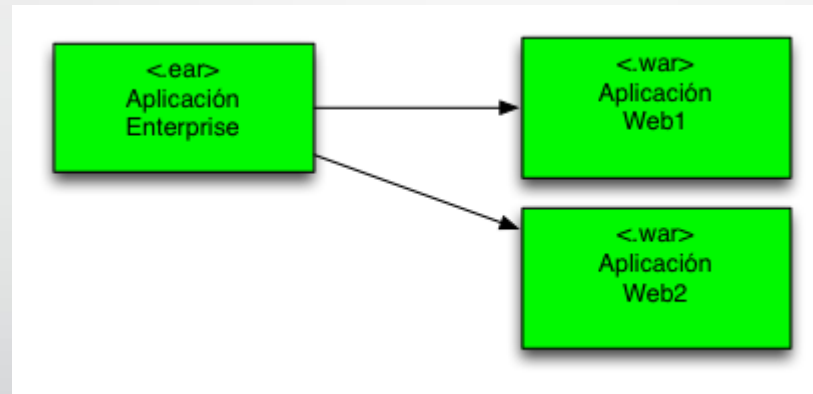
EAR

- Este tipo de ficheros son desplegables en servidores de aplicaciones que soporten el stack completo de JEE .Es decir contengan tanto un web container como un EJB Container . Los ficheros EAR que tienen la siguiente estructura:



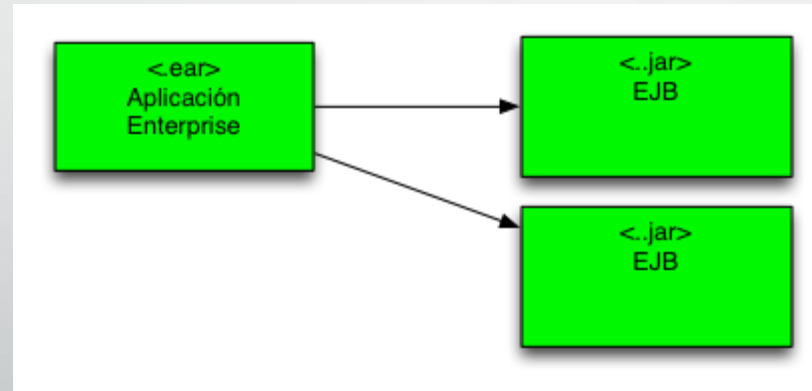
EAR -> ficheros war

- un EAR es capaz de albergar varios ficheros WAR cada uno de los cuales contiene una aplicación web completa.



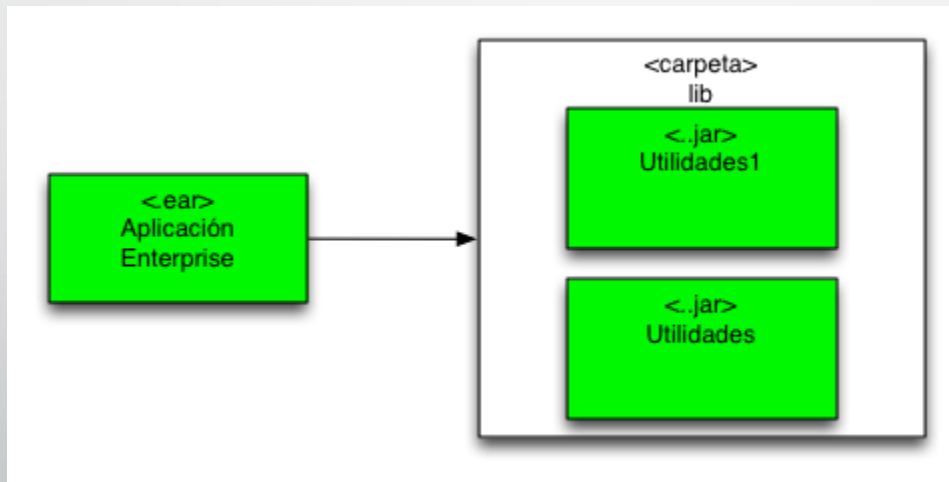
EAR -> ficheros JAR

- Aparte de albergar varias aplicaciones web también tiene la capacidad de contener EJBs que son empaquetados en archivos JAR para su utilización en las distintas aplicaciones web



EAR-> lib

- Aparte de estos ficheros disponemos de la carpeta lib para almacenar JARs que contenga clases que sean utilizadas o por los EJBs o por las aplicaciones web que tenemos empaquetadas.



EAR-> application.xml

- Los EAR disponen del fichero application.xml (directorio META-INF) que define la estructura de módulos incluidos en este tipo de empaquetamiento .Un ejemplo simplificado.

```
<?xml version=<em>"1.0"</em> encoding=<em>"UTF-8"</em>?>
<application xmlns:xsi.....>
<display-name>MiEAR</display-name>
<module>
<web>
<web-uri>Web01.war</web-uri>
<context-root>Web01</context-root>
</web>
</module>
<module>
<ejb>EJB.jar</ejb>
</module>
</application>
```