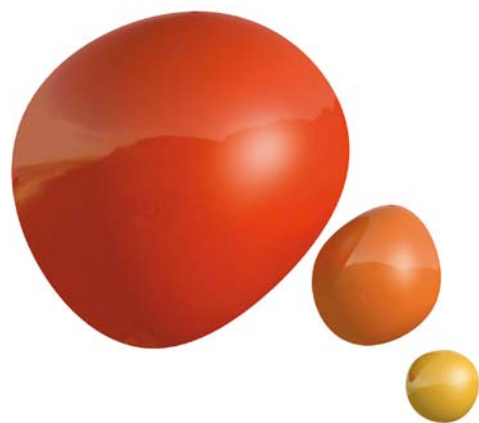




EL MODELO DE TRES CAPAS EN INTERNET



ÍNDICE

EL MODELO DE TRES CAPAS EN INTERNET

- 1. Características de las aplicaciones en tres capas 3
- 2. Los servidores de aplicaciones: Capa intermedia 5



El modelo de tres capas en internet

1. Características de las aplicaciones en tres capas

La arquitectura multicapa en dos capas

Es una técnica muy utilizada en aplicaciones empresariales. Consiste en partir nuestro software en capas. Básicamente hay dos arquitecturas: arquitectura en 2 capas y arquitectura en 3 capas.

Arquitectura en 2 capas

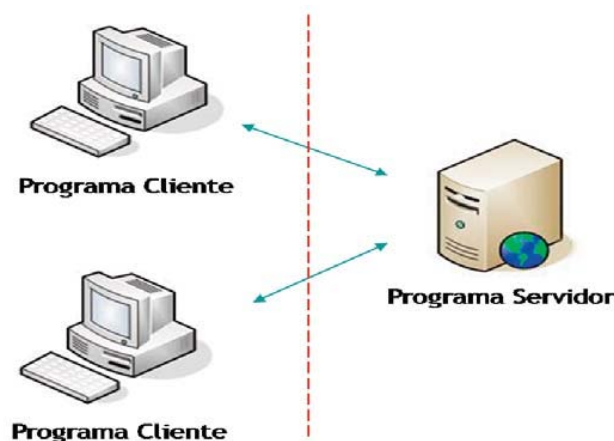
Consiste en **dividir** el programa en dos, uno el programa cliente y otro el programa servidor.

Se comunican a través de una red. La capa cliente va a tener una interfaz gráfica para el usuario y la capa servidor tiene la Base de Datos de la empresa.

El problema surge en la siguiente pregunta: ¿Dónde poner la lógica de negocio (**Bussiness Logic**)? Logica de negocio: algoritmos que resuelven los problemas propios de la empresa.

Se puede **poner en el cliente** o en el servidor.

Llevar parte de la lógica de negocio al servidor. Tendremos procedimientos almacenados pero se necesitan Bases de Datos grandes. La ventaja es que disminuye el tráfico por la red (enviamos datos ya calculados).



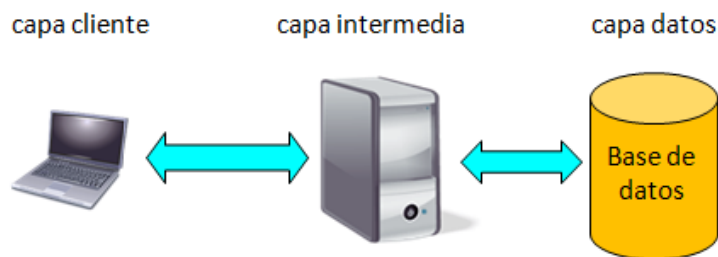
El modelo de tres capas en internet

La arquitectura multicapa en 3 capas

Es una arquitectura donde hay una separación física entre el cliente que solicita la información, los programas que la procesan y los datos sobre los que operan.

Vamos a tener 3 capas diferenciadas

Esta separación supone que el código de negocio es independiente de cómo y dónde se presenta. La capa de lógica de negocio no necesita saber qué tipo de cliente muestra los datos.



Este tipo de arquitectura es la que se utiliza en el desarrollo de aplicaciones Web .

Las aplicaciones para Web son un ejemplo de arquitectura de tres capas, donde los clientes basados en navegador, acceden a la capa intermedia para ejecutar la aplicación. Esta responde al cliente mediante el envío de una página Web con los datos que tiene que presentar.

La comunicación entre el navegador y la capa intermedia se realiza a través del protocolo HTTP. El navegador realiza peticiones HTTP al software de la capa intermedia (normalmente, un servidor Web) y, tras procesar la petición, se le envía al cliente una respuesta HTTP con la información que tiene que presentar en pantalla en formato HTML.

Los monitores de transacciones

Los **monitores** de transacciones fueron los primeros sistemas de 3 capas que se inventaron.

- CICS (IBM)
- + Tuxedo (Bea Logic)

Como capa de presentación usaban un terminal de pantalla verde.

En la capa de negocio tenían lenguajes de programación como COBOL, PL/1, RPG y en la capa de datos una

El modelo de tres capas en internet

base de datos como DB2 de IBM por ejemplo.

Eran muy estables y aún se usan en sistemas críticos, sistemas que no se puede permitir que fallen (Bancos). por ejemplo asignar el nivel de recursos de la aplicación, transacciones, etc...

Después con la aparición de los sistemas de objetos distribuidos se empezó a utilizar CORBA para implementar la lógica de negocio. Es lo mismo pero orientado a objetos, no tiene todos los servicios de un monitor de transacciones por lo que el programador tiene que implementar todos sus servicios a mano.

2. Los servidores de aplicaciones: Capa intermedia

Los servidores de aplicaciones. Capa intermedia

Son una arquitectura sobre la que se ejecuta la lógica de aplicación. Siempre sobre el modelo de 3 capas. En los servidores de aplicaciones se instalan los "componentes de negocio".

- Son objetos que implementan la funcionalidad de la lógica de negocio.
- Los llaman componentes porque se busca la reutilización y porque hay compañías que los fabrican y otras que los usan.
- Permite el desarrollo rápido de aplicaciones.
- En el caso de Java se les denomina EJB (Enterprise Java Beans).

Además de los componentes de negocio, en el servidor de aplicaciones se ejecutan otro tipo de componentes encargados de la interacción con la capa cliente, es decir, de la recogida de datos y la generación de respuestas.

En el caso de Java EE, estos componentes son los servlets y páginas JSP.

Herramientas para trabajar un componente de negocio

Para trabajar con un componente de negocio vamos a necesitar dos herramientas:

1. Herramientas para desarrollar y depurar componentes

- Netbeans
- Eclipse
- JBuilder
- Rational Application Server



El modelo de tres capas en internet

2. Un contenedor de componentes de negocio

Java dispone de varios servidores de aplicaciones:

- WebSphere de IBM
- WebLogic de Bea Systems
- JBoss de Jboss
- Glassfish de Sun
- Tomcat de apache (servidor limitado, pues no permite la ejecución de EJBs)

Para poder comunicar mediante HTTP con el cliente

Es importante diferenciar entre Java Beans y EJB. Los dos son componentes Java. Los Java Bean tienen unas reglas para la fase de desarrollo y los EJB siguen una arquitectura tanto para el desarrollo como para la distribución.

Tecnologías de servidores de aplicaciones actuales: Microsoft DNA

Microsoft DNA es el servidor de aplicaciones de Microsoft.

Distributed Internet Applications 1996

Es una solución propietaria. Como sistema operativo usa Windows NT, como tecnología de componentes, usa DCOM.

Como lenguajes utiliza Visual Basic y Visual C. Tiene un sistema de comunicación entre componentes mediante **mensajes asíncronos**, que lo que proporciona es escalabilidad al sistema, incluir los mensajes en colas.

Permite hacer un cluster de servidores. Para esto utilizamos Microsoft WolfPack. Como base de datos utiliza SQL Server y como servidor web IIS (Internet Information Server).

Disponemos de la herramienta Microsoft Management Console que es una utilidad de instalación y administración, con menús tipo visual basic y podemos ir viendo las propiedades de los componentes.

Especificación SUN Java EE

Resuelve el problema de complejidad y coste del desarrollo de servicios multi- capa.

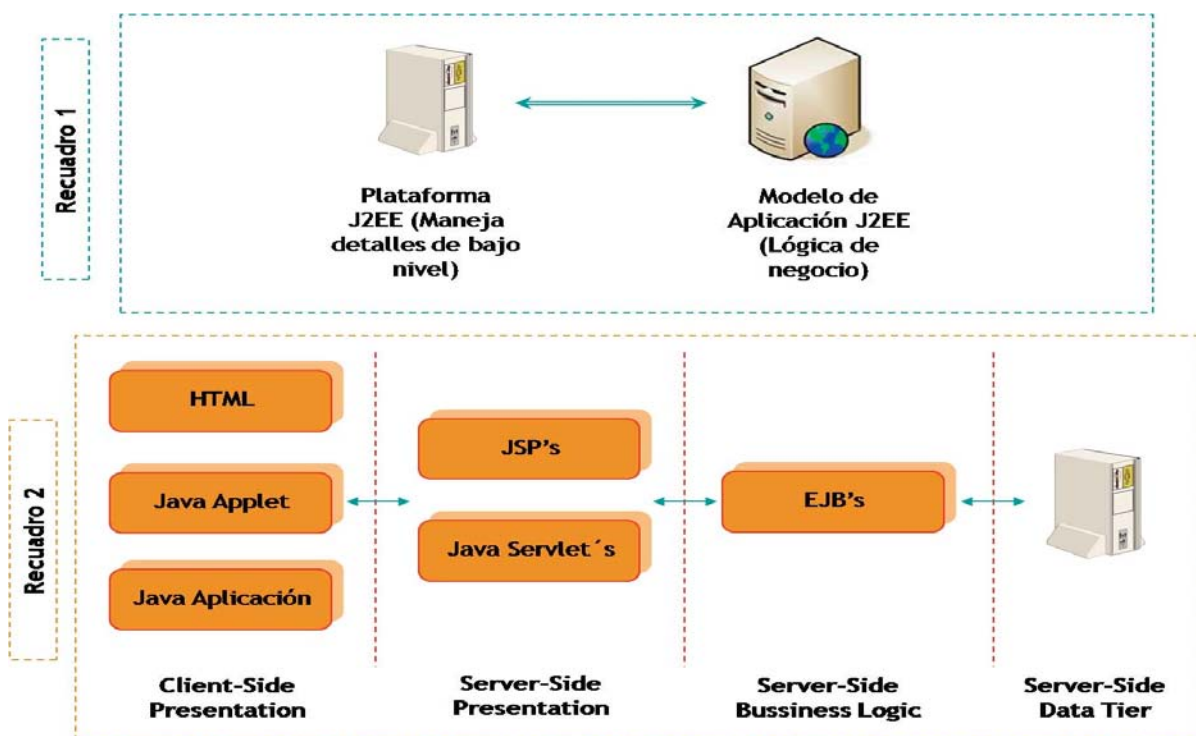
- Con Java EE podemos desarrollar sistemas seguros, eficientes y escalables.
- Java EE es una arquitectura de estándar abierto que ofrece lo que es la plataforma Java EE y el modelo de aplicación Java EE

El modelo de tres capas en internet

- Java EE comprende un conjunto de servicios, protocolos e interfaces de programación.
- Java EE es una arquitectura basada en componentes para programación multicapa.

En los servicios multicapa se requieren aplicaciones cliente, lógicas de negocio y presentación que obtienen, actualizan y presentan los datos además del código de infraestructura que son componentes de bajo nivel del sistema (acceden a recursos del sistema, seguridad, etc...).

El desarrollador sólo ha de preocuparse por la lógica de negocio y presentación



El modelo de tres capas en internet

Especificación Sun Java EE: ventajas

- Java EE proporciona soluciones Java entre todas las capas.
- Separación de tareas en la plataforma multi-capa.
- Conjunto de estándares abiertos (EJB, Servlets, JDBC, JNDI, RMI, etc...).
- Portabilidad (WORA, escribe una vez, ejecuta en cualquier parte).
- Escalabilidad (Arquitectura basada en componentes integrados).
- La lógica de presentación, la lógica de negocio y la lógica de acceso a datos están separadas en diferentes componentes y desplegada en varios servidores.
- La recolección de basura y el manejo automático de excepciones reduce el problema de que un componente bloquee la operación del servidor u otro componente.

Especificación Sun Java EE: componentes

- El control de transacciones y gestión del ciclo de vida de los componentes son funcionalidades definidas en la especificación Java EE y, por tanto, son proporcionadas por el servidor de aplicaciones.
- El modelo de aplicación de J2EE encapsula la funcionalidad de las capas en componentes de tipos específicos, por ejemplo la lógica de negocio está encapsulada en componentes EJB (Enterprise Java Beans).
- La iteración con los clientes puede representarse a través de:
 - Páginas HTML
 - Applets de Java
 - Java Servlets
 - Páginas JSP
 - Aplicaciones Java Solitarias
- Los componentes se comunican de forma transparente usando varios estándares como HTML, XML, http, RMI, IIOP, y otros.