

Sistemas Informáticos

Tema 1: Introducción a los sistemas distribuidos



Contenido

- Sistemas distribuidos. Definiciones.
- Modelos de sistemas distribuidos.
- Arquitectura de la WWW
- Características de los sistemas C/S
- Evolución de los sistemas distribuidos.



Sistemas distribuidos. Definiciones

- Sistemas centralizados:
 - Ordenador central y red de terminales sin capacidad de proceso.
- Sistemas distribuidos:
 - Un conjunto de elementos de proceso autónomos
 - no necesariamente homogéneos,
 - que están interconectados por una red de comunicaciones de cualquier tipo
 - y que cooperan para realizar las tareas que tienen asignadas.
- Atendiendo al grado de acoplamiento hay dos tipos de sistemas distribuidos:
 - Fuerte: Procesadores comparten memoria o buses de entrada/salida. Aplicaciones multiprocesadores.
 - Débil: Procesadores autónomos interconectados por sistemas de comunicaciones.
- En este curso se tratarán únicamente los segundos.



Sistemas Informáticos



Modelo Arquitectónico de la WWW



Arquitecturas de sistemas distribuidos

Existen dos arquitecturas básicas de sistemas distribuidos:

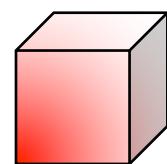
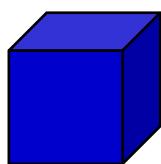
- Cliente / Servidor:
 - Sistema asimétrico.
 - Procesos clientes solicitan servicios.
 - Procesos servidores los ejecutan y devuelven los resultados.
- Igual a igual (*peer to peer, p2p*)
 - Sistema simétrico.
 - Todos los procesos desempeñan tareas semejantes.
 - Interactúan para realizar una actividad distribuida.

Pero antes...



¿Qué es la Arquitectura de Software?

- La *arquitectura de un sistema de software* define al sistema en términos de componentes computacionales e interacciones entre esos componentes



Estructura y Topología



Arquitectura de Software

- Considera:
 1. La ubicación de cada componente en una red de ordenadores.
 2. Interrelación de los componentes
 3. Determinar la distribución de datos y tareas computacionales entre los nodos físicos de la red.
 4. Evaluar el rendimiento, confiabilidad, escalabilidad y otras propiedades del sistema de software.



Arquitectura de Software (AdS)

- Componentes:
 - Clientes
 - servidores
 - bases de datos
 - Etc.
- Interacciones:
 - llamadas a procedimientos
 - compartición de variables
 - protocolos cliente servidor
 - protocolos de acceso a BB.DD.
 - Etc.



¿Por qué es importante la AdS?

- Forma la columna vertebral para construir un sistema de software.
- Es en gran medida responsable de permitir o no ciertos atributos de calidad del sistema (p.e. confiabilidad y rendimiento).
- Es un modelo abstracto reutilizable
 - Puede transferirse de un sistema a otro.
 - Representa una inversión.
- Representa un medio de comunicación y discusión entre participantes del proyecto.



Arquitecturas de sistemas distribuidos

Existen dos arquitecturas básicas de sistemas distribuidos:

- Cliente / Servidor:
 - Sistema asimétrico.
 - Procesos clientes solicitan servicios.
 - Procesos servidores los ejecutan y devuelven los resultados.
- Igual a igual (*peer to peer, p2p*)
 - Sistema simétrico.
 - Todos los procesos desempeñan tareas semejantes.
 - Interactúan para realizar una actividad distribuida.



Volvemos... 

Cliente-Servidor

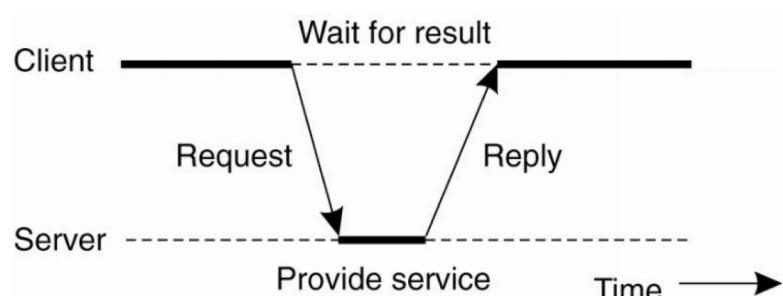
- Objetivo: proveer arquitectura escalable
 - Cada ordenador o proceso en la red es o bien cliente o bien servidor.
- Servidor:
 - Pasivo (esclavo).
 - Espera peticiones.
 - Cuando recibe una petición, la procesa y envía respuesta.
 - Puede conservar o no el estado de la comunicación.
- Cliente:
 - Activo (maestro).
 - Envía peticiones.
 - Espera hasta que llega la respuesta.



Cliente-Servidor

Client-server model:

Known for more than 25 years, very popular in DS design



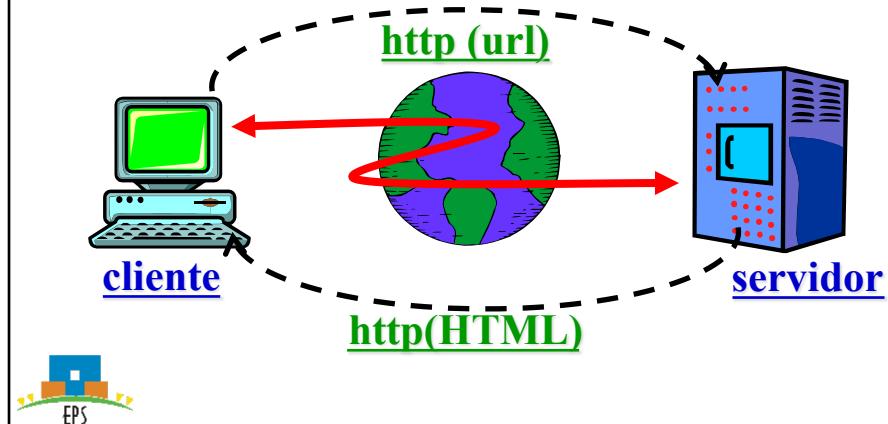
General interaction between a client and a server.



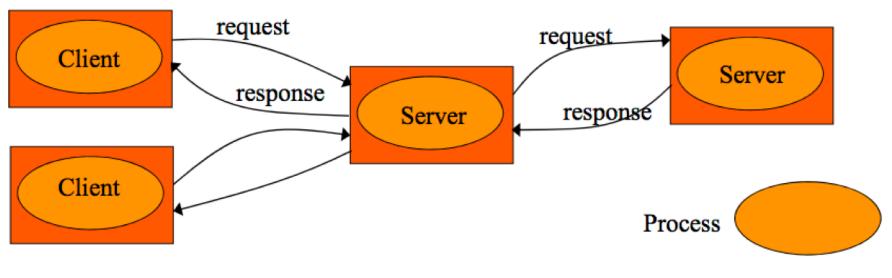
Cliente-Servidor

- **Ejemplo: WWW**

- El cliente realiza una petición utilizando el protocolo *http*.
- El servidor responde con un documento codificado en *html*.



Cliente-Servidor: vista de componentes



Arquitectura Cliente / Servidor. Características

- Arquitectura más extendida de implementación de sistemas distribuidos.
- Sistemas de Información (SI) distribuidos basados en las siguientes características:
 - **Servicio:** unidad básica de diseño. El servidor los proporciona y el cliente los utiliza.
 - **Recursos compartidos.** Muchos clientes utilizan los mismos servidores y, a través de ellos, comparten recursos lógicos o físicos.
 - **Protocolos de aplicación asimétricos.** Los clientes inician conversaciones. Servidores esperan su establecimiento pasivamente. Relación 1 - n.
 - **Transparencia:** El sistema aparece como una única unidad de proceso. Independencia de la plataforma HW y SW que se emplee.



Arquitectura Cliente / Servidor. Características (II)

- Características (continuación):
 - Sistemas **débilmente acoplados**. Interacción basada en envío de mensajes.
 - Encapsulamiento de servicios. Los detalles de la implementación de un servicio son **transparentes al cliente**.
 - **Escalabilidad** horizontal (añadir clientes o servidores) y vertical (ampliar potencia de servidores).
 - Datos y programas centralizados en servidores facilitan su **integridad y mantenimiento**.



Cliente-Servidor

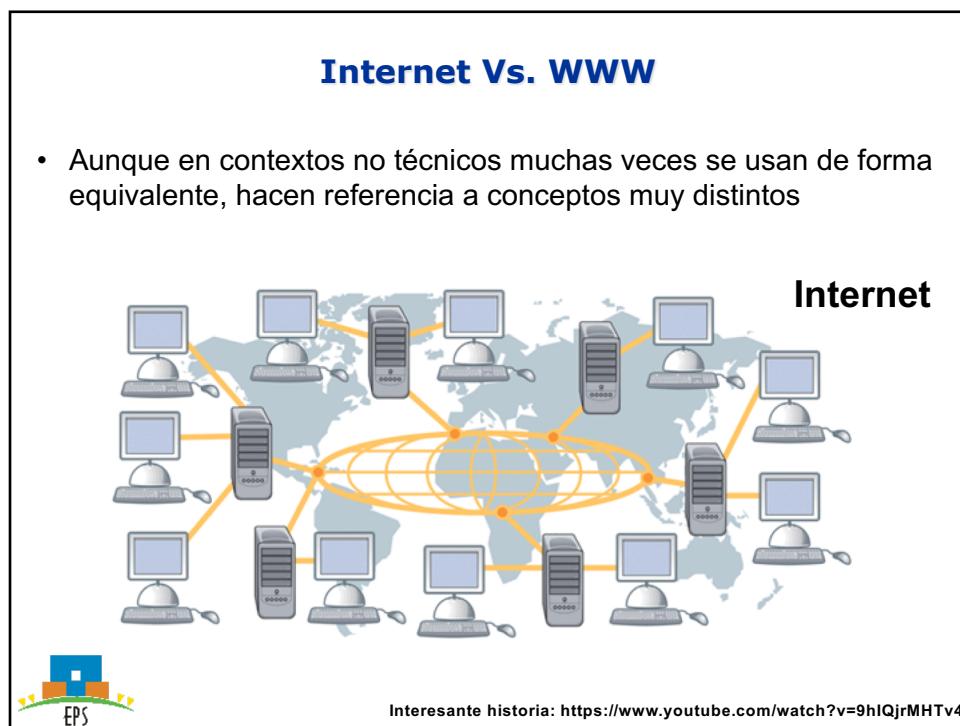
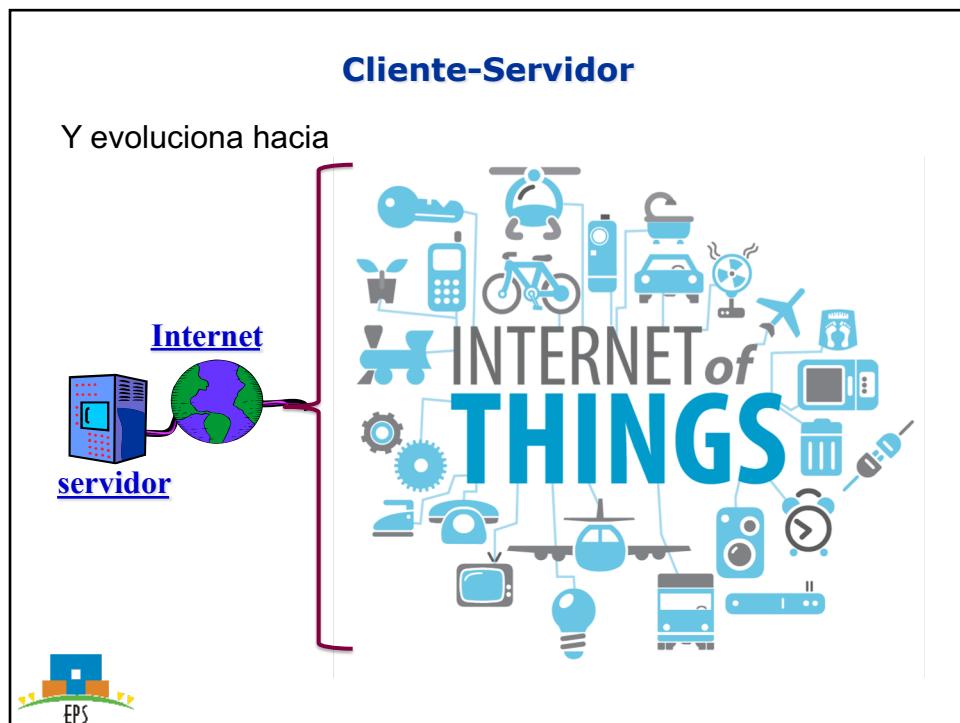
- ¿Escalable?

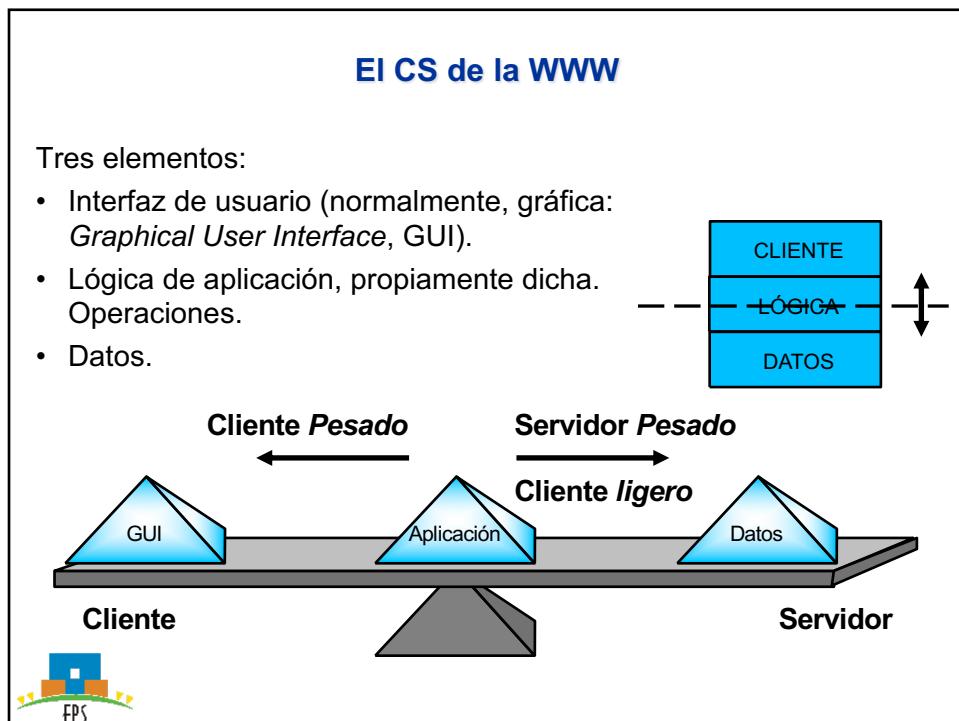
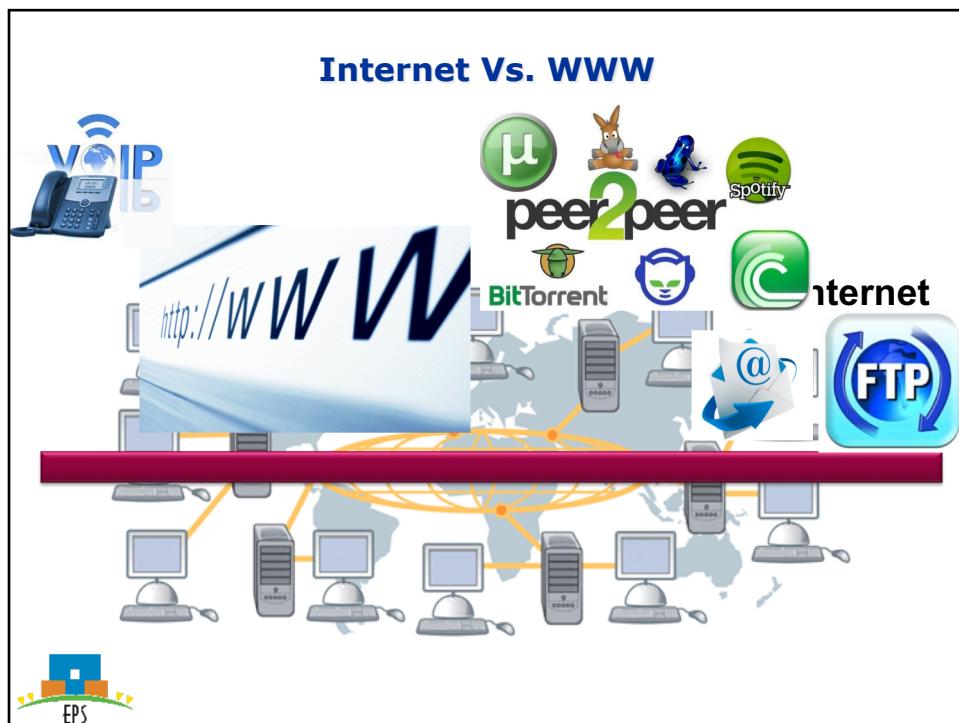
The diagram illustrates a client-server architecture. At the center is a blue server tower labeled "servidor". A purple line labeled "Internet" connects it to a green globe icon. From the globe, several purple lines branch out to various client devices: three desktop computers (one with Windows logo, one with Linux logo, one with Mac logo), a laptop, a smartphone, and a mobile phone. Below the diagram is a small logo consisting of three colored squares (blue, orange, yellow) with the letters "EPS" underneath.

Cliente-Servidor

- ¿Escalable?

This diagram is similar to the one above, showing a server tower connected via the Internet to a globe, which then connects to various clients. In addition to the previous clients, it includes a tablet, a smartwatch, an Android smartphone, and a modern refrigerator. Below the diagram is a small logo consisting of three colored squares (blue, orange, yellow) with the letters "EPS" underneath. At the bottom, there is text providing statistics: "Total servidores WWW (agosto 2015): [1.814,996,408](#)" and "Total dispositivos conectados : **¿~ 1200 Millones ?**". A source link "Fuente: <http://news.netcraft.com/>" is also present.

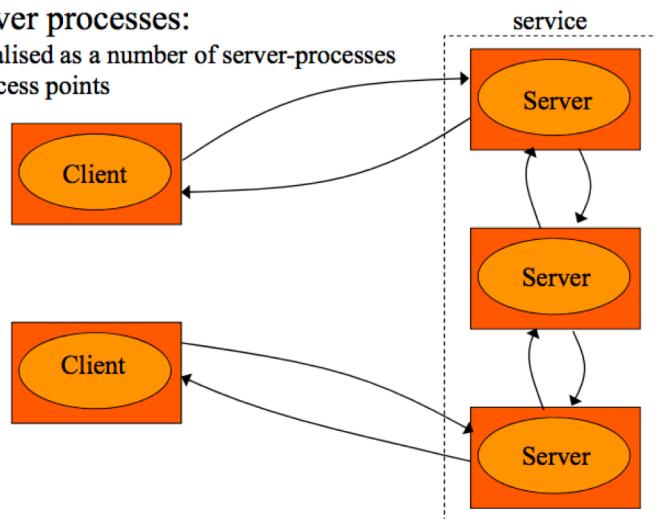




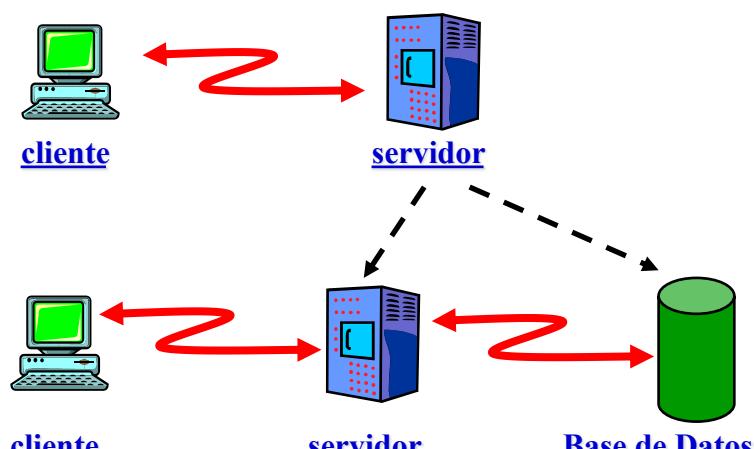
Variaciones del modelo C/S

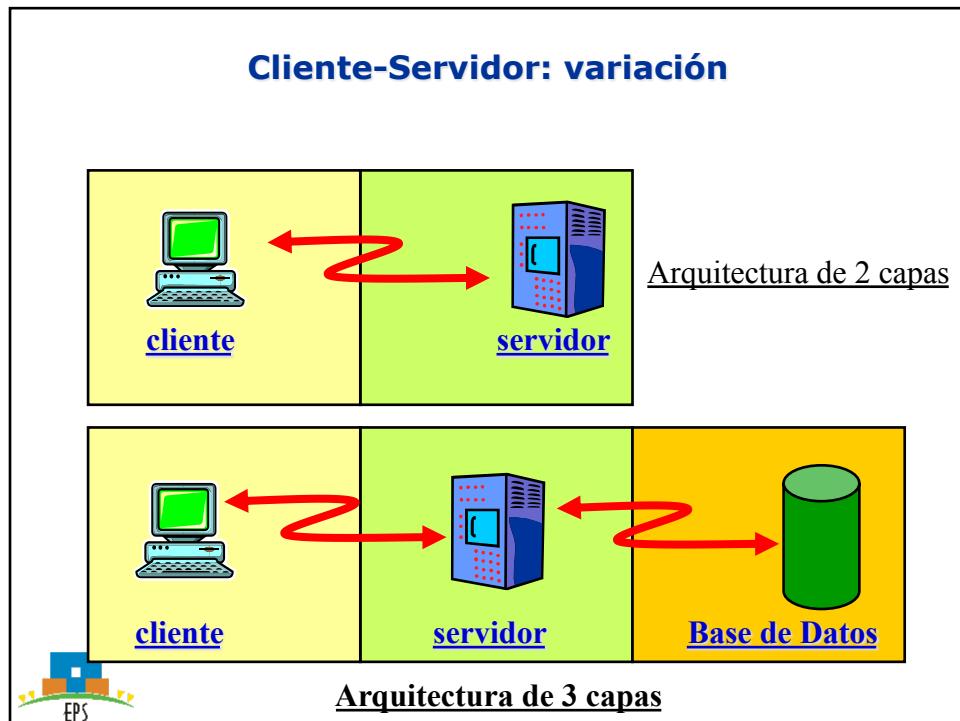
Multiple server processes:

- service realised as a number of server-processes
- several access points



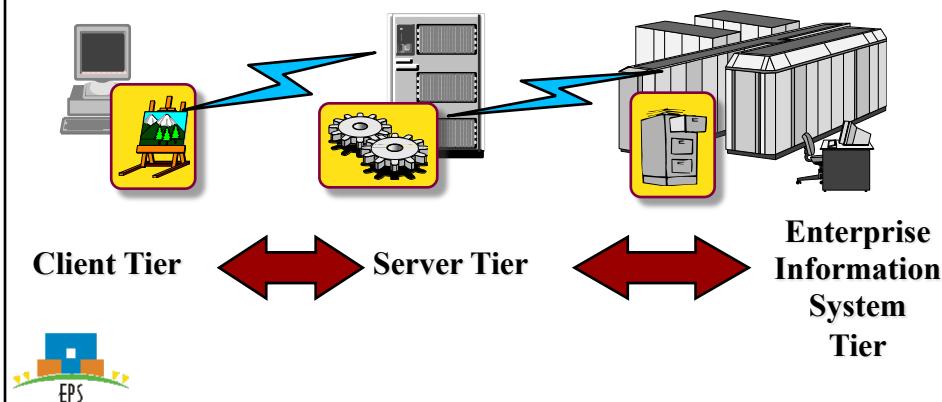
Cliente-Servidor: variación





Aplicaciones Cliente / Servidor de 3 niveles (3-tier)

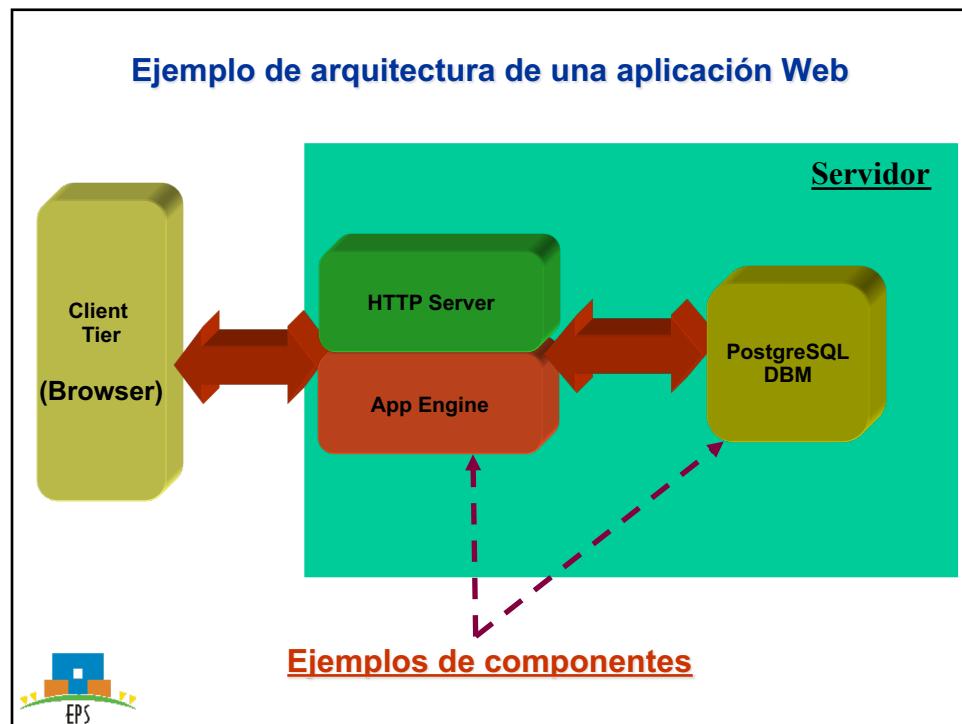
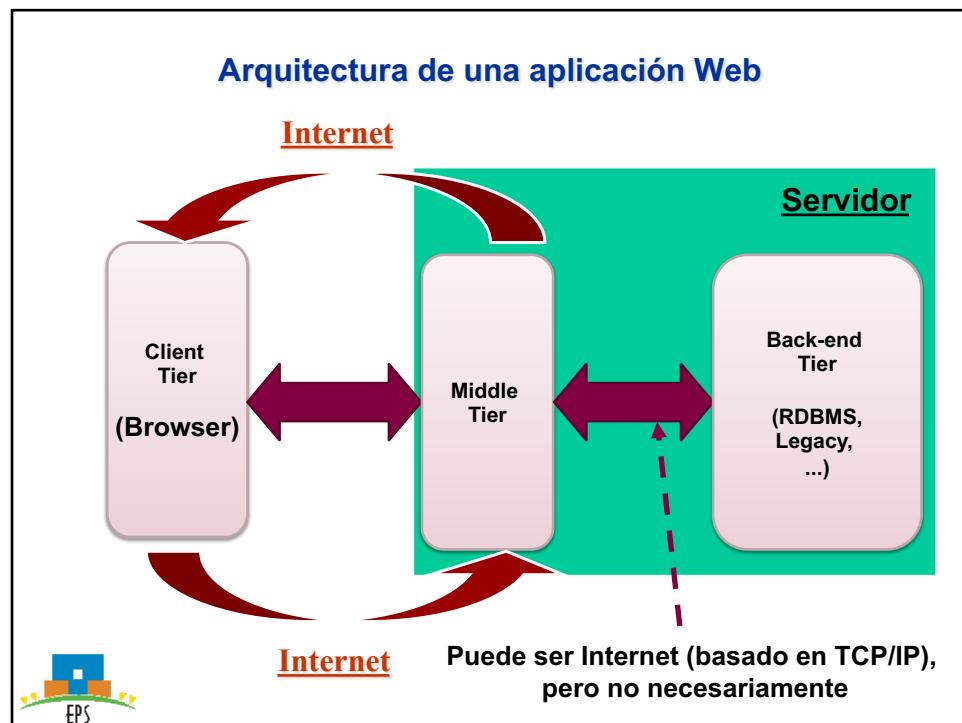
- Lógica de aplicación y de presentación separada (**Modelo Cliente Ligero**).
- La lógica de aplicación reside en cualquier sistema de la red.
- Aplicaciones más robustas y fácilmente escalables.

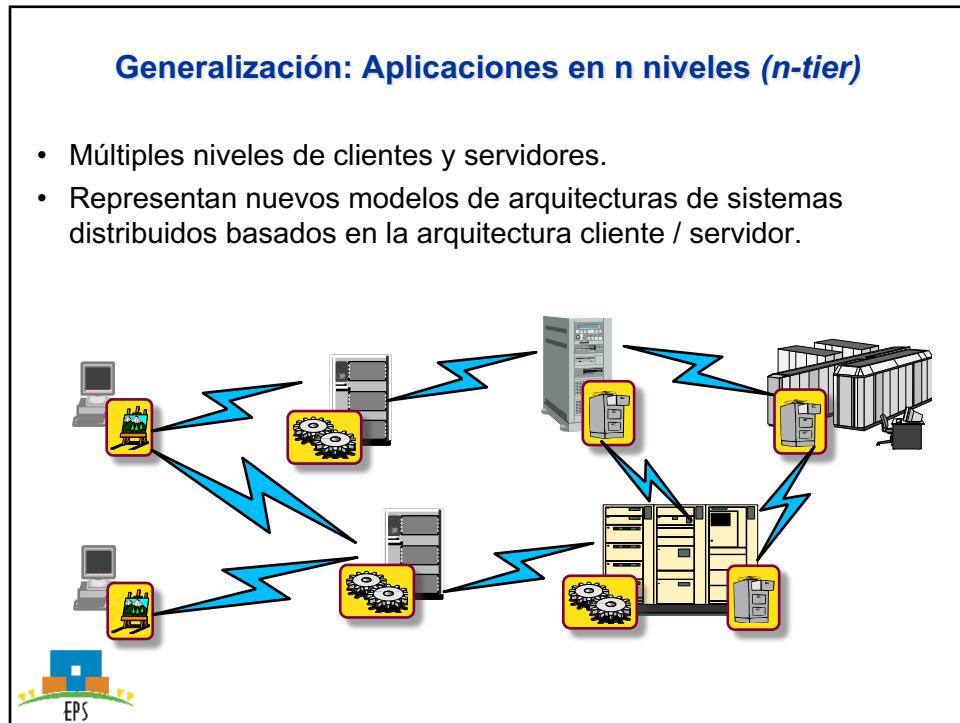
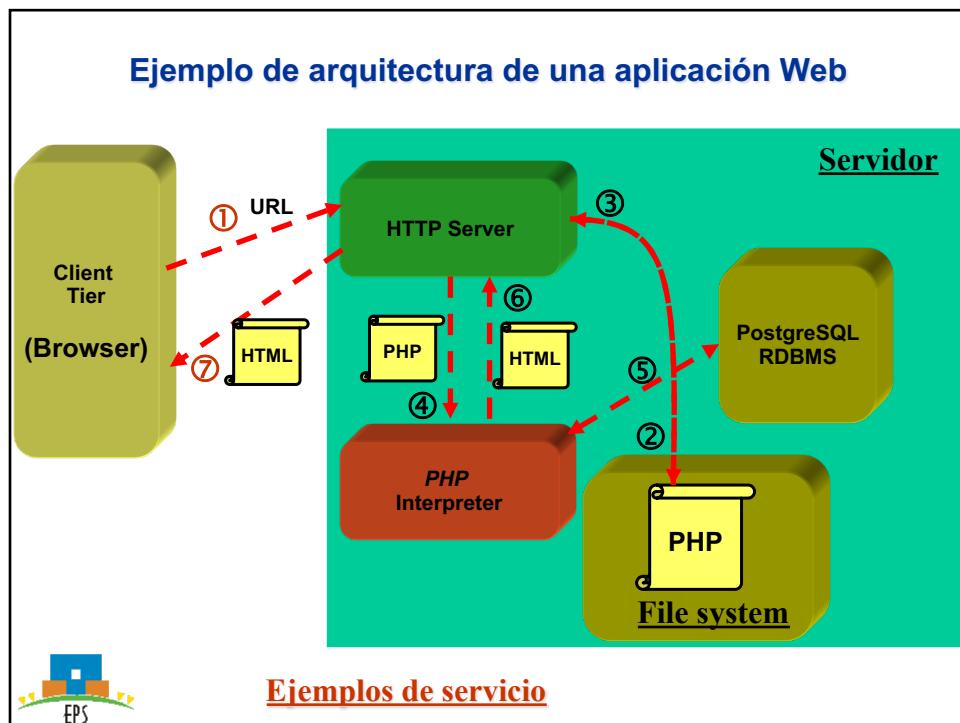


Sistemas distribuidos basados en la World Wide Web

- Sistemas distribuidos, basado en la extensión del modelo de cliente ligero:
- Clientes:
 - Universales: visualizadores Web (*Web Browsers*).
 - Específicos: programas ejecutados en el cliente (antiguamente *Applets*. Actualmente algunas aplicaciones para smartphones).
- Servidores Web + Servidor de Aplicaciones:
 - Repositorios de documentos.
 - Entorno de ejecución de aplicaciones o *Servidor de Aplicaciones*.
- Servidores de Back-End:
 - Enlace con programas ya existentes o desarrollos antiguos (*legacy systems*).
 - Programas en el servidor Web realizan consultas a ellos.
 - A través de algún tipo de *middleware* más o menos elaborado.
 - Los casos más comunes son:
 - Servidores de **bases de datos**
 - Servidores de **proceso de transacciones**.

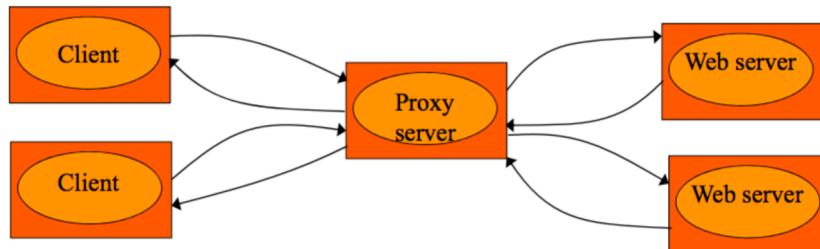






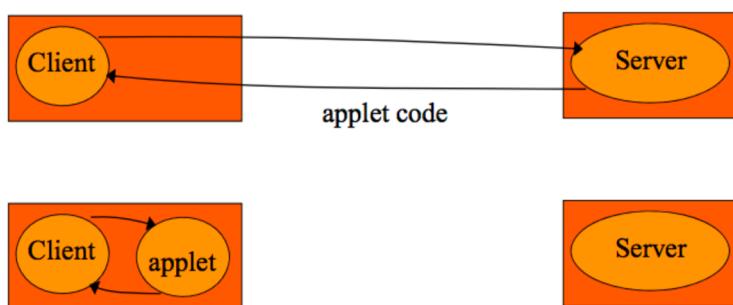
Otras variantes del C/S: Proxy

- Caché: almacena datos usados recientemente más cerca que los datos originales (red local, por ej)
- Servidor Proxy: caché compartida entre varios clientes (los clientes de la red local, por ej).



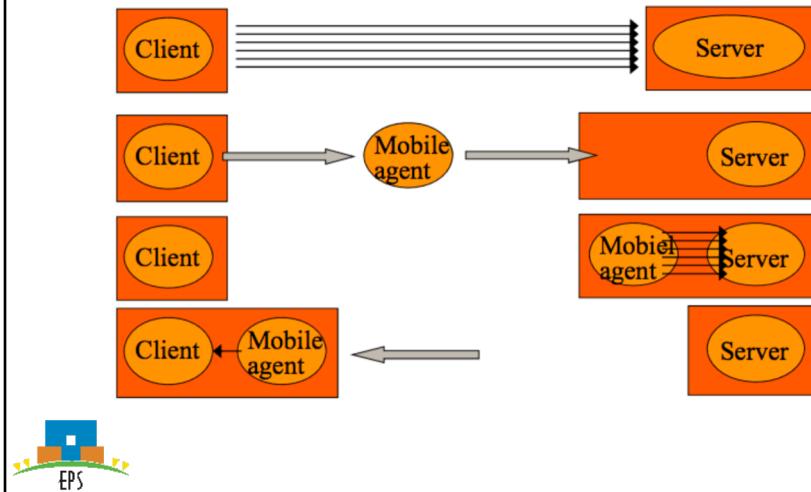
Otras variantes del C/S: Código Móvil

- Ej applets: permiten implementar, entre otras cosas, “modelos push” (el servidor parece invocar al cliente) y GUI más avanzadas.



Otras variantes del C/S: Agentes móviles

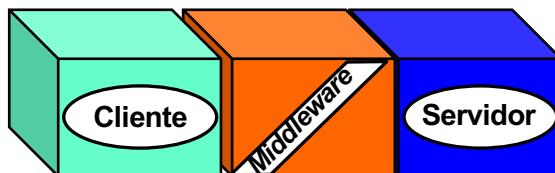
- Programas (código y datos) migran entre ordenadores, y ejecutan una tarea en representación de alguien



Componentes de un sistema Cliente / Servidor

Tres módulos:

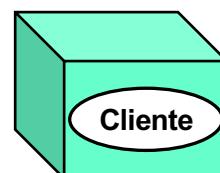
- Cliente. Sigue servicios.
- Middleware. Enlace entre Cliente y Servidor o entre Servidores.
- Servidor. Proporciona servicios.



Clients

Cliente es todo proceso que reclama servicios a otro.

- No es equivalente a aplicación que interacciona con el usuario, aunque es el modelo de cliente más corriente.
- Un servidor, a su vez, puede ser cliente de otros servidores.
- La funcionalidad del proceso cliente marca la operativa de la aplicación (flujo de información o lógica de negocio).



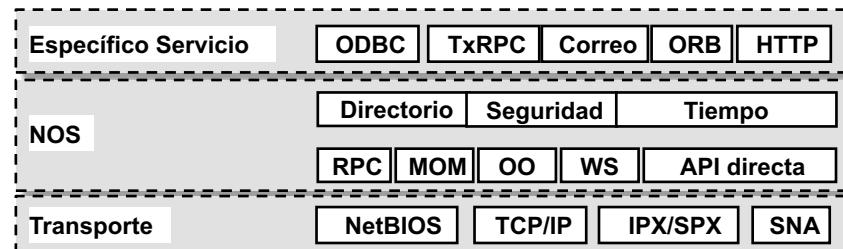
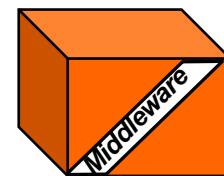
Servidores

- Servidor es todo proceso que proporcione un servicio a otros.
 - Servidores de disco.
 - Servidores de impresión.
 - Servidores de comunicaciones.
 - Servidores de presentación.
 - Servidores de procesos.
 - Servidores de bases de datos.
 - ...
- Ambigüedad: Normalmente se llama servidores tanto a las aplicaciones o procesos que realizan un servicio como a los ordenadores que les sirven de soporte. En esta asignatura normalmente nos referiremos a los procesos de servicio.



Middleware

- Se ejecuta en el servidor y el cliente.
- Interfaz Cliente / Servidor o Servidor / Servidor.
- Tres niveles:
 - Protocolo específico del servicio.
 - *Network Operating System, NOS*.
 - Protocolo de transporte.



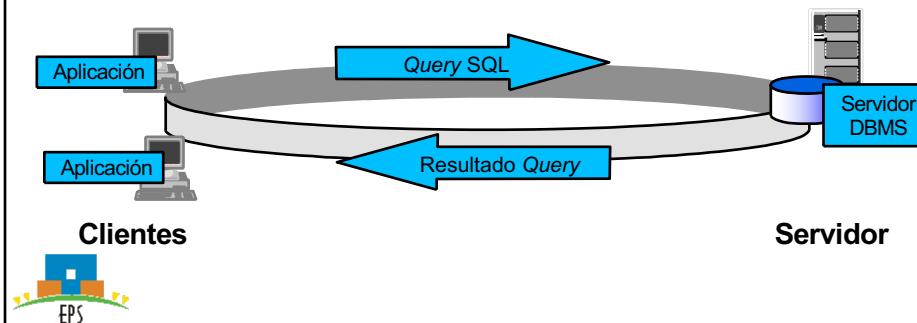
Modelos de sistemas distribuidos

- Red de compartición de recursos.
- **Servidores de bases de datos.**
- **Proceso de transacciones.**
- Sistemas de soporte de trabajo en grupo.
- Sistemas de objetos distribuidos.
- **Servicios Web.**
- **Sistemas distribuidos con clientes basados en la WWW.**



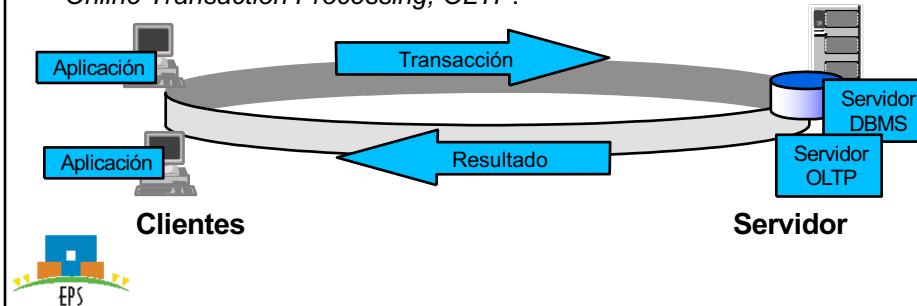
Servidores de base de datos

- El cliente pasa una petición (*query*) SQL en un mensajes al servidor (*Data Base Management Server, DBMS*).
- Los resultados de cada consulta SQL se devuelven por la red.
- El código que ejecuta la petición SQL se encuentra en el mismo ordenador que los datos.



Proceso de transacciones

- El cliente solicita la ejecución de un procedimiento remoto en el servidor.
- El procedimiento ejecuta un grupo de peticiones SQL o de otro tipo (transacción).
- Todas las peticiones se ejecutan o fallan como una unidad.
- El resultado final se devuelve al cliente.
- La aplicación se desarrolla escribiendo el cliente y el código de las transacciones en el servidor.
- *Online Transaction Processing, OLTP*.



Servicios Web

- *Middleware* con modelo de llamadas a procedimientos remotos sobre WWW.
 - Publicación de servicios por parte de los servidores.
 - Descubrimiento dinámico de servicios por los clientes.
 - Ejecución de los servicios por los clientes siguiendo un modelo RPC.
- Modelos de ejecución punto a punto o pasando a través de un intermediario (*broker*) único: *Enterprise Service Bus*.
- Base para la *Services Oriented Architecture (SOA)*.

The diagram illustrates the Web Services architecture. At the top, a **Service Broker** box is connected to a **Service Provider** box via a line labeled "Publish (WSDL)" and to a **Service Requester** box via a line labeled "Find (UDDI)". A horizontal line connects the **Service Provider** and **Service Requester** boxes, labeled "Bind (SOAP)". Below this central structure is a network graph consisting of several colored nodes (green, pink, blue, yellow, red) interconnected by lines. To the right is a diagram of an **Enterprise Service Bus (ESB)**, represented by a black rectangle labeled "ESB" with several colored nodes connected to it.



Redes “espontáneas”

- Los usuarios portan dispositivos móviles (portátil, smartphone, etc) entre distintas redes (casa, trabajo, transporte público, aeropuerto, hotel, bar, ...) y puede acceder a servicios locales o remotos en itinerancia, usando 3(4)G, WiFi, ...

The diagram shows a central circular area labeled "Hotel's wireless network". Several devices are connected to it: a **TV/PC** (solid line), a **Laptop** (dashed line), a **PDA** (dotted line), and a **Mobile device of guest** (dash-dot line). Outside the central network, there is a **Discovery service** box with a line labeled "register lookup" pointing towards the central network. To the left, an **Internet gateway** is connected to the central network. Above the gateway is a cloud-like shape labeled "Internet". Other service boxes include **Print/fax service**, **Music service**, and **Alarm service**. The entire setup is labeled "(ubiquitous computing)" at the top.

