

# SISTEMAS DISTRIBUIDOS

Instituto Tecnológico superior de Mulege  
Tecnología de la información y la comunicación.

Oscar Manuel Medellín Meza

Samir Arturo Navarro Meza

Sistema Operativo II

## Contenido

Contenido.....	3
Características .....	3
Evolución .....	3
Cliente Servidor.- .....	4
CLIENTE-SERVIDOR.....	4
Categorías de Servidores: .....	4
Componentes de Software: .....	5
Arquitectura Cliente-Servidor de Tres Capas- .....	5
Clasificación de los sistemas cliente servidor:.....	6
Base de datos distribuida .....	8
Desarrollo web .....	9
Ventajas de los sistemas distribuidos .....	9
Desventajas de los sistemas distribuidos .....	9

## Indice de ilustraciones

Ilustración 1-presentación distribuida .....	6
Ilustración 2-presentación remota.....	6
Ilustración 3- Logica Distribuida .....	7
Ilustración 4-Gestión remota de datos .....	7
Ilustración 5-Base de datos distribuidas .....	7
Ilustración 6-cliente servidor a tres niveles .....	8

## Contenido

"Sistemas cuyos componentes hardware y software, que están en computadoras conectadas en red, se comunican y coordinan sus acciones mediante el paso de mensajes, para el logro de un objetivo. Se establece la comunicación mediante un protocolo preestablecido".

### Características

- Concurrencia.- Esta característica de los sistemas distribuidos permite que los recursos disponibles en la red puedan ser utilizados simultáneamente por los usuarios y/o agentes que interactúan en la red.
- Carencia de reloj global.- Las coordinaciones para la transferencia de mensajes entre los diferentes componentes para la realización de una tarea, no tienen una temporización general, está más bien distribuida en los componentes.
- Fallos independientes de los componentes.- Cada componente del sistema pudiera fallar de manera independientemente, y los demás continuar ejecutando sus acciones. Esto permite el logro de las tareas con mayor efectividad, pues el sistema en su conjunto continua trabajando.

### Evolución

Procesamiento central (Host).- Refiere a uno de los primeros modelos de computadoras interconectadas, llamados centralizados, donde todo el procesamiento de la organización se llevaba a cabo en una sola computadora, normalmente un Mainframe, y los usuarios empleaban sencillas computadoras personales.

Algunos problemas de este modelo son:

- Cuando la carga de procesamiento aumentaba se tenía que cambiar el hardware del Mainframe, lo cual es más costoso que añadir más computadores personales clientes o servidores que aumenten las capacidades.
- El otro problema que surgió son las modernas interfases gráficas de usuario, las cuales podían conllevar a un gran aumento de tráfico en los medios de comunicación y por consiguiente podían colapsar a los sistemas.

Grupo de Servidores.- Otro modelo que entró a competir con el anterior, también un tanto centralizado, son un grupo de computadoras actuando como servidores, normalmente de archivos o de impresión, poco inteligentes para un número de minicomputadores que hacen el procesamiento conectados a una red de área local.

Algunos problemas de este modelo son:

- Podría generarse una saturación de los medios de comunicación entre los servidores poco inteligentes y los minicomputadores, por ejemplo cuando se solicitan archivos grandes por varios clientes a la vez, podían disminuir en gran medida la velocidad de transmisión de información. La Computación

Cliente Servidor.- Este modelo, que predomina en la actualidad, permite descentralizar el procesamiento y recursos, sobre todo, de cada uno de los servicios y de la visualización de la Interfaz Gráfica de Usuario. Esto hace que ciertos servidores estén dedicados sólo a una aplicación determinada y por lo tanto ejecutarla en forma eficiente.

### CLIENTE-SERVIDOR

Sistema en donde el cliente es una máquina que solicita un determinado servicio y se denomina servidor a la máquina que lo proporciona. Los servicios pueden ser:

- Ejecución de un determinado programa.
- Acceso a un determinado banco de información.
- Acceso a un dispositivo de hardware.

La presencia de un medio físico de comunicación entre las máquinas, es un elemento primordial, y dependerá de la naturaleza de este medio la viabilidad del sistema.

### Categorías de Servidores:

A continuación se presenta una lista de los servidores más comunes:

- Servidores de archivos.- Proporciona archivos para clientes. Si los archivos no fueran tan grandes y los usuarios que comparten esos archivos no fueran muchos, esto sería una gran opción de almacenamiento y procesamiento de archivos. El cliente solicita los archivos y el servidor los ubica y se los envía.
- Servidores de Base de Datos.- Son los que almacenan gran cantidad de datos estructurados, se diferencian de los de archivos pues la información que se envía está ya resumida en la base de datos. Ejemplo: El Cliente hace una consulta, el servidor recibe esa consulta (SQL) y extrae sólo la información pertinente y envía esa respuesta al cliente.
- Servidores de Software de Grupo.- El software de grupo es aquel, que permite organizar el trabajo de un grupo. El servidor gestiona los datos que dan soporte a estas tareas. Por ejemplo: almacenar las listas de correo electrónico. El Cliente puede indicarle, que se ha terminado una tarea y el servidor se lo envía al resto del grupo.
- Servidores WEB.- Son los que guardan y proporcionan páginas HTML. El cliente desde un browser o navegador hace un llamado de una página

(link) y el servidor recibe el mensaje para después enviar la página solicitada.

- Servidores de correo.- Gestiona el envío y recepción de correo de un grupo de usuarios (el servidor no necesita ser muy potente). El servidor sólo debe utilizar un protocolo de correo.
- Servidor de objetos.- Permite almacenar objetos que pueden ser activados de manera remota. Los clientes pueden ser capaces de activar los objetos que se encuentren en el servidor.
- Servidores de impresión.- Gestionan las solicitudes de impresión de los clientes. El cliente envía la solicitud de impresión, el servidor recibe la solicitud y la ubica en la cola de impresión, ordena a la impresora que lleve a cabo las operaciones y luego avisa a la computadora cliente que ya acabo su respectiva impresión.
- Servidores de aplicación.- En el pasado refería a un servidor que se dedicaba a una única aplicación. Era básicamente una aplicación a la que podían acceder los clientes. En la actualidad refiere más a un servidor Web con capacidad de procesamiento, por lo que suele ser a la vez servidor Web con algunas funciones de lógica de negocio.

Componentes de Software:

Se distinguen tres componentes básicos de software:

- Presentación.- Tiene que ver con la presentación al usuario de un conjunto de objetos visuales y llevar a cabo el procesamiento de los datos producidos por el mismo y los devueltos por el servidor.
- Lógica de aplicación.- Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- Base de datos.- Esta compuesta de los archivos que contienen los datos de la aplicación. Arquitecturas Cliente / Servidor A continuación mostramos las arquitecturas cliente-servidor más populares:
- Arquitectura Cliente-Servidor de Dos Capas.- Consiste en una capa de presentación y lógica de la aplicación; y la otra de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones: o Cuando se requiera poco procesamiento de datos en la organización. o Cuando se tiene una base de datos centralizada en un solo servidor. o Cuando la base de datos es relativamente estática. o Cuando se requiere un mantenimiento mínimo.

Arquitectura Cliente-Servidor de Tres Capas-

Consiste en una capa de la Presentación, otra capa de la lógica de la aplicación y otra capa de la base de datos. Normalmente esta arquitectura se utiliza en las siguientes situaciones:

- Cuando se requiera mucho procesamiento de datos en la aplicación.

- En aplicaciones donde la funcionalidad este en constante cambio.
- Cuando los procesos no están relativamente muy relacionados con los datos.
- Cuando se requiera aislar la tecnología de la base de datos para que sea fácil de cambiar.
- Cuando se requiera separar el código del cliente para que se facilite el mantenimiento.
- Esta muy adecuada para utilizarla con la tecnología orientada a objetos.

Clasificación de los sistemas cliente servidor:

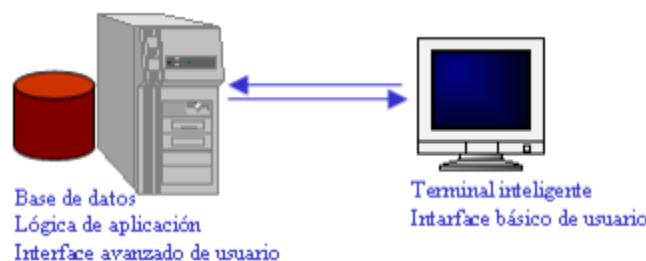
A continuación mostramos la clasificación de de los sistemas cliente/servidor de acuerdo al nivel de abstracción del servicio que ofrecen:

1. Representación distribuida.- La interacción con el usuario se realiza en el servidor, el cliente hace de pasarela entre el usuario y el servidor.



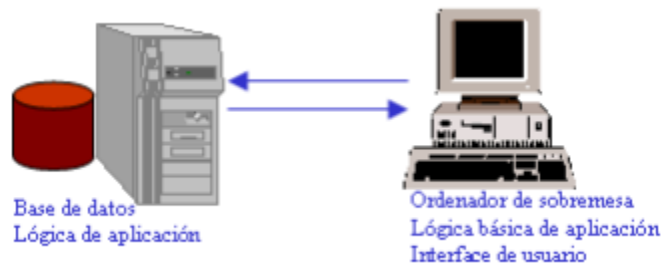
*Ilustración 1-presentación distribuida*

2. Representación Remota.-La lógica de la aplicación y la base de datos se encuentran en el servidor. El cliente recibe y formatea los datos para interactuar con el usuario.

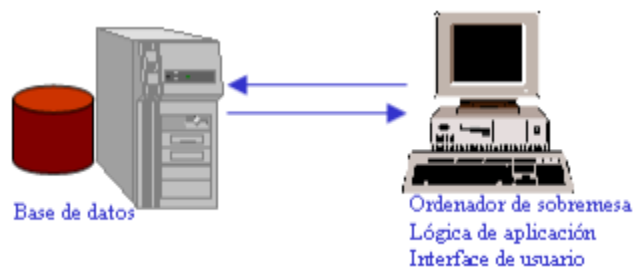


*Ilustración 2-presentación remota*

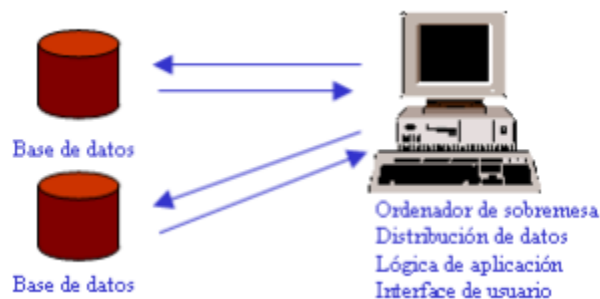
3. Lógica Distribuida.- El cliente se encarga de la interacción con el usuario y de algunas funciones triviales de la aplicación. Por ejemplo controles de rango de campos, campos obligatorios, etc. Mientras que el resto de la aplicación, junto con la base de datos, están en el servidor

*Ilustración 3- Logica Distribuida*

4. Gestión Remota de Datos.- El cliente realiza la interacción con el usuario y ejecuta la aplicación y el servidor es quien maneja los datos.

*Ilustración 4-Gestión remota de datos*

5. Base de Datos Distribuidas.- El cliente realiza la interacción con el usuario, ejecuta la aplicación, debe conocer la topología de la red, así como la disposición y ubicación de los datos. Se delega parte de la gestión de la base de datos al cliente.

*Ilustración 5-Base de datos distribuidas*

6. Cliente servidor a tres niveles.- El cliente se encarga de la interacción con el usuario, el servidor de la lógica de aplicación y la base de datos puede estar en otro servidor.





*Ilustración 6-cliente servidor a tres niveles*

7. Niveles de una aplicación Web. El nivel de interfaz de usuario está compuesto por las páginas HTML que el usuario solicita a un servidor Web y que visualiza en un cliente Web (normalmente, un navegador). El nivel de lógica de negocio está compuesto por los módulos que implementan la lógica de la aplicación y que se ejecutan en un servidor de aplicaciones. El nivel de datos está compuesto por los datos, normalmente gestionados por un sistema de gestión de bases de datos (servidor de datos), que maneja la aplicación web.

## Base de datos distribuida

Es una colección de datos (base de datos) construida sobre una red y que pertenecen, lógicamente, a un solo sistema distribuido, la cual cumple las siguientes condiciones:

La información de la base de datos esta almacenada físicamente en diferentes sitios de la red.

- En cada sitio de la red, la parte de la información, se constituye como una base de datos en sí misma.
- Las bases de datos locales tienen sus propios usuarios locales, sus propios DBMS y programas para la administración de transacciones, y su propio administrador local de comunicación de datos.
- Estas base de datos locales deben de tener una extensión, que gestione las funciones de sociedad necesarias; la combinación de estos componentes con los sistemas de administración de base de datos locales, es lo que se conoce como Sistema Administrador de Base de Datos Distribuidas.
- Este gestor global permite que usuarios puedan acceder a los datos desde cualquier punto de la red, como si lo hicieran con los datos de su base de datos local, es decir, para el usuario, no debe existir diferencia en trabajar con datos locales o datos de otros sitios de la red.

En consecuencia, la base de datos distribuida, es como una unidad virtual, cuyas partes se almacenan físicamente en varias bases de datos "reales" distintas, ubicadas en diferentes sitios.

### Desarrollo web

Caso particular de los sistemas Cliente-Servidor con representación remota. En donde se dispone de un protocolo estándar: HTTP y un Middleware denominado WebServer. En la actualidad la aplicación de sistemas informáticos basados en Internet, es una herramienta fundamental para las organizaciones que desean tener cierta presencia competitiva.

### Ventajas de los sistemas distribuidos

Con respecto a Sistemas Centralizados:

- Una de las ventajas de los sistemas distribuidos es la economía, pues es mucho más barato, añadir servidores y clientes cuando se requiere aumentar la potencia de procesamiento.
- El trabajo en conjunto. Por ejemplo: en una fábrica de ensamblado, los robots tienen sus CPUs diferentes y realizan acciones en conjunto, dirigidos por un sistema distribuido.
- Tienen una mayor confiabilidad. Al estar distribuida la carga de trabajo en muchas máquinas la falla de una de ellas no afecta a las demás, el sistema sobrevive como un todo.
- Capacidad de crecimiento incremental. Se puede añadir procesadores al sistema incrementando su potencia en forma gradual según sus necesidades.

Con respecto a PCs Independientes:

- Se pueden compartir recursos, como programas y periféricos, muy costosos. Ejemplo: Impresora Láser, dispositivos de almacenamiento masivo, etc.
- Al compartir recursos, satisfacen las necesidades de muchos usuarios a la vez. Ejemplo: Sistemas de reservas de aerolíneas.
- Se logra una mejor comunicación entre las personas. Ejemplo: el correo electrónico.

### Desventajas de los sistemas distribuidos

El principal problema es el software, el diseño, implantación y uso del software distribuido, pues presenta numerosos inconvenientes. Los principales interrogantes son los siguientes:

- ¿Qué tipo de S. O., lenguaje de programación y aplicaciones son adecuados para estos sistemas?.
- ¿Cuánto deben saber los usuarios de la distribución?.

- ¿Qué tanto debe hacer el sistema y qué tanto deben hacer los usuarios?.

Un aspecto primordial en este tipo de sistemas tiene que ver con las redes de comunicación. Por ejemplo: -Pérdida de mensajes, saturación en el tráfico, etc. Otro problema que puede surgir al compartir datos es la seguridad de los mismos. En general se considera que las ventajas superan a las desventajas, si estas últimas se administran seriamente.

Tecnologías de la lógica de la aplicación en el servidor web:

- CGI: Common Gateway Interface..- Son programas que se ejecutan en el servidor, pueden servir como pasarela con una aplicación o base de datos o para generar documentos html de forma automática. Cada petición http ejecuta un proceso, el cual analiza la solicitud y genera un resultado. Son independientes del SO, y presentan la ventaja de que, dado un programa escrito en un lenguaje cualquiera, es fácil adaptarlo a un CGI. Entre los lenguajes que se usan para CGIs, el más popular es el Perl.
- Servlets: Pequeños programas en Java que se ejecutan de forma persistente en el servidor, y que, por lo tanto, tienen una activación muy rápida, y una forma más simple de hacerlo. Estos programas procesan una petición y generan la página de respuesta.
- ASP (Active Server Pages): Una página ASP es un fichero de sólo texto que contiene las secuencias de comandos, junto con el HTML necesario, y que se guarda con la extensión ".asp". Al ser llamado por el navegador, el motor ASP del IIS (Internet Information Server) se encarga automáticamente de ejecutarlo como se suele hacer con un programa cualquiera, pero cuya salida siempre será a través del navegador que le invoca. Es un entorno propietario de Microsoft y el lenguaje de secuencia de comandos predeterminado del IIS es el VBScript, aunque puede cambiarse.
- JSP (Java Server Pages), que consisten en pequeños trozos de código en Java que se insertan dentro de páginas web, de forma análoga a los ASPs. Ambas opciones, hoy en día, son muy populares en sitios de comercio electrónico. Frente a los ASPs, la ventaja que presentan es que son independientes del sistema operativo y del procesador de la máquina.
- PHP es un lenguaje cuyos programas se insertan también dentro de las páginas web, al igual que los ASPs y JSPs; es mucho más simple de usar, y el acceso a bases de datos desde él es muy simple. Es tremendamente popular en sitios de comercio electrónico con poco tráfico, por su facilidad de desarrollo y rapidez de implantación.

Aplicaciones de los sistemas distribuidos

- Sistemas Comerciales.- Inicialmente fueron contruidos con hardware dedicado y entornos centralizados, son, por sus características de distribución geográfica y necesidad de acceso a sistemas distintos, ideales para implementarse en sistemas distribuidos. Requieren ciertas características de fiabilidad, seguridad y protección.
- Algunos ejemplos son:
  - Sistemas de reservas de líneas aéreas.
  - Aplicaciones bancarias.
  - Cajas y gestión de grandes almacenes.
- Redes WAN.- Debido al gran crecimiento de este tipo de redes (Internet), ha tomado gran importancia en el intercambio de información a través de la red. Y para esto tenemos los siguientes ejemplos:
  - Los servicios comunes que brinda Internet: Correo electrónico, servicio de noticias, transferencia de archivos, la World Wide Web, etc. o Aplicaciones Multimedia.- Son las últimas incorporaciones a los sistemas distribuidos. Estas aplicaciones imponen ciertas necesidades de hardware para poder tener una velocidad y regularidad de transferencia de una gran cantidad de datos. Los ejemplos de estos sistemas son:
    - Videoconferencia.
    - Televigilancia.
    - Juegos multiusuarios.
    - Enseñanza asistida por computadora.
- Áreas de la informática aplicada a los Sistemas Distribuidos.- En este punto se tienen en cuenta toda la variedad de aplicaciones de los sistemas distribuidos, pues su diseño involucra a muchas áreas, por ejemplo:
  - Comunicaciones.
  - Sistemas operativos distribuidos.
  - Base de datos distribuidas.
  - Servidores distribuidos de archivos
  - Lenguajes de programación distribuidos.
  - Sistemas de tolerancia de fallos.

## Bibliografía

Distributed Systems: Concepts and Design G. Coulouris, J. Dollimore, T. Kindberg, Editorial: Addison Wesley, 2005, 4th edition. ISBN: 0321263545

Sistemas Distribuidos (español –versión anterior -) George Coulouris; Jean Dollimore; Sebastián Dormido; Tim Kindberg Editorial: Addison Wesley | 3era Edición Idioma: Español ISBN: 8478290494.

Distributed Systems: Principles and Paradigms\*\* Andrew S. Tanenbaum, Maarten van Oteer Editorial: Prentice Hall; United States 2nd edition (Oct 2, 2006) ISBN: 0132392275

Andrew S. Tanenbaum and Robbert Van Renesse, Distributed Operating Systems. ACM Computing Surveys (CSUR), Volume 17, Issue 4. Pags. 419-470. ISSN:0360-0300. The MIT Press scientific computation series. 1985.

Eliezer Levy and Abraham Silberschatz, Distributed file systems: concepts and examples. ACM Computing Surveys (CSUR), Volume 22, Issue 4. Pags. 321-374. ISSN:0360-0300. 1990.