

TEMA 7

INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS

7.1 INTRODUCCION A LOS SISTEMAS DISTRIBUIDOS: A partir de la década de los '80 aparecen dos avances tecnológicos fundamentales:

- ♣ Desarrollo de microprocesadores poderosos y económicos
- ♣ Desarrollo de redes de AREA LOCAL (LAN) de alta velocidad.

Esto es lo que permite la aparición de los SISTEMAS DISTRIBUIDOS que necesitan un software distinto al de los sistemas centralizados.

En los SISTEMAS DISTRIBUIDOS los usuarios pueden acceder a gran velocidad a los recursos computacionales de hardware y software distribuidos entre varios sistemas conectados.

7.2 VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LOS CENTRALIZADOS: los sistemas tienden a la descentralización por la economía. Aquí no se aplica la economía de que si se paga el doble, se obtiene el cuádruplo de desempeño, con la tecnología del microprocesador la solución en costo es: limitarse a un gran número de CPU baratos reunidos en un mismo sistema.

La potencia de un sistema distribuido en “precio/desempeño” es mejor que la de un único sistema centralizado.

- *sistemas distribuidos*: diseñados para que muchos usuarios trabajen en forma conjunta.
- *sistemas paralelos*: diseñados para lograr la máxima rapidez en un único problema.

En los *sistemas distribuidos* existen varias cpu conectadas entre si y trabajan de manera conjunta. (Ciertas aplicaciones son distribuidas en forma inherente).

Otra ventaja es mayor confiabilidad al distribuir la carga de trabajo en muchas máquinas, la falla de uno no afecta a los demás. El sistema sobrevive como un todo.

Otra ventaja es el crecimiento incremental o por incrementos, podría añadirse procesadores al sistema, incrementando el poder de cómputo, pero no en breves lapsos de tiempo.

7.3 VENTAJAS DE LOS SISTEMAS DISTRIBUIDOS CON RESPECTO A LAS PC INDEPENDIENTES: los sistemas distribuidos satisfacen la necesidad de los usuarios de compartir ciertos datos (Ej. reserva de línea aérea). También se puede compartir otros recursos como programas y periféricos costosos (Ej. impresora láser).

Otra ventaja es lograr una mejor comunicación entre las personas (Ej. correo electrónico).

También hay mayor flexibilidad al distribuir la carga de trabajo entre las máquinas disponibles en la forma más eficaz para disminuir los costos. Los equipos distribuidos pueden no ser siempre PC.

7.4 DESVENTAJAS DE LOS SISTEMAS DISTRIBUIDOS: El principal problema es el software (diseño, implantación y uso). El S.O., lenguaje de programación y aplicaciones para estos sistemas es diferente según el criterio de cada uno.

Otro problema tiene que ver con las redes de comunicaciones (pérdida de mensajes, saturación). La ventaja de compartir datos tiene el problema de la seguridad.

7.5 CONCEPTOS DE HARDWARE: Los sistemas distribuidos constan de varias cpu, organizadas de diversas formas respecto de:

- La forma de interconectarlas entre si.
- Los esquemas de comunicación utilizados.

Esquemas de clasificación de los sistemas de cómputos con varias CPU:

Taxonomía de flynn: según el n° de flujo de instrucciones y el n° de flujos de datos (son características especiales).

- **sisd** (*un flujo de instrucciones y un flujo de datos*). Posee un único procesador.

- **simd** (*un flujo de instrucciones y varios flujos de datos*): ordenar procesadores con una unidad de instrucción.
- **misd** (*un flujo de varias instrucciones y un solo flujo de datos*) Varias instrucciones, un flujo de datos. No se presenta en la actualidad.
- **mimd** (*múltiples instrucciones y múltiples datos*) Un grupo de PCs, cada una con su propio contador de programa, programas y datos. Todos los sistemas distribuidos son de este tipo.

La categoría MIND se divide en:

- **multiprocesadores**: poseen memoria compartida. Los distintos procesadores comparten el mismo espacio de direcciones virtuales.
- **multicomputadoras**: no poseen memoria compartida. (Ej. grupo de PC conectadas mediante red).
Cada una de estas categorías se puede clasificar según la arquitectura de la red de interconexión en:
 - **Esquema de bus**: existe una sola red, bus, cable u otro medio que conecta todas las máquinas:
 - **Esquema con conmutador**: no existe una sola columna vertebral de conexión. Hay múltiples conexiones, los mensajes se mueven a través de los medios de conexión, se decide la conmutación en cada etapa para dirigir el mensaje a lo largo de los cables de salida (Ej. sistema mundial telefonía pública).

Otra clasificación considera el acoplamiento entre los equipos:

- **Sistemas fuertemente acoplados**: el retraso al enviar un mensaje de una computadora a otra es corto y la tasa de transmisión es alta. Se los utiliza como sistemas paralelos. (Ej. multiprocesadores).
- **Sistemas débilmente acoplados**: el retraso de los mensajes entre las máquinas es grande y la tasa de transmisión es baja. Se los utiliza como sistemas distribuidos (Ej. multicomputadoras).

7.6 MULTIPROCESADORES CON BASE EN BUSES: constan de cierto n° de cpu conectadas a un bus común, junto con un módulo de memoria. Todos los elementos precedentes operan en paralelo. Solo existe una memoria, la cual presenta la propiedad de la COHERENCIA: las modificaciones hechas por una CPU se reflejan de inmediata en las lecturas de la misma o de otra CPU. El problema de este esquema es que el bus tiende a sobrecargarse y el rendimiento a disminuir.

La solución es añadir una **memoria cache** de alta velocidad entre la cpu y el bus. El cache guarda las palabras de acceso reciente y todas las solicitudes pasan a través de la caché, si la solicitud está en la caché, esta responde a la CPU sin solicitarle nada al bus.

Un importante problema debido al uso de cache y otra CPU leer desde su caché la misma palabra y obtener distintos resultados.

Una solución consiste en diseñar la cache para que *cuando una palabra sea escrita al cache, también sea escrita a la memoria* **cache de escritura**.

7.7 MULTIPROCESADORES CON CONMUTADOR: El esquema de multiprocesador con base en buses es bueno hasta con 64 procesadores, pero al superarlo se utiliza otro método de conexión entre CPU y memoria.

1)- Se divide la memoria en módulos y son conectados a las cpu con un **conmutador de cruceta (cross-bar switch)**.

La ventaja es que muchas CPU pueden tener acceso a la memoria al mismo tiempo, aunque no a la misma memoria simultáneamente. El problema es el alto número de conmutadores ($n \text{ CPU} \times m \text{ memorias}$), ya que en cada intersección está un conmutador del punto de cruce electrónico que el hardware puede abrir o cerrar. Este conmutador se cierra momentáneamente cuando una CPU desea a una memoria en particular.

2)- un esquema que utiliza menos conmutador es la RED OMEGA, ya que posee conmutadores 2X2 (i entradas y i salidas), eligiendo los estados adecuados. Cada CPU podrá tener acceso a cada memoria.

El problema de la red omega es el retraso.

3)- otro esquema es según sistemas jerárquicos, cada CPU tiene asignada cierta memoria local que su acceso será muy rápido a su propia memoria y mas lenta a la memoria de los demás CPU: es el esquema o máquina NUMA (acceso no uniforme a la memoria).

7.8 MULTICOMPUTADORAS CON BASE EN BUSES: Es un esquema sin memoria compartida, cada cpu tiene una conexión directa con su propia memoria local.

El problema es la forma en que las cpu se comuniquen entre si. El tráfico es solo entre una cpu y otra; su volumen de tráfico es menor. Son una colección de estaciones de trabajo en una lan (red de area local).

7.8 MULTICOMPUTADORAS CON CONMUTADOR: Cada cpu tiene acceso directo y exclusivo a su propia memoria particular. Las topologías mas comunes son:

1)- La retícula: se basan en las tarjetas de circuitos impresos.

2)- El hipercubo: es un cubo n – dimensional, donde cada vértice es una CPU y cada arista es una conexión. Ej. un cubo de dimensión 4 se lo puede considerar como dos cubos ordinarios, cada uno de ellos con P vértices y 12 aristas.

7.10 CONCEPTOS DE SOFTWARE: La imagen de un sistema es determinada por el software del S.O. y no por el hardware.

Los S.O. se clasifican en dos tipos:

- El software débilmente acoplado de un sistema distribuido. Permite que las máquinas y usuarios sean independientes entre si y facilita que interactúen cuando sea necesario, los equipos individuales se distinguen bien.

- El software fuertemente acoplado: las máquinas necesitan interactuar siempre, y los equipos individuales no se distinguen bien.

Combinando los distintos tipos de hardware distribuidos con software distribuidos se logran distintas soluciones. Pero no todas son importantes desde el punto de vista del usuario.

7.11 SISTEMAS OPERATIVOS DE REDES: Una forma es el **software débilmente acoplado en hardware débilmente acoplado**; es muy utilizada (Ej. una red de estaciones de trabajo conectados mediante una LAN).

Cada usuario tiene una estación de trabajo para su uso exclusivo, con su propio s. o. y los requerimientos se resuelven localmente. Un usuario puede conectarse de manera remota con otra estación de trabajo.

Cuando una o mas máquinas soportan al sistema de archivo global compartidos, que es accesible desde todas las máquinas, son denominadas SERVIDORES DE ARCHIVOS que aceptan solicitudes de los programas de usuarios.

Los programas se ejecutan en las máquinas clientes, las solicitudes se examinan, se ejecutan y se envía la respuesta.

No es necesario que todas las máquinas ejecuten el mismo S.O.

El s. o. de este tipo debe controlar las estaciones de trabajo, los servidores de archivo y encargarse de la comunicación entre los servidores. Si los clientes y servidores ejecutan diversos S.O., deben coincidir en el formato y significado de los mensajes que intercambian, para comunicarse deben tener el mismo protocolo.

Este esquema es un S.O. de red donde cada máquina se rige por si mismas y hay pocos requisitos a lo largo del sistema.

NES (network file System): soportan sistemas heterogéneos y los equipos pueden ser también de hardware heterogéneos. Los aspectos más importantes son:

La arquitectura de NFS. Permite que una colección arbitraria de clientes y servidores compartan un sistema de archivo común. A veces todos los clientes y servidores están en una LAN pero esto no es necesario.

Protocolo de NFS: el objetivo del NFS es lograr soportar un sistema heterogéneo en donde los clientes y servidores ejecuten distintos S.O. en hardware diversos, y que la interfaz entre clientes y servidores estén bien definidas. NFS logra esta definiendo dos protocolos: “CLIENTE – SERVIDOR”

A)- Un **protocolo** es un conjunto de solicitudes que de clientes a servidores y las respuestas de servidores a clientes. Dice ¿Qué y cómo se solicita?

Un protocolo de NFS maneja el **montaje** de un directorio en su sistema, o la alternativa del AUTOMONTAJE, según pedido.

Si el cliente utiliza varios servidores en particular obtiene:

- ° Tolerancia a fallos.
- ° Mejora del rendimiento.

B)- El otro protocolo es para el acceso a directorios y archivos.

° Implantación de NFS: la implantación del código del cliente y el servidor es independiente de los protocolos NFS:

° NFS: solo trata del sistema de archivos, no hace referencia a otros aspectos, como la ejecución de un proceso.

7.12 SISTEMAS REALMENTE DISTRIBUIDOS: NFS es un ejemplo de *software débilmente acoplado en hardware débilmente acoplado*. Cada computadora ejecuta su propio s. o.; solo se dispone de un sistema compartido de archivos; el tráfico cliente - servidor debe obedecer los protocolos NFS.

Las *multicomputadoras* son un ejemplo de *software fuertemente acoplado en hardware débilmente acoplado*: crea la ilusión de que toda la red de computadoras es un solo sistema de tiempo compartido, en vez de una colección de máquinas diversas.

Un sistema distribuido es una colección de máquinas sin memoria compartida, pero que aparece ante sus usuarios como una sola computadora.

Características:

- Debe existir un mecanismo de comunicación global entre los procesos, cualquier proceso intercambie información con otro.
- Debe existir un esquema global de protección.
- La administración de procesos debe ser igual en todas partes.
- Se debe tener una misma interfaz de llamadas al sistema en todas partes:
- Es necesario un sistema global de archivos.

7.13 SISTEMAS DE MULTIPROCESADOR CON TIEMPO COMPARTIDO: Corresponde a software fuertemente acoplado en hardware fuertemente acoplado.

Los multiprocesadores operan con un sistema de tiempo compartido, pero son varias CPU en vez de una sola.

- ° MIPS: mide la velocidad del procesador. 1MIPS= 1.000.000 de instrucciones por segundo.
- ° CARACTERÍSTICAS: existe una sola cola de ejecución contenido en la memoria compartida.

Los programas de los procesos están en la memoria compartida y también el S.O. El planificador de procesos del S.O. se ejecutan como una región crítica, con ello se evita que dos CPU elijan el mismo proceso para su ejecución inmediata. Ninguna CPU tiene memoria local, todos los programas se almacenan en la memoria global compartida.

7.14 ASPECTOS DEL DISEÑO: Los aspectos claves en el diseño de s. o. distribuidos son:

- ° Transparencia: imagen de un único sistema.
- ° Flexibilidad: en cuanto a la estructura del S.O.
- ° Confiabilidad: si una máquina falta, otra hará su trabajo.
- ° Desempeño: muchas actividades en ejecución paralela.
- ° Escalabilidad: evitar los cuellos de botella con muchos usuarios.

7.15 TRANSPARENCIA: Se debe lograr la imagen de un único sistema. Los usuarios deben percibir que la colección de máquinas conectadas son un sistema de tiempo compartido de un solo procesador, entonces el sistema es **transparente**.

La transparencia (*desde el punto de vista de los usuarios*), se logra cuando se satisfacen los pedidos de los usuarios con ejecuciones en paralelo en distintas máquinas; se utilizan una variedad de servidores de archivos.

La transparencia *desde el punto de vista de los programas* significa diseñar la interfaz de llamadas al sistema de modo que no sea visible la existencia de varios procesadores.

Tipos de transparencia

- **de localización:** los usuarios no pueden indicar la localización de los recursos.
- **de migración:** los recursos se pueden mover sin cambiar sus nombres.
- **de replica:** usuarios no pueden indicar el n° de copias existentes.
- **de concurrencia:** varios usuarios pueden compartir recursos de manera automática.
- **de paralelismo:** las actividades pueden ocurrir en paralelo sin el conocimiento de los usuarios.

7.16 FLEXIBILIDAD: con respecto a la estructura de los sistemas distribuidos, existen:

- **núcleo monolítico:** proporciona el sistema de archivos, toda la administración de procesos, gran parte del manejo de las llamadas al sistema. cada máquina ejecuta un núcleo que proporciona la mayoría de los servicios.
- **micro núcleo (microkernel):** el núcleo proporciona lo menos posible, es más flexible. Proporciona solo cuatro servicios mínimos: mecanismos de comunicación entre procesos, cierta administración de la memoria, una cantidad limitada de planificación y administración de procesos de bajo nivel y E/S de bajo nivel.

7.17 CONFIABILIDAD: tolerancia a fallos. El objetivo de un sistema distribuido es que si una máquina falla, otra debe encargarse del trabajo. La confiabilidad teórica del sistema es el “OR” voléanos de los componentes, pero a veces se requiere que ciertos servidores estén en servicio simultáneamente para que todo funcione.

Otro aspecto es la DISPONIBILIDAD que es la fracción de tiempo en que se usa el sistema.

Otro es la SEGURIDAD, es decir, que los archivos y otros recursos deben protegerse para evitar el mal uso.

Otro es la TOLERANCIA A FALLOS, la falla de oculta y se hace una recuperación transparente para el usuario, aunque haya degradación de performance.

7.18 DESEMPEÑO: tener muchas actividades en ejecución paralela. Tener en cuenta el desempeño de respuesta, rendimiento (número de trabajos por hora, uso del sistema, velocidad de proceso).

7.19 ESCALABILIDAD: se debe evitar los cuellos de botellas con miles de usuarios. Se utilizan algoritmos y componentes descentralizados.