

INDICE

• INTRODUCCIÓN A LOS SISTEMAS DISTRIBUIDOS

1.1 ¿Qué es un sistema distribuido?.....1

1.2 Objetivos

- Ventajas de los sistemas distribuidos con respecto de los centralizados.....1
- Ventajas de los sistemas distribuidos con respecto de las PC independientes.2

1.2.3 Desventajas de los sistemas distribuidos..2

1.3 Conceptos de hardware.2

1.3.1 Multiprocesadores con base en buses4

1.3.2 Multiprocesadores con conmutador4

1.3.3 Multicomputadoras con base en buses..4

1.3.4 Multicomputadoras con conmutador.5

1.4 Conceptos de software..5

1.4.1 Sistemas operativos de redes.5

1.4.2 Sistemas realmente distribuidos..6

1.4.3 Sistemas de multiprocesador con tiempo compartido.6

1.5 Aspectos del diseño.6

1.5.1 Transparencia6

1.5.2 Flexibilidad7

1.5.3 Confiabilidad..7

1.5.4 Desempeño.7

1.5.5 Escalabilidad..7

Bibliografía.8

1. INTRODUCCIÓN A LOS SISTEMAS DISTRIBUIDOS

• ¿Qué es un sistema distribuido?

Un sistema distribuido es una colección de computadoras independientes; es decir autónomas, que aparecen

ante los usuarios del sistema como una única computadora.

Algunos ejemplos de sistemas distribuidos son:

- Una red de estaciones de trabajo en un departamento de una universidad o compañía, donde además de cada estación personal, podría existir una pila de procesadores en el cuarto de máquinas, que no estén asignados a usuarios específicos sino que se utilicen de manera dinámica cuando sea necesario.
- Una fábrica de robots, donde los robots actúan como dispositivos periféricos unidos a la misma computadora central.
- Un banco con muchas sucursales por el mundo, cada oficina tiene una computadora maestra para guardar las cuentas locales y el manejo de las transacciones locales, la cuál se puede comunicar con cualquier computadora de la red. Las transacciones hechas se realizan sin importar dónde se encuentre la cuenta o el cliente.

- **Objetivos**

- **Ventajas de los sistemas distribuidos con respecto de los centralizados.**

- **Economía:** es la razón número uno de la tendencia hacia los sistemas distribuidos ya que estos sistemas tienen en potencia una proporción precio/desempeño mucho mejor que la de un sistema centralizado.
- **Velocidad:** un sistema distribuido puede tener mayor poder de cómputo que una mainframe.
- **Distribución inherente:** otra razón para la construcción de un sistema distribuido es que ciertas aplicaciones son distribuidas en forma inherente; es decir, algunas aplicaciones utilizan máquinas que están separadas a cierta distancia.
- **Confiabilidad:** un sistema distribuido ofrece mayor confiabilidad: al distribuir la carga de trabajo en muchas máquinas, la falla de un circuito descompondrá a lo más una máquina y el resto seguirá intacto.
- **Crecimiento por incrementos:** si se necesita añadir poder de cómputo a la compañía, con un sistema distribuido, podrían añadirse sólo más procesadores al sistema, lo que permite un desarrollo gradual conforme surjan las necesidades.

- **Ventajas de los sistemas distribuidos con respecto de las PC independientes**

- **Datos compartidos:** Un sistema distribuido permite que varios usuarios tengan acceso a una base de datos común.
- **Dispositivos compartidos:** De igual manera, se pueden compartir periféricos entre diversos usuarios como puede ser una impresora.
- **Comunicación:** un sistema distribuido facilita la comunicación entre computadoras aisladas con el e-mail, por ejemplo.
- **Flexibilidad:** Un sistema distribuido difunde la carga de trabajo entre las máquinas disponibles en la forma más eficaz en cuanto a los costos.

- **Desventajas de los sistemas distribuidos**

- **Software:** No hay mucha experiencia en el diseño, implantación y uso del software distribuido, además existe poco software para los sistemas distribuidos en la actualidad.
- **Redes:** Una vez que el sistema llega a depender de la red, la pérdida o saturación de ésta puede negar algunas de las ventajas que el sistema distribuido debía conseguir.
- **Seguridad:** si las personas pueden tener acceso a los datos en todo el sistema, entonces también

pueden tener acceso a datos con los que no tienen nada que ver.

- **Conceptos del hardware**

Con el paso de los años, se han propuesto diversos esquemas de clasificación para los sistemas de cómputo con varios CPU, pero ninguno de ellos ha tenido un éxito completo ni se ha adoptado de manera amplia. A continuación se muestra la taxonomía presentada por Flynn (1972) que considera dos características esenciales: el número de flujo de instrucciones y número de flujos de datos.

SISD: Una computadora con un flujo de instrucciones y uno de datos. Todas las computadoras tradicionales de un procesador caen dentro de esta categoría.

SIMD: Un flujo de Instrucciones y varios flujos de datos. Este tipo se refiere a ordenar procesadores con unidad de instrucción que busca una instrucción y después instruye a varias unidades de datos para que la lleven a cabo en paralelo, cada una con sus propios datos.

MISD: Un flujo de varias instrucciones y un flujo de datos.

MIMD: Un grupo de computadoras independientes, cada una con su propio contador del programa y datos. Todos los sistemas distribuidos son MIMD.

Las computadoras MIMD se clasifican en dos grupos: aquellas que tienen memoria compartida, que por lo general se llaman **multiprocesadores** y aquellas que no, que a veces reciben el nombre de **multicomputadoras**. La diferencia esencial es ésta: en un multiprocesador, existe un espacio de direcciones virtuales, compartido por todos los CPU. En contraste, en una multicomputadora, cada máquina tiene su propia memoria.

Cada una de estas categorías se puede subdividir, con base en la arquitectura de la red de interconexión: con **bus** y con **conmutador**. En la primera queremos indicar que existe una red, plano de base, bus, cable u otro medio que conecta todas las máquinas. Los sistemas con conmutador no tienen sólo una columna vertebral como en la televisión por cable, sino que tienen cables individuales de una máquina a otra y utilizan varios patrones diferentes de cableado.

Otra dimensión de la taxonomía es que, en ciertos sistemas, las máquinas están **fuertemente acopladas** y en otras están **débilmente acopladas**. En un sistema fuertemente acoplado, el retraso que se experimenta al enviar un mensaje de una computadora a otra es corto y la tasa de transmisión de los datos, es decir, el número de bits por segundo que se puede transferir, es alta. En un sistema débilmente acoplado ocurre lo contrario: el retraso de los mensajes entre las máquinas es grande y la tasa de transmisión de los datos es baja. Los sistemas fuertemente acoplados tienden a utilizarse como sistemas distribuidos aunque esto no siempre es cierto.

1.3.1. Multiprocesadores con base en buses

Los multiprocesadores con base en buses constan de cierta cantidad de CPU, conectados a un bus común, junto con un módulo de memoria. Una configuración sencilla consta de un plano de base de alta velocidad o tarjeta madre, en el cual se pueden insertar las tarjetas de memoria y el CPU. Un bus típico tiene 32 o 64 líneas de direcciones, 32 o 64 líneas de datos y 32 o más líneas de control, todo lo cual opera en paralelo. Para leer una palabra de memoria, un CPU coloca la dirección de la palabra deseada en las líneas de direcciones del bus y coloca una señal en las líneas de control adecuadas para indicar que desea leer. La memoria responde y coloca el valor de la palabra en las líneas de datos para permitir la lectura de ésta por parte del CPU solicitante. La escritura funciona de manera similar.

1.3.2 Multiprocesadores con conmutador

Para construir un multiprocesador con más de 64 procesadores, es necesario un método distinto para conectar cada CPU con la memoria. Una posibilidad es dividir la memoria en módulos y conectarlos a las CPU con un **conmutador de cruceta**, cada CPU y cada memoria tiene una conexión que sale de él. En cada intersección está un delgado **conmutador de punto de cruce** electrónico que el hardware puede abrir y cerrar. Cuando un CPU desea tener acceso a una memoria particular, el conmutador del punto de cruce que los conecta se cierra de manera momentánea, para permitir dicho acceso. La virtud del conmutador de cruceta es que muchos CPU pueden tener acceso a la memoria al mismo tiempo, aunque si dos CPU intentan tener acceso a la misma memoria en forma simultánea, uno de ellos deberá esperar.

1.3.3 Multicomputadoras con base en buses

Por otro lado, la construcción de una multicomputadora es fácil. Cada CPU tiene conexión directa con su propia memoria local. El único problema restante es la forma en que los CPU se comunicarán entre sí. Es claro que aquí también se necesita cierto esquema de interconexión, pero como sólo es para la comunicación entre un CPU y otro, el volumen del tráfico será de varios órdenes menor en relación con el uso de una red de interconexión para el tráfico CPU–memoria.

1.3.4 Multicomputadoras con conmutador

Se han propuesto y construido varias redes de interconexión, pero todas tienen la propiedad de que cada CPU tiene acceso directo y exclusivo a su propia memoria particular. Hay dos topologías populares, una **retícula** y un **hipercubo**. Las retículas se basan en las tarjetas de circuitos impresos. Se adecuan mejor a los problemas con naturaleza bidimensional inherente, como la teoría de gráficas o la visión. Un hipercubo es un cubo n -dimensional. Se puede pensar como dos cubos ordinarios, cada uno de los cuales cuenta con 8 vértices y 12 aristas. Cada vértice es un CPU. Cada arista es una conexión entre dos CPU. Se conectan los vértices correspondientes de cada uno de los cubos.

• Conceptos de software

Aunque el hardware es importante, el software lo es más. La imagen que presenta y la forma de pensar de los usuarios de un sistema, queda determinada en gran medida por el software del sistema operativo, no por el hardware.

Se puede distinguir dos tipos de sistemas operativos para los de varios CPU: los débilmente acoplados y los fuertemente acoplados.

El software débilmente acoplado permite que las máquinas y los usuarios de un sistema distribuido sean independientes entre sí en lo fundamental, pero que interactúen en cierto grado cuando sea necesario.

En el software fuertemente acoplado el programa de aplicación y el sistema operativo necesario para soportarlo, están muy acoplados.

1.4.1 Sistemas Operativos de red

Los Sistemas Operativos de red permiten a los usuarios en estaciones de trabajo independientes la comunicación por medio de un sistema compartido de archivos, pero dejan que cada usuario domine su propia estación de trabajo.

1.4.2 Sistemas realmente distribuidos

Los sistemas operativos distribuidos convierten toda la colección de hardware y software en un sistema integrado, muy parecido a un sistema tradicional de tiempo completo.

1.4.3 Sistemas de multiprocesador con tiempo compartido

Los multiprocesadores con memoria compartida también ofrecen la imagen de único sistema, pero lo hacen mediante la vía de centralizar todo, por lo que en realidad, este caso es un sistema. Los multiprocesadores con memoria compartida no son sistemas distribuidos.

- **Aspectos del diseño**

A continuación se analiza en forma breve algunos de los aspectos claves del diseño con los que deben trabajar las personas que piensan construir un sistema operativo distribuido:

- **Transparencia**

Diseñar el sistema de forma que todas las personas piensen que la colección de máquinas es tan sólo un sistema de tiempo compartido, de un procesador, a la manera antigua.

La transparencia se puede lograr en dos niveles distintos. Lo más fácil es ocultar la distribución a los usuarios, aunque también es posible hacer que el sistema sea transparente para los programas.

Los distintos tipos de transparencia en un sistema distribuido son:

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

- **Flexibilidad**

El diseño de un sistema distribuido debe hacerse con la idea de facilitar los cambios futuros. A este respecto, los micronúcleos son superiores a los núcleos monolíticos.

- **Confiabilidad**

Uno de los objetivos originales de la construcción de sistemas distribuidos fue el hacerlos más confiables que los sistemas con un procesador. La idea es que si una máquina falla, alguna otra máquina se encargue del trabajo. En otras palabras, en teoría, la confiabilidad global del sistema podría ser el OrBooleano de la confiabilidad de los componentes.

Un aspecto importante dentro de la confiabilidad es la **disponibilidad** que se refiere a la fracción de tiempo en que se puede utilizar el sistema.

Otro aspecto de la confiabilidad general es la **seguridad**. Los archivos y otros recursos deben ser protegidos contra el uso no autorizado.

También otro aspecto de la confiabilidad es la **tolerancia de fallas**. los sistemas distribuidos se deben diseñar de forma que escondan las fallas; es decir, ocultarlo a los usuarios.

- **Desempeño**

Cuando se ejecuta una aplicación en un sistema distribuido, no debe parecer peor que su ejecución en un procesador. Se pueden utilizar diversas métricas de desempeño. El tiempo de respuesta es una, pero también lo son el rendimiento, uso del sistema y cantidad consumida de la capacidad de la red. Además es frecuente que el resultado de cualquier parámetro dependa de la naturaleza de éste.

- **Escalabilidad**

Diseñar un sistema distribuido que pueda escalar hacia sistemas más grandes en un futuro.

Bibliografía

Tanenbaum, A. S. (1996). *Sistemas Operativos distribuidos*. Ed Prentice Hall. 1ª Ed.

Carina Muñiz Macias karilyn_rpg@hotmail.com ITC 9no. Informática.