

Definición básica de conceptos:

- ¿Qué es el multiproceso?

Muchos procesos a la vez, mas de una CPU compartiendo recursos.

- ¿Qué es el multiusuario?

Muchos usuarios a la vez usando el mismo sistema/cpu

- ¿Cuál es la diferencia entre un sistema SMP y un sistema AMP?

los dos comparten recursos de forma equitativa, el otro el master controla el sistema y reparte los recursos

- ¿Qué es MIMD?

Va repartiendo procesos entre los procesadores secundarios trabajando en paralelo

Componentes y tipos:

- ¿Qué es una CPU y qué papel juega en un sistema de multiprocesador?

El procesador se encargan de procesar instrucciones, pueden ser identicas o no pero, procesan en paralelo y hacen que aumente la capacidad de procesamiento

- ¿Qué es la memoria compartida en un sistema de multiprocesador?

Todas las CPU tienen acceso a una misma memoria que ha de tener sistemas de control de coherencia e integridad ya que acceden a ella de forma simultánea

- Nombra dos ejemplos de sistemas operativos que soporten SMP.

Linux y Windows server

- ¿Qué es una interconexión en un sistema multiprocesador?

Como están conectados los sistemas entre anillo, bus, estrella,

Ventajas y Desafíos:

- Menciona una ventaja de los sistemas multiprocesadores.

Rendimiento y eficiencia en lo que hacen, más procesos y mejor.

- ¿Qué desafío representa la coherencia de caché en un sistema multiprocesador?

Al trabajar varios procesadores pueden acceder a la misma posicion de memoria y si las variable s no se actualizan correctamente puede dar lugar a errores. MESI MOESI

- ¿Por qué pueden ser costosos los sistemas multiprocesador?

La complejidad adicional en el hardware y el software.

Arquitecturas y sistemas operativos:

- Explica la arquitectura de un sistema de multiprocesador.

Varias CPU que comparten memoria, Almacenamiento o periféricos y recursos. y que

- ¿Qué es un sistema de tiempo real (RTOS) y en qué situaciones se utiliza?

Diseñados para respuesta rápida y predecible

- ¿Cuáles son las características principales de un sistema operativo de procesamiento distribuido?

Escalabilidad extrema, al no necesitar que las computadoras estén físicamente unidas,
Tolerancia de fallos, al existir redundancia y distribución de tareas

Comunicación eficiente entre nodos

- Describe cómo funciona un clúster de computadoras.

Grupo de computadoras que trabajan de forma eficiente de manera cohesiva y unificada.
Muchos son uno. (supercomputadores).

Modelos y ventajas/desventajas:

- Describe las ventajas de utilizar sistemas multiprocesadores.

Vide supra

- ¿Cuáles son los desafíos que presentan los sistemas multiprocesadores en términos de coherencia de caché?

Vide supra

- ¿Qué es una multicomputadora y cómo difiere de un sistema multiprocesador?

Muchas computadoras cada una con su hardware no comparten memoria, recursos o almacenamiento. comunicación entre ellos con protocolos LAN

- ¿Qué es un modelo de usuario único y cuáles son sus ventajas y desventajas?

Modelo donde sólo hay un tipo de usuario, mas sencillo e ideal para pocos usuarios pero de escalabilidad limitada.

Componentes y Operación:

- ¿Cuál es el papel de los controladores de entrada/salida en un sistema multiprocesador?

Deben manejar múltiples solicitudes simultáneamente de diferentes CPUs así que tienen que estar diseñados de forma eficiente para evitar cuellos de botella y garantizar un acceso rápido a dispositivos de almacenamiento, redes y otros periféricos.

- ¿Qué son los nodos de procesamiento en una multicomputadora?

Las unidades independiente de computación que componen el sistema. Estos nodos trabajan en conjunto para resolver tareas más grandes, y cada uno tiene su propio conjunto de recursos, como CPU, memoria y almacenamiento

- ¿Qué funciones cumple el middleware en una multicomputadora?

Coordinación de transacciones, comunicación, seguridad, interoperatividad, gestión de sesiones y orquestación de servicios.

Análisis de componentes y organización:

- Analiza los diferentes modelos de organización de usuarios y sus respectivas ventajas y desventajas.

Cuanto mas usuarios mas complejo el mantenimiento del sistema pero mas seguridad.

- Explica cómo funcionan los controladores de entrada/salida en un sistema multiprocesador.

Manejar múltiples solicitudes simultáneamente de diferentes CPUs así que tienen que estar diseñados de forma eficiente para evitar cuellos de botella y garantizar un acceso rápido a dispositivos de almacenamiento, redes y otros periféricos.

- Discute las implicaciones de utilizar un modelo de identidad federada en términos de seguridad y gestión.

al tener una sola contraseña es mas fácil de hackear y mas difícil de tener controlado en que servicios tienes admitidos los permisos.

- ¿Cómo se implementa la coherencia de caché en sistemas multiprocesadores?

Protocolo MESI: El protocolo MESI utiliza cuatro estados y la variable va cambiando de estado :

1. Modified (M): El bloque de datos ha sido modificado (es diferente de la memoria principal) y existe únicamente en la caché actual.
2. Exclusive (E): El bloque de datos es coherente con la memoria principal y existe únicamente en la caché actual.
3. Shared (S): El bloque de datos es coherente con la memoria principal y puede estar presente en múltiples cachés.
4. Invalid (I): El bloque de datos no es válido en la caché actual.

Protocolo MOESI: igual que el MESI, pero incluye a continuación de M

- Owned (O): El bloque de datos puede estar compartido con otras cachés, pero esta caché es responsable de actualizar la memoria principal. Es coherente para todas las cachés que lo comparten.

Implementación y casos de uso:

- ¿Cómo se logra la sincronización y comunicación eficiente entre nodos en una multicomputadora?

Mediante MiddleWare y conexiones a través de LAN, Router y switches.

- Describe un caso práctico donde un sistema operativo de tiempo real (RTOS) sería esencial.

Cualquiera que necesite que los datos de entrada se procesen y evalúen en el momento como sistemas de control de vuelo, misiles y drones, dispositivos médicos...

- Analiza el impacto de la topología de interconexión en el rendimiento de un sistema multiprocesador.

Dependiendo de como se conectan son mas sensibles a caídas de la red o fallos de sistema o no.

Latencia, ancho de banda, escalabilidad y tolerancia de errores. son las tres caracterisrticas que mas se ven afectadas dependiendo de la topología usada.

- ¿Qué ventajas ofrece el uso de frameworks como Apache Hadoop y Apache Spark en entornos de multicomputadoras?

Apache Hadoop: procesamiento distribuido y almacenamiento de grandes datos

Apache Spark: procesamiento de datos en tiempo real.

Escalabilidad, redundancia y flexibilidad.

Optimización y Seguridad:

- ¿Qué técnicas se utilizan para evitar la contención de recursos en sistemas multiprocesadores?

La contención de recursos ocurre cuando múltiples procesadores intentan acceder simultáneamente a un recurso compartido, como memoria o dispositivos de entrada/salida, lo que lleva a problemas de rendimiento y eficiencia.

Bloqueos Finos(fine-grained Locking), algoritmos no bloqueantes (Non-blocking Algorithms), división de datos (Data Partitioning), algoritmos de consenso (Consensus Algorithms) y copias locales de datos.

- ¿Cómo se gestionan los permisos y roles en un modelo de Roles Basados en Acceso (RBAC)?

Se crean diferentes roles de usuario con acceso a las partes que cada rol pueda necesitar.

un usuario puede tener varios roles.

- Analiza los desafíos de seguridad en sistemas operativos distribuidos.

Que puede al afectar a una vulnerabilidad puede "infectar" al resto al haber entrado dentro de la red y ser difícil de parar o de controlar los daños por no tener que estar físicamente en el mismo sitio geográfico

- ¿Qué estrategias se pueden utilizar para balancear la carga en una multicomputadora?

estática vs dinámica

1. Round Robin (RR)

- **Descripción:** El algoritmo de round robin distribuye las solicitudes de manera secuencial y cíclica entre los servidores disponibles.
- **Ventajas:**
 - Implementación simple y fácil de entender.
 - Distribución uniforme de la carga en situaciones ideales.
- **Desventajas:**
 - No tiene en cuenta la carga actual de los nodos, lo que puede llevar a desigualdades en la carga.

2. Least Connections (LC)

- **Descripción:** Asigna la solicitud al servidor con el menor número de conexiones activas. Ideal para aplicaciones donde las conexiones persisten durante diferentes períodos de tiempo.
- **Ventajas:**
 - Eficiente en entornos donde las cargas de trabajo varían.
- **Desventajas:**
 - Puede no funcionar bien si la variabilidad en la duración de las conexiones es alta.

3. Least Response Time (LRT)

- **Descripción:** Envía las solicitudes al servidor con el menor tiempo de respuesta. Este enfoque considera tanto la cantidad de conexiones como el tiempo de respuesta.
- **Ventajas:**
 - Ideal para entornos sensibles al tiempo de respuesta.
- **Desventajas:**
 - Necesita monitorización constante del tiempo de respuesta, lo que puede añadir complejidad.

4. Weighted Round Robin (WRR)

- **Descripción:** Similar al algoritmo round robin, pero asigna un peso a cada servidor que determina la proporción de solicitudes que debe recibir.
- **Ventajas:**
 - Permite control granular sobre la distribución de carga, útil si los servidores tienen diferentes capacidades.
- **Desventajas:**
 - Necesita una configuración adecuada de los pesos iniciales.

5. Weighted Least Connections (WLC)

- **Descripción:** Combina el enfoque de menos conexiones con pesos para distribuir la carga basada en las capacidades de los servidores.
- **Ventajas:**
 - Más efectivo en entornos con heterogeneidad en los recursos de servidores.
- **Desventajas:**
 - Complejidad en la configuración y mantenimiento de los pesos y la monitorización de las conexiones.

6. IP Hashing

- **Descripción:** Utiliza una función hash en la dirección IP del cliente para determinar a qué servidor se asigna la solicitud. Asegura que solicitudes sucesivas del mismo cliente se envíen al mismo servidor.

- **Ventajas:**
 - Buena para mantener la afinidad de sesión o "sticky sessions".
- **Desventajas:**
 - Puede resultar en una distribución desigual si las solicitudes provienen de un pequeño número de direcciones IP.

7. **Random (Aleatorio)**

- **Descripción:** Asigna solicitudes a servidores de manera completamente aleatoria.
- **Ventajas:**
 - Simple de implementar.
- **Desventajas:**
 - Puede llevar a una mala distribución de carga.

Xtrap