



# Sistemas de Bases de Datos 2

**2024**

**Ing. Luis Alberto Arias Solórzano**

**Unidad 2**



# Failover

- El concepto de **tolerancia a fallos** (en inglés: *failover*) hace referencia a la capacidad de un sistema de acceder a la información, aun en caso de producirse algún fallo o anomalía en el sistema.
- Una posibilidad es que el fallo se deba a daños físicos en uno o más componentes de hardware, con la consiguiente pérdida de la información almacenada. La implementación de la tolerancia a fallos requiere que el sistema de almacenamiento guarde la misma información en más de un soporte físico (redundancia), o en un equipo o dispositivo externo a modo de respaldo. De esta forma, si se produce alguna falla que pueda ocasionar pérdida de datos, el sistema debe ser capaz de restablecer toda la información, recuperando los datos necesarios a partir de algún medio de respaldo disponible.



# Failover y failback

- Failover, es el proceso de temporalmente cambiar el ambiente de producción por el ambiente de backup/recovery seguido de un periodo de mantenimiento o despues de un desastre de producción.
- Failback, es el proceso de retornar producción a su lugar y estado original. Estas operaciones usan funciones remotas para reducir el tiempo que es requerido para sincronizar volúmenes despues de intercambiar los ambientes despues de un Failover.

# Paralelismo

- De forma general el concepto de paralelismo en las bases de datos lo podríamos definir como la partición de la base de datos (normalmente a nivel de relaciones) para poder procesar de forma paralela en distintos discos y con distintos procesadores una sola operación sobre la base de datos.
- El paralelismo se usa para mejorar la velocidad en la ejecución de consultas. Además el paralelismo se usa para proporcionar dimensionabilidad ya que la creciente carga de trabajo se trata sin incrementar el tiempo de respuesta pero incrementando el grado de paralelismo.
- Existen cuatro arquitecturas de sistemas paralelos:
  - De memoria compartida: Todos los procesadores comparten una memoria común.
  - De discos compartidos: Todos los procesadores comparten un conjunto de discos común.
  - Sin compartimiento: Los procesadores no comparten ni memoria ni disco.
  - Jerárquica: Este modelo es un híbrido de las arquitecturas anteriores.

# Paralelismo

- **Paralelismo de grano grueso**, disponiendo de unos pocos procesadores (normalmente dos o cuatro) que comparten la misma memoria principal. Las bases de datos que se ejecutan en tales máquinas habitualmente no intentan dividir una consulta simple entre los distintos procesadores, sino que ejecuta cada consulta en un único procesador posibilitando la concurrencia de varias consultas. Así, estos sistemas soportan una mayor productividad, es decir, permiten ejecutar un mayor número de transacciones por segundo, a pesar de que cada transacción individualmente no se ejecute más rápido.
- **Paralelismo de grano fino** tienen un gran número de procesadores y los sistemas de bases de datos que se ejecutan sobre ellas intentan hacer paralelas las tareas simples (consultas) que solicitan los usuarios.

# Memoria compartida

- En una arquitectura de **memoria compartida** los procesadores y los discos tienen acceso a una memoria común, normalmente a través de un bus o de una red de interconexión. El beneficio de la memoria compartida es la extremada eficiencia en cuanto a la comunicación entre procesadores; cualquier procesador puede acceder a los datos de la memoria compartida sin necesidad de la intervención del software. Un procesador puede enviar mensajes a otros procesadores utilizando escrituras en la memoria de modo que la velocidad de envío es mucho mayor (normalmente es inferior a un microsegundo) que la que se alcanza con un mecanismo de comunicación.
- El inconveniente de las máquinas con memoria compartida es que la arquitectura no puede ir más allá de 32 o 64 procesadores porque el bus o la red de interconexión se convertirían en un cuello de botella (ya que está compartido por todos los procesadores). Llega un momento en el que no sirve de nada añadir más procesadores, ya que éstos emplean la mayoría de su tiempo esperando su turno para utilizar el bus y así poder acceder a la memoria.
- Las arquitecturas de memoria compartida suelen dotar a cada procesador de una memoria caché muy grande para evitar las referencias a la memoria compartida siempre que sea posible. No obstante, en la caché no podrán estar todos los datos y no podrá evitarse el acceso a la memoria compartida. Además, las cachés necesitan mantener la coherencia; esto es, si un procesador realiza una escritura en una posición de memoria, los datos de dicha posición de memoria se deberían actualizar en o eliminar de cualquier procesador donde estuvieran los datos en caché. El mantenimiento de la coherencia de la caché aumenta la sobrecarga cuando aumenta el número de procesadores. Por estas razones las máquinas con memoria compartida no pueden extenderse llegado un punto; las máquinas actuales con memoria compartida no pueden soportar más de 64 procesadores.

# Disco Compartido

- En el modelo de **disco compartido** todos los procesadores pueden acceder directamente a todos los discos a través de una red de interconexión, pero los procesadores tienen memorias privadas. Las arquitecturas de disco compartido ofrecen dos ventajas respecto de las de memoria compartida. Primero, el bus de la memoria deja de ser un cuello de botella, ya que cada procesador dispone de memoria propia. Segundo, esta arquitectura ofrece una forma barata para proporcionar una cierta **tolerancia ante fallos**: si falla un procesador (o su memoria) los demás procesadores pueden hacerse cargo de sus tareas, ya que la base de datos reside en los discos, a los cuales tienen acceso todos los procesadores.
- Utilizando una arquitectura RAID también puede conseguirse que el subsistema de discos sea tolerante ante fallos por sí mismo. La arquitectura de disco compartido tiene aceptación en bastantes aplicaciones. El problema principal de los sistemas de discos compartidos es, de nuevo, la ampliabilidad. Aunque el bus de la memoria no es cuello de botella muy grande, la interconexión con el subsistema de discos es ahora el nuevo cuello de botella; esto es especialmente grave en situaciones en las que la base de datos realiza un gran número de accesos a los discos. Los sistemas de discos compartidos pueden soportar un mayor número de procesadores en comparación con los sistemas de memoria compartida, pero la comunicación entre los procesadores es más lenta (hasta unos pocos milisegundos si se carece de un hardware de propósito especial para comunicaciones), ya que se realiza a través de una red de interconexión. Las agrupaciones DEC con Rdb constituyen uno de los primeros usuarios de la arquitectura de bases de datos de disco compartido (Rdb ahora es propiedad de Oracle y se denomina Oracle Rdb).

# Sin compartimiento

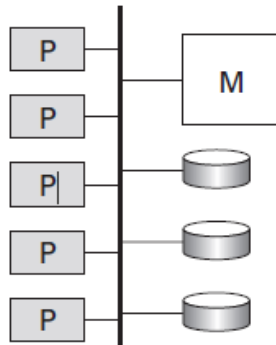
- En un sistema **sin compartimiento** cada nodo de la máquina consta de un procesador, memoria y uno o más discos. Los procesadores de un nodo pueden comunicarse con un procesador de otro nodo utilizando una red de interconexión de alta velocidad. Un nodo funciona como el servidor de los datos almacenados en los discos que posee. El modelo sin compartimiento salva el inconveniente de requerir que todas las operaciones de E/S vayan a través de una única red de interconexión, ya que las referencias a los discos locales son servidas por los discos locales de cada procesador; solamente van por la red las peticiones, los accesos a discos remotos y las relaciones de resultados. Es más, habitualmente las redes de interconexión para los sistemas sin compartimiento se diseñan para ser ampliables por lo que su capacidad de transmisión crece a medida que se añaden nuevos nodos. Como consecuencia, las arquitecturas sin compartimiento son más ampliables y pueden soportar con facilidad un gran número de procesadores. El principal inconveniente de los sistemas sin compartimiento es el coste de comunicación y de acceso a discos remotos, coste que es mayor que el que se produce



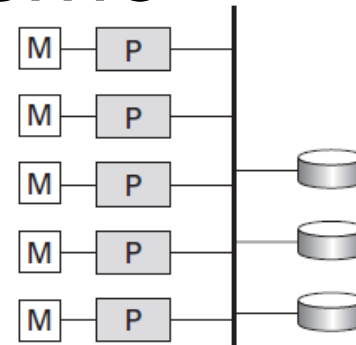
# Jerárquica

- La **arquitectura jerárquica** combina las características de las arquitecturas de memoria compartida, de disco compartido y sin compartimiento. A alto nivel el sistema está formado por nodos que están conectados mediante una red de interconexión y que no comparten ni memoria ni discos.
- Así, el nivel más alto es una arquitectura sin compartimiento. Cada nodo del sistema podría ser en realidad un sistema de memoria compartida con algunos procesadores. Alternativamente, cada nodo podría ser un sistema de disco compartido y cada uno de estos sistemas de disco compartido podría ser a su vez un sistema de memoria compartida. De esta manera, un sistema podría construirse como una jerarquía con una arquitectura de memoria compartida con pocos procesadores en la base, en lo más alto una arquitectura sin compartimiento y quizá una arquitectura de disco compartido en el medio.
- Hoy en día los sistemas paralelos comerciales de bases de datos pueden ejecutarse sobre varias de estas arquitecturas.

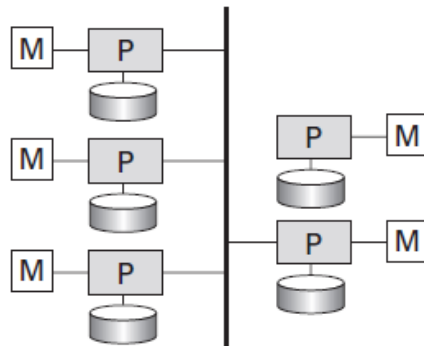
# Paralelismo



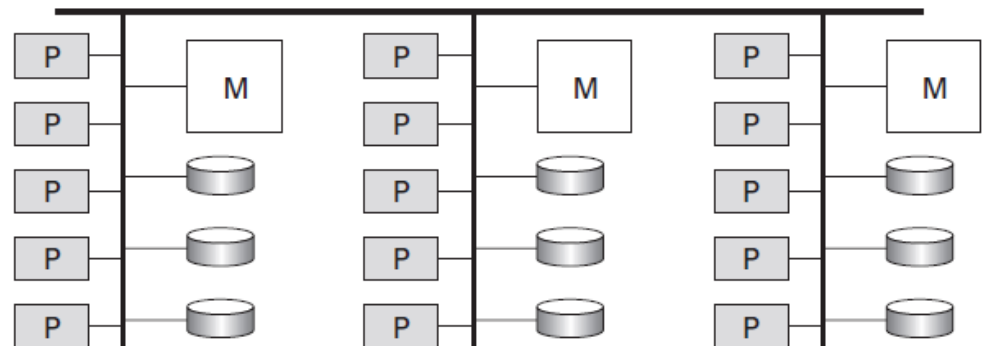
(a) Memoria compartida



(b) Disco compartido



(c) Sin compartimiento



(d) Jerárquico



# TAREA

Tarea: Leer 430-435 de fundamentos de bases de datos y 623-628.



Gracias