



# Sistemas de Bases de Datos 2

**2024**

**Ing. Luis Alberto Arias Solórzano**

**Unidad 2**



# Recuperación

- El sistema de bases de datos debe realizar con anticipación acciones que garanticen que las propiedades de atomicidad y durabilidad de las transacciones
- Una parte integral de un sistema de bases de datos es un **esquema de recuperación**, el cual es responsable de la restauración de la base de datos al estado consistente previo al fallo. El esquema de recuperación también debe proporcionar **alta disponibilidad**; esto es, debe minimizar el tiempo durante el que la base de datos no se puede usar después de un fallo.



# Recuperación

- Para determinar cómo el sistema debe recuperarse de los fallos, es necesario identificar los modos de fallo de los dispositivos de almacenamiento e identificar cómo afectan estos modos de fallo a los contenidos de la base de datos.
- Entonces se pueden proponer algoritmos para garantizar la consistencia de la base de datos y la atomicidad de las transacciones a pesar de los fallos. Estos algoritmos se conocen como algoritmos de recuperación, aunque constan de dos partes:
  - **1.** Acciones llevadas a cabo durante el procesamiento normal de transacciones para asegurar que existe información suficiente para permitir la recuperación frente a fallos.
  - **2.** Acciones llevadas a cabo después de ocurrir un fallo para restablecer el contenido de la base de datos a un estado que asegure la consistencia de la base de datos, la atomicidad de la transacción y la durabilidad.



# Clasificación de Fallos

En un sistema pueden producirse varios tipos de fallos, cada uno de los cuales requiere un tratamiento diferente. El tipo de fallo más fácil de tratar es el que no conduce a una pérdida de información en el sistema. Los fallos más difíciles de tratar son aquellos que provocan una pérdida de información.

- **Fallo en la transacción.** Hay dos tipos de errores que pueden hacer que una transacción falle:
  - **Error lógico.** La transacción no puede continuar con su ejecución normal a causa de alguna condición interna, como una entrada incorrecta, datos no encontrados, desbordamiento exceso del límite de recursos.
  - **Error del sistema.** El sistema se encuentra en un estado no deseado (por ejemplo, de interbloqueo) como consecuencia del cual una transacción no puede continuar con su ejecución normal. La transacción, sin embargo, se puede volver a ejecutar más tarde.



# Clasificación de Fallos

- **Caída del sistema.** Un mal funcionamiento del hardware o un error en el software de la base de datos o del sistema operativo causa la pérdida del contenido de la memoria volátil y aborta el procesamiento de una transacción. El contenido de la memoria no volátil permanece intacto y no se corrompe. La suposición de que los errores de hardware o software fueren una parada del sistema, pero no corrompan el contenido de la memoria no volátil, se conoce como **supuesto de fallo-parada**.
- **Fallo de disco.** Un bloque del disco pierde su contenido como resultado de bien una colisión de la cabeza lectora, bien un fallo durante una operación de transferencia de datos. Las copias de los datos que se encuentran en otros discos o en archivos de seguridad en medios de almacenamiento secundario, como cintas, se utilizan para recuperarse del fallo.



# Recuperación de Transacciones

- Una transacción comienza con la ejecución satisfactoria de una instrucción `BEGIN TRANSACTION` y termina con la ejecución satisfactoria de una instrucción `COMMIT` o `ROLLBACK`. `COMMIT` establece lo que es conocido, entre muchas otras acepciones, como **punto de confirmación**, también se conoce como **punto de sincronización**. Un punto de confirmación corresponde entonces al final de una unidad de trabajo lógica y por lo tanto, a un punto en el cual la base de datos está o debería estar en un estado consistente.
- Por el contrario, `ROLLBACK` regresa la base de datos al estado en que estaba antes de `BEGIN TRANSACTION`, lo que en efecto significa regresar al punto de confirmación anterior.

# Recuperación de Transacciones

Al establecer un punto de confirmación:

- 1. Todas las actualizaciones hechas por el programa en ejecución desde el punto de confirmación anterior, son confirmadas; es decir, se **vuelven permanentes**. Antes del punto de confirmación, todas estas actualizaciones deben ser vistas como *meramente tentativas*, en el sentido de que pueden ser deshechas posteriormente. Es un hecho que una vez que una transacción ha sido confirmada nunca podrá ser deshecha.
- 2. Todo el posicionamiento de la base de datos se pierde y todos los bloqueos de tuplas son liberados. El "posicionamiento de la base de datos" se refiere en este caso a la idea de que en cualquier momento, un programa en ejecución tendrá una direccionalidad hacia ciertas tuplas: esta direccionalidad se pierde en un punto de confirmación.



# Recuperación de Transacciones

- Por lo tanto, ahora podemos ver que las transacciones no sólo representan la unidad de trabajo sino también la unidad de **recuperación**.
- Si una transacción es confirmada satisfactoriamente, el sistema garantizará que sus actualizaciones queden grabadas permanentemente en la base de datos, aunque el sistema llegue a fallar en el siguiente momento.
  - Por ejemplo, es probable que el sistema falle después de haber recibido a COMMIT, pero antes de escribir físicamente las actualizaciones en la base de datos; éstas podrían seguir esperando en un búfer de memoria principal y por lo tanto, estar perdidas en el momento de la falla. Aunque esto suceda, el procedimiento de reinicio del sistema grabará esas actualizaciones en la base de datos. Éste es capaz de descubrir los valores que hay que escribir, examinando las entradas relevantes de la bitácora.
- Se deduce que la bitácora debe ser escrita físicamente antes de que termine el procesamiento de COMMIT; lo que se conoce como **regla de escritura anticipada de la bitácora**.
- Por lo tanto, el procedimiento de reinicio recuperará cualquier transacción que haya terminado satisfactoriamente pero que no haya podido lograr que sus actualizaciones fueran escritas físicamente antes de la caída.





# Recuperación basada en bitácora

- La estructura más ampliamente utilizada para guardar las modificaciones de una base de datos es el **registro histórico o bitácora**.
- La bitácora es una secuencia de **registros** que mantiene un registro de todas las actividades de actualización de la base de datos. Existen varios tipos de registros del registro histórico. Un **registro de actualización del registro histórico** describe una única escritura en la base de datos y tiene los siguientes campos:
  - El **identificador de la transacción** es un identificador único de la transacción que realiza la operación escribir.
  - El **identificador del elemento de datos** es un identificador único del elemento de datos que se escribe. Normalmente suele coincidir con la ubicación del elemento de datos en el disco.
  - El **valor anterior** es el valor que tenía el elemento de datos antes de la escritura.
  - El **valor nuevo** es el valor que tendrá el elemento de datos después de la escritura.





# Recuperación del Sistema

- El sistema debe estar preparado para recuperarse, no solamente de las fallas puramente locales como una condición de desborde dentro de una transacción individual, sino también de las fallas "globales", como una falla en el suministro eléctrico. Por definición, una falla local afecta solamente a la transacción en la cual está ocurriendo. Por el contrario, una falla global afecta a todas las transacciones que están en progreso en el momento de la misma y por lo tanto, tiene implicaciones significativas al nivel del sistema. En esta sección y la siguiente, consideraremos brevemente lo que implica la recuperación de una falla global.

Tales fallas caen en dos categorías amplias:

- Las **fallas del sistema** (por ejemplo, falla en el suministro eléctrico) que afectan a todas las transacciones que están actualmente en progreso, pero que no dañan físicamente a la base de datos. A la falla de sistema se le conoce a veces como una *caída blanda*.
- Las **fallas del medio** (por ejemplo, un roce de las cabezas con el disco) que causan daño a la base de datos o a alguna parte de ella, y afectan al menos las transacciones que están usando actualmente esa parte. A una falla del medio se le conoce a veces como una *caída dura*.



# Recuperación del Sistema

¿Cómo sabe el sistema cuáles transacciones debe deshacer y cuáles rehacer al momento del reinicio?

La respuesta es la siguiente. A determinados intervalos prescritos, por lo general cuando ya se ha escrito una determinada cantidad de entradas en la bitácora, el sistema automáticamente **hace un punto de verificación**.

Hacer un punto de verificación involucra (a) escribir físicamente ("escritura forzada") el contenido de los búferes de la base de datos hacia la base de datos física y (b) escribir físicamente un **registro de punto de verificación** especial en la bitácora física.

El registro de punto de verificación da una lista de todas las transacciones que estaban en progreso en el momento en que se realizó el punto de verificación.



# Recuperación del Medio

- Una falla del medio es una falla en la cual parte de la base de datos se destruye físicamente.
- En general, la recuperación de este tipo de fallas implica básicamente volver a cargar (o *restaurar*) la base de datos a partir de una copia de seguridad (o *respaldo*), y luego usar la bitácora, tanto las partes activas como las archivadas, para volver a hacer todas las transacciones que se terminaron a partir de que se realizó el respaldo. No hay necesidad de deshacer transacciones que estaban en progreso al momento de la falla, ya que por definición todas las actualizaciones de esas transacciones ya han sido "deshechas" (de hecho se perdieron).
- La necesidad de poder realizar la recuperación del medio implica el uso de una *utilería de vaciado y restauración* (o *descarga/recarga*). La parte de vaciado de esa utilidad se usa para hacer respaldos de la base de datos cuando se requiera.
- Los respaldos pueden ser guardados en otro medio de almacenamiento, ya que no es necesario que estén en un medio de acceso directo. Después de que ha habido una falla en el medio, se usa la parte de restauración de la utilidad para volver a crear la base de datos a partir de un respaldo específico.



# TAREA

Tarea: Tipos de Backup en SQL Server y Oracle ; ademas los Modos de Backup en SQL Server y Oracle



Gracias