

"2015. Año del Bicentenario Luctuoso de José María Morelos y Pavón"

Centro Universitario UAEM Zumpango

Ingeniería en Computación

Unidad de Aprendizaje: Fundamentos de Bases de Datos

Unidad de Competencia V:

Analizar la concurrencia de las transacciones en un Sistema de administración de base de datos

MTE-MI. Rosa Erendira Reyes Luna

Febrero 2015





INTRODUCCIÓN

Los sistemas de bases de datos, según el número de usuarios que pueden utilizarlos de forma concurrente, se clasifican en sistemas monousuario y multiusuario.

Varios usuarios pueden usar un mismo equipo a la vez gracias a la multiprogramación: la computadora puede procesar al mismo tiempo múltiples transacciones.

Si el equipo tiene varias CPU, es posible un procesamiento simultáneo (paralelo) de transacciones. Si sólo hay una CPU, el SO de multiprogramación reparte el tiempo de CPU entre las transacciones: ejecución concurrente intercalada.





PROPÓSITO

Estudiar los conceptos fundamentales de Base de Datos, para el análisis, el diseño y la implementación de un Sistema de Base de Datos. Conocer la forma en que los sistemas de BD implementan la concurrencia y las formas de implementación para mantener la Base de Datos segura y la integridad de los datos. Todo lo anterior, haciendo uso del lenguaje SQL.





UAEM | Universidad Autónoma
del Estado de México.

UNIDAD DE COMPETENCIA V

Analizar la concurrencia de las transacciones en un Sistema de administración de base de datos



www.uaemex.mx



Contenido

1. Transacciones
2. Acciones
3. Recuperación
4. Atomicidad
5. Consistencia
6. Aislamiento





Transacciones

Definición: Una transacción en un Sistema de Gestión de Bases de Datos (SGBD), es un conjunto de órdenes que se ejecutan formando una unidad de trabajo, es decir, en forma indivisible o atómica.





Un SGBD se dice transaccional, si es capaz de mantener la integridad de los datos, haciendo que estas transacciones no puedan finalizar en un estado intermedio. Cuando por alguna causa el sistema debe cancelar la transacción, empieza a deshacer las órdenes ejecutadas hasta dejar la base de datos en su estado inicial (llamado punto de integridad), como si la orden de la transacción nunca se hubiese realizado



Acciones

1.- Acciones lógicas:

Introducir una nueva factura.

Dar de alta a un cliente.

Registrar un cobro.

Cambiar dirección de un proveedor.

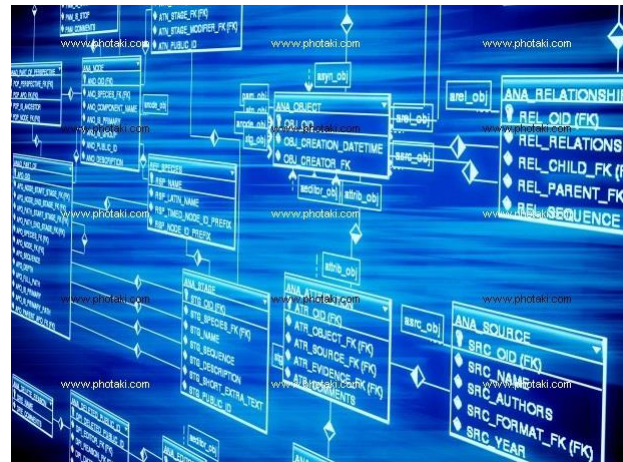
2.- Acciones física: (instrucción insert, delete, update)

Agrega un registro en la tabla factura.

Modifica un campo de un registro de la tabla clientes.

Elimina varias filas de las tablas.

Etc.





Recuperación

Significa que, si se da algún error en los datos, hay un bug de programa ó de hardware, el DBA (Administrador de base de datos) puede traer de vuelta la base de datos al tiempo y estado en que se encontraba en estado consistente antes de que el daño se causara.



Administrador de bases de datos DBA

Un administrador de bases de datos (también conocido como DBA, en inglés database administrator) es aquel profesional que administra las tecnologías de la información y la comunicación, siendo responsable de los aspectos técnicos, tecnológicos, científicos, inteligencia de negocios y legales de bases de datos.



Atomicidad

La atomicidad es la propiedad que asegura que una operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias. Se dice que una operación es atómica cuando es imposible para otra parte de un sistema encontrar pasos intermedios. Si esta operación consiste en una serie de pasos, todos ellos ocurren o ninguno.





Un ejemplo de atomicidad está ordenando un pasaje donde se requieren dos acciones: pago y una reserva de asiento. El pasajero potencial deberá:

- 1.-tanto pagar y reservar un asiento; O
- 2.-No pague ni reservar un asiento.

El sistema de reservas no considera aceptable para un cliente para pagar un boleto sin asegurar el asiento, ni para reservar el asiento sin éxito de pago.

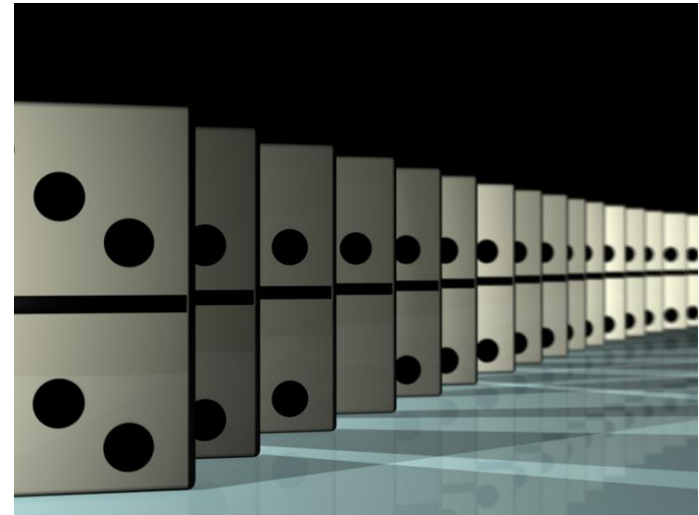
Otro ejemplo es que si uno quiere transferir cierta cantidad de dinero de una cuenta a otra, entonces el usuario comenzaría un procedimiento para hacerlo. Sin embargo, si ocurre una falla, entonces debido a la atomicidad, la cantidad tampoco será transferida completamente o incluso no arrancará. Por lo tanto atomicidad protege al usuario de perder dinero debido a una transacción fallida.





Consistencia

La ejecución aislada de la transacción (es decir, sin otra transacción que se ejecute concurrentemente) conserva la consistencia de la base de datos.



Consistencia

Una transacción finalizada (confirmada parcialmente) puede no confirmarse definitivamente (consistencia).

- Si se confirma definitivamente el sistema asegura la persistencia de los cambios que ha efectuado en la base de datos.
- Si se anulan los cambios que ha efectuado son deshechos.

La ejecución de una transacción debe conducir a un estado de la base de datos consistente (que cumple todas las restricciones de integridad definidas).

- Si se confirma definitivamente el sistema asegura la persistencia de los cambios que ha efectuado en la base de datos.
- Si se anulan los cambios que ha efectuado son deshechos.



Consistencia

Una transacción que termina con éxito se dice que esta comprometida (committed), una transacción que haya sido comprometida llevara a la base de datos a un nuevo estado consistente que debe permanecer incluso si hay un fallo en el sistema. En cualquier momento una transacción solo puede estar en uno de los siguientes estados:

- **Activa (Active):** El estado inicial; la transacción permanece en este estado durante su ejecución.
- **Parcialmente comprometida (Uncommitted):** Después de ejecutarse la ultima transacción.
- **Fallida (Failed)** Tras descubrir que no se puede continuar la ejecución normal.
- **Abortada (Rolled Back):** Después de haber retrocedido la transacción y restablecido la base de datos a su estado anterior al comienzo de la transacción.
- **Comprometida (Committed):** Tras completarse con éxito.





Consistencia

Ejemplo

En cierto caso el requisito de consistencia es que la suma de A y B no sea alterada al ejecutar la transacción. Sin el requisito de consistencia **¡la transacción podría crear o destruir dinero!** Se puede comprobar fácilmente que si una base de datos es consistente antes de ejecutar una transacción sigue siéndolo después de ejecutar dicha transacción.



Consistencia

En resumen

Una base de datos está en un estado consistente si obedece todas las restricciones de integridad definidas sobre ella.

Los cambios de estado ocurren debido a actualizaciones, inserciones y supresiones de información. Por supuesto, se quiere asegurar que la base de datos nunca entre en un estado de inconsistencia.

Durante la ejecución de una transacción, la base de datos puede estar temporalmente en un estado inconsistente.

Se debe asegurar que la base de datos regresa a un estado consistente al fin de la ejecución de una transacción.





- ***Aislamiento:*** es la propiedad que asegura que una operación no pueda afectar a otras. Esto asegura que la realización de 2 transacciones sobre la misma información sean independientes y no generen ningún tipo de error.





Niveles de asilamiento

- Los niveles de asilamiento se describen en cuanto a los efectos secundarios de la simultaneidad que se permiten, como las lecturas desfasadas o ficticias.





- Lectura no comprometida: menor nivel. Asegura que no se lean los datos corruptos físicamente
- Lectura comprometida: solo se permiten lecturas de datos comprometidos.





- Lectura repetible: las lecturas repetidas de la misma fila para la misma transacción das los mismos resultados.
- Secuenciable: mayor nivel de asilamiento. Las transacciones se aíslan completamente.





Niveles de asilamiento menor

- Significa que muchos usuarios pueden tener acceso a los datos simultáneamente, pero también aumenta los efectos de simultaneidad que pueden experimentar, como lecturas no confirmadas o periódica de actualizaciones





Niveles de aislamiento mayor

- El nivel de aislamiento mayor reduce los tipos de efectos de simultaneidad, pero requiere mas recursos del sistema aumenta las posibilidades de que una transacción bloquee otra.





Niveles aislamiento superior

- Es serializable, garantiza que una transacción recupera exactamente las mismo datos cada vez que repita una operación de lectura, aunque para ello aplicara un nivel de bloqueo que pueda afectar a los demás usuarios, y en los sistemas multiusuario



Nivel de aislamiento inferior

- Lectura sin confirmar. Puede recuperar datos modificados pero no confirmados por otras transacciones. En este nivel se pueden producir todos los efectos secundarios de simultaneidad, pero no hay bloquear ni versiones de lectura, por lo que se minimiza la sobrecarga





UAEM | Universidad Autónoma
del Estado de México

- ***Durabilidad:*** Es la propiedad que asegura que una vez realizada la transacción, ésta persistirá y no se podrá deshacer aunque falle el sistema.



CENTRO UNIVERSITARIO UAEM
ZUMPAANGO



www.uaemex.mx



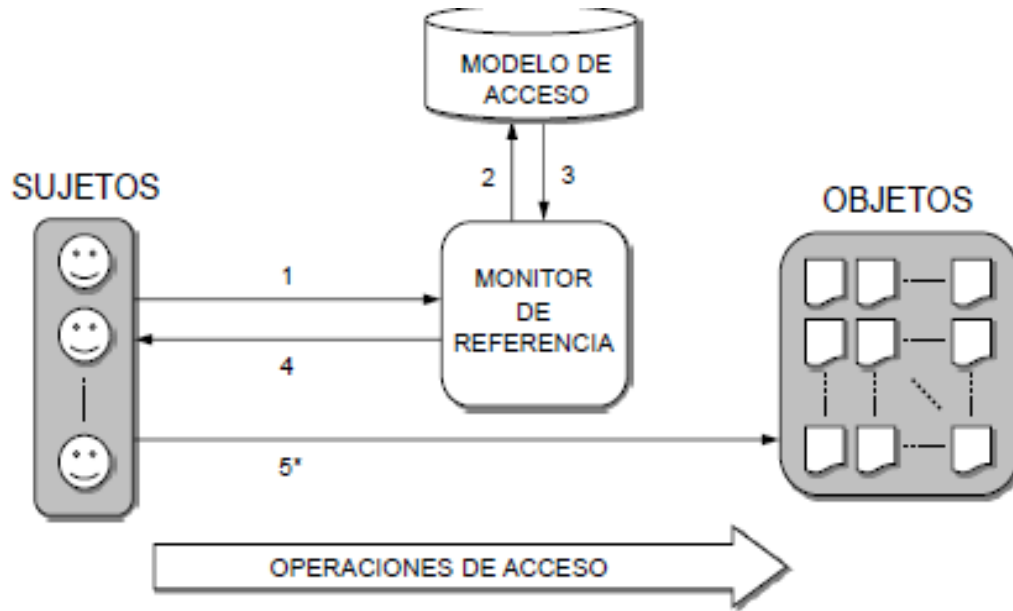
Ejemplo

- Al cumplir la operación bancaria y ocurre una falla del sistema como consecuencia, el sistema es capaz de acceder a dicha operación sin perder ninguna información.





Acceso discrecional





- ***Acceso discrecional:*** El control de acceso discrecional (Discretionary Access Control - DAC) es una política de control de acceso determinada por el dueño de un recurso (una base de datos, un archivo, una impresora, entre otros).
- Es discrecional porque el dueño del recurso decide de manera arbitraria a quien le brinda acceso y que permisos otorga.





Conceptos

- **Propiedad:** cada objeto en el sistema debe tener un dueño. La política de acceso es determinada por el dueño del recurso. En teoría un objeto sin dueño no se encuentra protegido. Normalmente el dueño de un recurso es el usuario que lo crea.
- **Permisos:** derechos de acceso que el dueño de un recurso asigna a usuarios individuales o a grupos de usuarios.



Conclusiones

- Anteriormente una transacción era complicada de hacer ya que se realizaba por medio manual , es decir haciendo transferencias de forma real de una cuenta a otra.
- Los niveles de aislamiento depende del nivel que se este trabajando donde depende de la forma de importancia de cada situación que se nos presenta.

Referencias

1. Date C. J. “Introducción a los Sistemas de Bases de Datos”, Ed. Addison Wesley Longman, México 2001.
2. Silberschatz, Korth, Sudarshan “Fundamentos de Base de Datos”, Mc Graw Hill 2006.
3. Miguel A. Rodríguez, “Bases de datos “, Mc Graw Hill, España 1992
4. R. Bayer, M. Heller y A. Reiser: “Parallelism and Recovery in Database Systems”, ACM TODS 5, No.2 (junio,1980).
5. Luque, I. “Base de Datos desde Chen hasta Codd con Oracle”. Alfa Omega 2002
6. Pérez C. “Oracle PL/SQL “ Alfa Omega 2008





UAEM | Universidad Autónoma
del Estado de México.

GRACIAS

Fundamentos de Bases de Datos

Unidad de Competencia V

MTE-MI. Rosa Erendira Reyes Luna

rereyesl@uaemex.mx



www.uaemex.mx



UAEM | Universidad Autónoma
del Estado de México

GUÍA EMPLEO DE MATERIAL

Fundamentos de Bases de Datos
Unidad de Competencia V



www.uaemex.mx

GUÍA EMPLEO DE MATERIAL

Las primeras diapositivas muestran el propósito, justificación y objetivos de la unidad de aprendizaje. Se presentan para que el alumno identifique dichos elementos.

A partir de la diapositiva 6 se definen las características de un sistema de bases de datos, así como ejemplos de atomicidad, aislamiento, consistencia y recuperación.

La intención de la presentación es esquematizar cada instrucción que se puede ejecutar para administrar la información.

