Índice de Contenidos

Integridad de Las Bases de Datos

- Integridad Semántica
  - Integridad de Clave
  - Integridad Referencial
  - Integridad de Dominio
  - Integridad funcional
    - ✓ Los asertos
    - ✓ Los triggers
- Todos aquellos mecanismos disponibles en el modelo de datos y en el SGBD que permiten que la base de datos se encuentren en un estado consistente con la representación del problema por la que están siendo usadas.
- Son reglas que se activan en las operaciones de inserción, modificación y borrado de elementos de la base de datos

Integridad de Las Bases de Datos

- Los asertos
  - Son predicados que expresan una condición que la base de datos debe satisfacer siempre.
  - ***create assertion <nombre-aserto> check <predicado>***
- La suma de todos los importes de los préstamos de cada sucursal debe ser menor que la suma de todos los saldos de las cuentas de esa sucursal.

```
create assertion restricción-suma check
(not exists (select * from sucursal
    where (select sum(importe) from préstamo
    where préstamo.nombre-sucursal = sucursal.nombre-sucursal) >=
        (select sum (importe) from cuenta
                where préstamo.nombre-sucursal = sucursal.nombre-sucursal)))
```

Integridad de Las Bases de Datos

- Los triggers (*véase Bases de Datos Activas*)
  - Son acciones que el sistema ejecuta de forma automática antes, en lugar de, o después de que se realiza una aoperación que puede ocasionar una modificación de la base de datos.
  - Incluidos en el estándar SQL 1999, son mecanismos útiles ejecutados por el SGBD para alertar a los usuarios o para realizar de manera automática ciertas tareas cuando se cumplen determinadas condiciones.
  - Se conocen bajo el nombre de bases de datos activas (ver Tema) y operan bajo el modelo ECA
    - ✓ Evento: Es el cambio hecho sobre la base de datos que activa al disparador.
    - ✓ Condición: Es la solicitud o prueba que se ejecuta cuando se activa el disparador.
    - ✓ Acción: Es un procedimiento que es ejecutado cuando el disparador es activado y la condición es verdadera.

```
CREATE TRIGGER <trigger name>
<BEFORE|AFTER> <INSERT|DELETE|UPDATE>
ON <relation name>
FOR EACH <ROW|STATEMENT>
EXECUTE PROCEDURE <procedure name>
(<function args>);
```

Integridad de Las Bases de Datos

- Los triggers

```
CREATE TRIGGER iniciar_conteo
BEFORE INSERT ON Estudiantes
DECLARE
    cont INTEGER;
BEGIN
    cont := 0;
END
```

```
CREATE TRIGGER contar
AFTER INSERT ON Estudiantes
WHEN (new.edad < 18)
FOR EACH ROW
BEGIN
    cont := cont + 1;
END
```

Recuperación de las Bases de Datos

- El objetivo del concepto de recuperación es el de proteger la BD contra fallos lógicos y físicos que destruyan los datos en todo o en parte. Independiente de la naturaleza de los fallos estos pueden afectar a dos aspectos del almacenamiento de la Base de Datos, como son:
  - Fallos que provocan la pérdida de memoria volátil
  - Fallos que provocan la pérdida del contenido de memoria secundaria.
- Transacción: *Una secuencia de operaciones que han de ejecutarse en forma atómica, es decir, se realizan todas las operaciones que comprende la transacción o no se realiza ninguna.*
- Las transacciones o terminan con éxito y son grabadas en la base o bien fracasan y debe ser restaurado el estado anterior de la BD.
- El componente del sistema encargado de lograr la atomicidad se conoce como administrador de transacciones y las operaciones COMMIT (comprometer) y ROLLBACK (retroceder) son la clave de su funcionamiento.

Recuperación de las Bases de Datos

- Las características de una transacción son:
  - *Atomicidad (**A**tomicity)*, en el sentido que hemos especificado anteriormente: se ejecutan todas las sentencias o ninguna.
  - *Preservación de la consistencia (**C**onsistent)*: la ejecución de una transacción deja la BD en un estado consistente.
  - *Aislamiento (**I**solation)*, ya que una transacción no muestra los cambios que produce hasta que finaliza.
  - *Persistencia (**P**ersistent)*, ya que una vez que finaliza la transacción con éxito, sus efectos perduran en la BD.
  - *Seriabilidad*, en el sentido de que el efecto de ejecutar transacciones concurrentemente debe ser el mismo que se produciría al ejecutarlas por separado en un orden secuencial según van entrando en el sistema.

Recuperación de las Bases de Datos

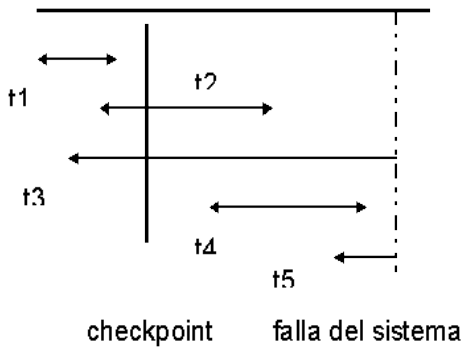
- Para conseguir anular y recuperar transacciones, el método más usado consiste en utilizar un archivo de diario o log en el que va guardando toda la información necesaria para deshacer (en caso de fracasar) o rehacer (en caso de recuperar) las transacciones. Este archivo consta, generalmente, de: Identificador de la transacción, Hora de modificación, Identificador del registro afectado, Tipo de acción, Valor anterior del registro, Nuevo valor del registro, Información adicional.
- Otra alternativa es manejar 2 archivos de *log*, uno con la imagen anterior a las modificaciones y otro con la imagen posterior a las modificaciones.
- El archivo log es usualmente una pila que una vez llena va eliminado registros según van entrando nuevos.

Recuperación de las Bases de Datos

- Un concepto relacionado con los archivos de log es el CHECKPOINT, que permite manejar en forma eficiente el contenido de los archivos log, ya que permiten no tener que recorrer todo el archivo de log, ante fallas.
- El establecimiento de puntos de revisión implica:
  - Grabar físicamente el contenido de los buffers de datos a la base de datos física
  - Grabar físicamente un registro de punto de revisión especial dentro del archivo de log o bitácora
- Los puntos marcados como checkpoint, permiten la recuperación de la base de datos en caliente, es decir, después de la caída del sistema se obtiene la dirección del registro de recuperación más reciente y se recorre el archivo de log desde el punto marcado como checkpoint.

Recuperación de las Bases de Datos

- La transacción t1 no se ve afectada por la falla del sistema, ni por el proceso de recuperación, por haberse completado antes del último punto de recuperación. Las transacciones t2 y t4, a pesar de haber terminado no han sido grabadas en la base de datos, ya que éstas serían cometidas en un checkpoint. Las transacciones t3 y t5 deberán rehacerse ya que no han concluido.
- El procedimiento que deberá realizar el sistema al reiniciarse consiste en:
  - Comenzar con dos listas de transacciones, la lista ANULAR y la lista REPETIR. Igualar la lista ANULAR a la lista de todas las transacciones incluidas en el registro de punto de revisión. Dejar vacía la lista REPETIR.
  - Examinar la bitácora hacia adelante a partir del registro de punto de revisión.
  - Si se encuentra una entrada de bitácora de "iniciar transacción" para la transacción T, añadir T a la lista ANULAR.
  - Si se encuentra una entrada una entrada de bitácora "comprometer" para la transacción T, pasar esa transacción de la lista ANULAR a la lista REPETIR.
  - Cuando se llegue al final de la bitácora, las listas ANULAR y REPETIR identificarán, respectivamente, las transacciones de los tipos T3 y T5 y las de los tipos T2 y T4.
- Posteriormente el sistema revisará la bitácora hacia atrás, anulando todas las transacciones de la lista ANULAR. A continuación la revisará hacia adelante, realizando de nuevo todas las transacciones en la lista REPETIR. Por último, una vez terminada todas las actividades de recuperación, el sistema estará listo para aceptar nuevos trabajos.



Concurrencia

- En sistemas multiusuario, es necesario un mecanismo para controlar la concurrencia. Se pueden producir inconsistencias importantes derivadas del acceso concurrente, como por ejemplo, el problema de la operación perdida.
- Técnicas de Bloqueo: está basada en una variable asociada a cada elemento de datos que describe el estado de dicho elemento respecto a las posibles operaciones (recuperación o actualización) que se pueden realizar sobre ellos en cada momento
  - Exclusivos: cuando una transacción mantiene un bloqueo de este tipo, ninguna otra transacción puede acceder a el objeto bloqueado, ni bloquearlo, hasta que sea liberado por la transacción que lo había retenido. Se utiliza cuando se quiere actualizar datos.
  - Compartidos: cuando una transacción bloquea en este modo, permite que otras transacciones retengan también el objeto en bloque compartido, pero no exclusivo. Este tipo se utiliza cuando no se requiere actualizar datos, pero se desea impedir cualquier modificación mientras los datos son consultados.

Concurrencia

- El algoritmo que se utiliza se llama bloqueo de dos fases (two phase locking).
- El problema de las técnicas de bloqueo es que puede producirse un interbloqueo (deadlock), dos o más transacciones están esperando cada una de ellas que la otra libere algún objeto antes de seguir, lo que se puede solucionar con algunas técnicas como:
  - *Prevenir el deadlock:* obliga a que las transacciones bloqueen todos los elementos que necesitan por adelantado. EN caso de no poder conseguir todos esos elementos no bloquea ninguno y se queda en espera hasta volver a intentarlo.
  - *Detectar el deadlock:* Se controla de forma periódica si se ha producido un deadlock. Se construye un grafo en espera, cada nodo es una transacción en ejecución y un arco de una transacción Ti a Tj, en caso que Ti esté esperando un elemento que ocupa Tj. Si existe un ciclo en el grafo tenemos un deadlock. La solución es escoger transacciones víctimas y deshacerlas, hasta que desaparezca el deadlock. Cada SGBD tiene políticas diferentes para escoger víctimas.
- *Granularidad* muy gruesa implica gestionar menor número de bloqueos, pero retrasa la ejecución de muchas transacciones (los objetos no se van liberando). Una granularidad muy fina, permite mayor concurrencia, pero aparecen mas situaciones de deadlock que han de ser resueltas.

Concurrencia

- Técnicas de marcas de tiempo
  - Las marcas de tiempo son identificadores únicos que se asignan a las transacciones, que se consideran como el tiempo de inicio de una transacción. Con esta técnica no existen bloqueos, las transacciones se ordenan en función de su marca de tiempo y se ejecutan o se retrasan.
- Técnicas optimistas
  - Las transacciones acceden libremente a los elementos, y antes de finalizar se determina si ha habido interferencias. Este tipo de técnicas considera que las transacciones tienen 3 fases:
    - ✓ *Lectura*: las transacciones realizan operaciones sobre copias privadas de los objetos (accesibles solo por la transacción)
    - ✓ *Validación* : en la que se comprueba si el conjunto de objetos modificados por una transacción se solapa con el conjunto de objetos modificados por alguna otra que haya hecho la validación durante la fase de lectura de dicha transacción
    - ✓ *Grabación*: en el caso de no detectar interferencias se graban las modificaciones, convirtiendo las versiones privadas de los objetos en versiones actuales.

Seguridad y Privacidad de las Bases de Datos

- La seguridad de las bases de datos es una área amplia que abarca varios temas, entre ellos se encuentran los siguientes:
  - Cuestiones éticas y legales relativas al derecho de tener acceso a cierta información
  - Cuestiones de política a nivel gubernamental, institucional o corporativo, relacionadas con el tipo de información que no debe estar disponible para el publico
  - Cuestiones relacionadas con el sistema, como los niveles del sistema en que deben manejarse diversas funciones de seguridad
- Las necesidades en las organizaciones de identificar múltiples niveles de seguridad y clasificar los datos y los usuarios según estos niveles
- La seguridad de las bases de datos se refiere a la protección frente a accesos malintencionados. Para proteger la base de datos hay que adoptar medidas de seguridad en varios niveles:
  - Sistema de bases de datos, Sistema operativo, Red, Físico, Humano

Seguridad y Privacidad de las Bases de Datos

- En relación al SGBD, debe mantener información de los usuarios, su tipo y los accesos y operaciones permitidas a éstos.
- Tipos de usuarios:
  - DBA, están permitidas todas las operaciones, conceder privilegios y establecer usuarios
  - Usuario con derecho a crear, borrar y modificar objetos y que además puede conceder privilegios a otros usuarios sobre los objetos que ha creado.
  - Usuario con derecho a consultar, o actualizar, y sin derecho a crear o borrar objetos.
- Privilegios sobre los objetos, añadir nuevos campos, indexar, alterar la estructura de los objetos, etc. Los SGBD tienen opciones que permiten manejar la seguridad, tal como GRANT, REVOKE, etc. También tienen un archivo de auditoría en donde se registran las operaciones que realizan los usuarios.
- Otro mecanismo de seguridad que ofrecen los SGBD en entregar información a los usuarios a través de vistas (CREATE VIEW)

Seguridad y Privacidad de las Bases de Datos

- En relación al SGBD, debe mantener información de los usuarios, su tipo y los accesos y operaciones permitidas a éstos.
- Tipos de usuarios:
  - DBA, están permitidas todas las operaciones, conceder privilegios y establecer usuarios
  - Usuario con derecho a crear, borrar y modificar objetos y que además puede conceder privilegios a otros usuarios sobre los objetos que ha creado.
  - Usuario con derecho a consultar, o actualizar, y sin derecho a crear o borrar objetos.
- Privilegios sobre los objetos, añadir nuevos campos, indexar, alterar la estructura de los objetos, etc. Los SGBD tienen opciones que permiten manejar la seguridad, tal como GRANT, REVOKE, etc. También tienen un archivo de auditoría en donde se registran las operaciones que realizan los usuarios.
- Otro mecanismo de seguridad que ofrecen los SGBD en entregar información a los usuarios a través de vistas (CREATE VIEW)

Seguridad de las Bases de Datos

- Las vistas: un medio de proporcionar a un usuario un modelo personalizado de la base de datos. Una vista puede ocultar los datos que un usuario no necesita ver. La capacidad de las vistas para ocultar datos sirve para simplificar el uso del sistema y para mejorar la seguridad.

```
CREATE VIEW mayoresedad AS
  SELECT * FROM estudiante
  WHERE edad>18;
```

  - La creación de vistas no necesita la autorización de recursos.
  - El usuario que crea una vista no recibe necesariamente todos los privilegios sobre la misma. Ese usuario sólo recibe los privilegios que no proporcionan autorizaciones adicionales respecto de las que ya posee.
  - Si un usuario crea una vista sobre la que no se puede conceder ninguna autorización, se deniega la solicitud de creación de la vista.
- Los privilegios y los papeles
  - Privilegios de autorización
    - ✓ LEER, ESCRIBIR, EJECUTAR
    - ✓ SELECCIONAR, INSERTAR, ACTUALIZAR, REFERENCIAR, INDEXAR

Seguridad de las Bases de Datos

- Un ejemplo general de asignación de privilegios sería de la forma:

  grant *<lista-privilegios>* ***on*** *<lista-relaciones> to <lista-usuarios>*

- Papel (rol): Un papel define un tipo de usuario de la base de datos que tiene concedidos una serie de autorizaciones sobre la misma.
- Los papeles permiten simplificar la concesión de privilegios a los usuarios de forma individualizada, asignándoles un papel que tiene asignados esos privilegios.
- Un ejemplo general de creación de papeles y asignación de privilegios sería de la forma:

  create *role <nombre-papel>*

  grant *<lista-privilegios>* ***on*** *<lista-relaciones> to <lista-papeles>*

  grant *<nombre-papel>* ***to*** *<lista-usuarios>*

Seguridad y Privacidad de las Bases de Datos

- La autenticación se refiere a la tarea de verificar la identidad de una persona o software que se conecte a una base de datos. La forma más simple consiste en una contraseña secreta que se debe presentar cuando se abra una conexión a la base de datos.
  - La autenticación basada en palabras clave se usa ampliamente por los sistemas operativos y bases de datos. Sin embargo, el uso de contraseñas tiene algunos inconvenientes, especialmente en una red. Si un husmeador es capaz de «oler» los datos que circulan por la red, puede ser capaz de encontrar la contraseña que se está enviando por la red.
  - Un esquema más seguro es el sistema de desafío/respuesta. El sistema de bases de datos envía una cadena de desafío al usuario. El usuario cifra la cadena de desafío usando una contraseña secreta como clave de cifrado y devuelve el resultado.
  - Este esquema asegura que las contraseñas no circulen por la red. Los sistemas de clave pública se pueden usar para cifrar en un sistema de desafío-respuesta. El sistema de bases de datos cifra una cadena de desafío usando la clave pública del usuario y lo envía al usuario. Éste descifra la cadena con su clave privada y devuelve el resultado al sistema de bases de datos. El sistema de bases de datos comprueba entonces la respuesta. Este esquema tiene la ventaja añadida de no almacenar la contraseña en la base de datos, donde podría ser vista potencialmente por administradores del sistema.
- Otra aplicación interesante de la criptografía está en las firmas digitales para verificar la autenticidad de los datos. La clave privada se usa para firmar los datos y los datos firmados se pueden hacer públicos. Cualquiera podría verificarlos con la clave pública, pero nadie podría haber generado los datos codificados sin tener la clave privada. Además, las firmas digitales también sirven para asegurar el rechazo. Es decir, en el caso de que una persona que creó los datos afirmase más tarde que no lo hizo (el equivalente electrónico de afirmar que no se ha firmado un talón) se puede probar que esa persona ha creado los datos (a menos que haya cedido su clave privada a otros).

Seguridad y Privacidad de las Bases de Datos

- Otra aplicación interesante de la criptografía está en las firmas digitales para verificar la autenticidad de los datos; las firmas digitales desempeñan el papel electrónico de las firmas físicas en los documentos. La clave privada se usa para firmar los datos y los datos firmados se pueden hacer públicos. Cualquiera podría verificarlos con la clave pública, pero nadie podría haber generado los datos codificados sin tener la clave privada. Por tanto, se puede comprobar que los datos fueron creados realmente por la persona que afirma haberlos creado.
- Además, las firmas digitales también sirven para asegurar el rechazo. Es decir, en el caso de que una persona que creó los datos afirmase más tarde que no lo hizo (el equivalente electrónico de afirmar que no se ha firmado un talón) se puede probar que esa persona ha creado los datos (a menos que haya cedido su clave privada a otros).