**Administración de transacciones**

**Uriel Gustavo Padilla Bustamante**

**Introducción.**

El tema que a continuación se presenta Administración de transacciones se describirán las propiedades de las transacciones, así como las propiedades de del procesamiento de estas. También se abarcará el tema del control de concurrencias y el manejo de recuperaciones.

Una transacción se puede considerar una unidad lógica de trabajo en la base de datos. Puede ser un programa completo, fragmento de programa o comando único. Involucra cualquier número de operaciones en la base de datos.

Todas las transacciones deben mostrar cuatro propiedades importantes, por lo general llamadas **ACID**, que es un acrónimo en inglés de las propiedades siguientes: (atómica,consistente, aislada (isolated) y durable).

* **Atómica** significa que una transacción no se puede subdividir.
* **Consistente** significa que, si las limitaciones aplicables son ciertas antes de empezar la transacción, éstas también lo serán al terminarla.
* **Aislada** significa que las transacciones no interfieren entre sí, excepto en formas permitidas.
* **Durable** significa que cualquier cambio que resulte de una transacción es permanente. Ninguna falla va a borrar ningún cambio después de terminar la transacción.

Los DBMS ofrecen dos servicios para garantizar que las transacciones tienen las propiedades ACID: transparencia de recuperación y transparencia de concurrencia.

* La **transparencia de recuperación** significa que el DBMS restaura automáticamente una base de datos a un estado consistente después de una falla.
* La **transparencia de concurrencia** significa que los usuarios perciben la base de datos como un sistema para un solo usuario, aunque haya varios usuarios simultáneos.

**Objetivo del control de concurrencia**

El objetivo del control de la concurrencia es maximizar el caudal de procesamiento de transacciones y, al mismo tiempo, evitar la interferencia entre varios usuarios. El caudal de procesamiento de transacciones, el número de transacciones procesadas por unidad de tiempo, es una importante medida del trabajo que realiza un DBMS.

Las transacciones que se ejecutan de manera simultánea no pueden interferirse entre sí, a

menos que manipulen datos comunes. La mayor parte de las transacciones concurrentes manipulan sólo pequeñas cantidades de datos comunes.

Un punto decisivo son los datos comunes que varios usuarios tratan de cambiar al mismo tiempo. En esencia, un punto decisivo representa un recurso escaso por el que los usuarios deben esperar.

**Problemas de interferencia**

Hay tres problemas que se pueden presentar debido al acceso simultáneo a una base de datos:

*Actualización perdida*

Una actualización perdida es el problema de interferencia más serio porque los cambios a una base de datos se pierden sin que nadie se dé cuenta. En una actualización perdida, la actualización de un usuario sobrescribe la de otro.

Una actualización perdida comprende dos o más transacciones que tratan de cambiar (escribir

en) la misma parte de la base de datos.

*Dependencia sin realización*

Una dependencia sin realización sucede cuando una transacción lee los datos que escribe otra

antes de que la otra transacción se realice. Una dependencia sin realización también se conoce

como lectura sucia porque la provoca una transacción que lee datos sucios (sin realizar).

*Problemas que comprenden recuperaciones inconsistentes*

Todos los problemas de recuperaciones inconsistentes comprenden una transacción que lee y una segunda transacción que cambia la misma parte de la base de datos.

Herramientas de control de concurrencia

*Candados*

Los candados ofrecen una forma de evitar que otros usuarios tengan acceso a un elemento de

una base de datos que está en uso.

*Granularidad de candados*

La granularidad de candados representa una complicación en relación con los candados. La

granularidad se refiere al tamaño del elemento de la base de datos que tiene candado. La mayor

parte de los DBMS manejan candados de distintas granularidades.

*Puntos muertos*

El uso de candados para evitar problemas de interferencia puede dar lugar a puntos muertos. Un

punto muerto es un problema de espera mutua. Una transacción tiene un recurso que necesita

una segunda transacción, y esta última ocupa un recurso que la primera necesita. La mayor parte de los DBMS de escritorio emplean una política más sencilla de tiempo fuera para controlar los puntos muertos.

*Protocolo de candado en dos etapas*

Para asegurarse de que no se presenten problemas de actualizaciones perdidas, el administrador

de control de concurrencia pide que todas las transacciones sigan el protocolo de candados en

dos etapas (2PL). *Protocolo* es un término elegante para referirnos a una regla de comportamiento

de grupo.

*Planteamientos optimistas*

El uso de candados y el 2PL es un planteamiento pesimista para el control de concurrencias. El uso de candados supone que todas las transacciones entran en conflicto. Los planteamientos de control de concurrencia optimistas suponen que los conflictos son raros. Si es así, es más eficiente revisar la presencia de conflictos que utilizar candados que obliguen

a esperar.

Administración de recuperaciones

La administración de recuperaciones es un servicio que restaura la base de datos a un estado

consistente después de una falla.

Dispositivos de almacenamiento de datos y tipos de fallas

Desde la perspectiva de las fallas en las bases de datos, la volatilidad es una característica importante de los dispositivos para almacenamiento de datos.

La memoria principal es volátil porque pierde su estado en caso de que no haya corriente eléctrica. En contraste, un disco duro es no volátil porque conserva su estado aun cuando no haya corriente eléctrica.

Algunas fallas afectan sólo la memoria principal, mientras que otras afectan la memoria

volátil y la no volátil.

Herramientas de recuperación

Tres de las herramientas: registro de transacciones, punto de revisión y respaldo de base de datos.

*Registro de transacciones*

Un registro de transacciones proporciona el historial de los cambios en la base de datos. Cada

cambio realizado a una base queda asentado también en el registro. El registro es una tabla oculta

que no está disponible para los usuarios normales.

*Control*

El propósito de un control es reducir el tiempo de recuperación de las fallas. A intervalos, se

escribe un registro de control en el registro para guardar todas las transacciones activas. Además, todos los búfers de registro y algunos de la base de datos se escriben en el disco. En el momento de reiniciar, el administrador de recuperaciones depende del registro del control y de su conocimiento sobre las escrituras en el registro y las páginas de la base de datos para reducir la cantidad de trabajo de reinicio.

*Escritura forzada*

La habilidad de controlar el momento en que los datos se transfieren a un almacenamiento no

volátil se conoce como escritura forzada. Sin esta capacidad, la recuperación no es posible.

La escritura forzada significa que el DBMS, y no el sistema operativo, controla el momento de escribir los datos en un almacenamiento no volátil.

*Respaldo de base de datos*

Un respaldo es una copia de todo un disco o parte de éste. El respaldo se utiliza cuando está

dañado el disco que contiene la base de datos o el registro.

Aspectos del diseño de transacciones

Un diseñador de transacciones desarrolla una aplicación para realizar parte del procesamiento

de una base de datos. Por lo general, el diseñador de transacciones tiene la opción de realizar una transacción extensa que contiene todos los enunciados SQL, o dividir estos enunciados en varias transacciones menores.

Niveles de aislamiento

El nivel de aislamiento especifica el grado en el que una transacción se separa de las acciones

de otras transacciones. Un diseñador de transacciones puede equilibrar el trabajo adicional

del control de concurrencia con los problemas de interferencia potenciales especificando el nivel

de aislamiento apropiado.