



“TRANSACCIONES, BLOQUEOS Y NIVELES DE AISLAMIENTO”

TALLER DE BASE DE DATOS

INGENIERIA EN SISTEMAS COMPUTACIONALES

PRESENTA:

JESÚS ALFONSO CUEVAS ÁVILA

JIQUILPAN, MICHOACÁN, NOVIEMBRE DE 2024





Contenido

¿Qué son las transacciones?	3
¿Cómo aplico las transacciones a mi proyecto?.....	4
¿Qué son los tipos de bloqueos?	5
¿Qué son los niveles de aislamiento y cómo funcionan?	7



¿Qué son las transacciones?

Son operaciones SQL que se llevan a cabo en bloques, para evitar distorsiones en los datos obtenidos de las consultas a causa de consultas intermedias entre modificaciones. Estas operaciones deben cumplir con determinadas características que se agrupan bajo el acrónimo ACID (Atomicity, Consistency, Isolation y Durability):

1. **Atomicidad (Atomicity):** La transacción se realiza en su totalidad o no se realiza en absoluto. Si alguna operación falla, se revierte todo el conjunto de cambios.
2. **Consistencia (Consistency):** La transacción debe dejar la base de datos en un estado consistente. Después de que se complete la transacción, los datos deben cumplir con todas las reglas de integridad y restricciones establecidas.
3. **Aislamiento (Isolation):** Las operaciones dentro de una transacción son independientes de otras transacciones en ejecución simultánea, evitando que se vean afectadas o lean datos intermedios que aún no están confirmados.
4. **Durabilidad (Durability):** Una vez que una transacción se ha confirmado (mediante un COMMIT), los cambios realizados son permanentes, incluso en caso de fallas del sistema.



¿Cómo aplico las transacciones a mi proyecto?

En mis tablas, las transacciones son utilizadas para evitar el cumplimiento de determinadas operaciones en base a ciertas condiciones. Ejemplo de este comportamiento puede verse en las transacciones involucrando 'Apartados' o 'Préstamos' las cuales siguen un patrón de comportamiento muy similar basado en el historial de préstamos.

Ambas transacciones frenan el curso de las mismas en el caso de que exista una reincidencia en el retraso de las devoluciones de préstamos realizados previamente, pues prestar o permitir el apartado de un manga está estrechamente relacionado con el "crédito" que el cliente construye mediante la devolución de los mangas en tiempo y forma.

También relacionado al historial de devoluciones del cliente está la transacción realizada para la deshabilitación de un cliente dependiendo de si presenta una reincidencia en los préstamos no devueltos a tiempo, para de plano deshabilitar su cuenta en vez de una simple advertencia.

En resumen, en mi participación dentro del proyecto de la biblioteca de mangas, se abarcan más transacciones de restricción en base al control del historial del cliente



¿Qué son los tipos de bloqueos?

El bloqueo es el proceso mediante el cual un sistema de administración de bases de datos restringe el acceso a una fila en un entorno multiusuario. (*Microsoft*)

Pero ya en palabras más mundanas, son impedimentos que se pueden aplicar tanto a tablas o filas con el fin de evitar lecturas sucias, lecturas fantasmas o lecturas no repetibles, en conjunto, lecturas que devuelvan (o no) datos que no se correspondan con los datos reales.

Estos bloqueos se dividen de la siguiente forma:

1. Bloqueo Compartido (Shared Lock - S)

- **Función:** Permite que múltiples transacciones lean un recurso (por ejemplo, una fila o una tabla) sin que ninguna pueda modificarlo.
- **Restricción:** Impide que otra transacción obtenga un bloqueo exclusivo sobre el mismo recurso hasta que el bloqueo compartido se libere.

2. Bloqueo Exclusivo (Exclusive Lock - X)

- **Función:** Permite que una única transacción modifique un recurso.
- **Restricción:** Bloquea el acceso a otras transacciones, ya que ninguna otra transacción puede ni leer ni modificar el recurso hasta que se libere el bloqueo exclusivo.

3. Bloqueo de Intención (Intent Lock - I)



- **Función:** Permite que el sistema declare su intención de adquirir un bloqueo compartido o exclusivo en niveles más bajos de una jerarquía (por ejemplo, en una tabla y después en filas específicas).
- **Tipos de bloqueo de intención:**
 - **Intent Shared (IS):** Indica que la transacción desea adquirir un bloqueo compartido en algún nivel inferior.
 - **Intent Exclusive (IX):** Indica que la transacción desea adquirir un bloqueo exclusivo en algún nivel inferior.
 - **Shared Intent Exclusive (SIX):** Indica que la transacción ha bloqueado de forma compartida un recurso, pero también quiere bloqueos exclusivos en niveles inferiores.

4. Bloqueo de Actualización (Update Lock - U)

- **Función:** Permite a una transacción preparar un recurso para una actualización, evitando que otras transacciones obtengan un bloqueo exclusivo.
- **Restricción:** No permite que otra transacción obtenga un bloqueo exclusivo, pero sí permite bloqueos compartidos.





¿Qué son los niveles de aislamiento y cómo funcionan?

Controlan los efectos de las operaciones concurrentes y ayudan a evitar problemas como lecturas sucias, lecturas no repetibles y lecturas fantasma.

Los niveles de aislamiento más comunes son **Read Uncommitted**, **Read Committed**, **Repeatable Read**, **Serializable**, y en Microsoft SQL Server, se incluyen también **Snapshot** y **Read Committed Snapshot**.

Las funciones de cada uno de los niveles de aislamiento son las siguientes:

1. Read Uncommitted (Lectura No Confirmada)

- **Descripción:** Permite que una transacción lea datos modificados por otras transacciones antes de que estas se hayan confirmado.
- **Problemas permitidos:** Puede causar lecturas sucias, lo que significa que una transacción podría leer datos que luego se revierten si la transacción original realiza un rollback.

2. Read Committed (Lectura Confirmada)

- **Descripción:** Permite leer solo datos confirmados. Las transacciones no pueden ver los cambios realizados por otras transacciones hasta que estos sean confirmados.
- **Problemas permitidos:** Evita lecturas sucias, pero permite lecturas no repetibles (el valor de un dato leído una vez puede cambiar si otra transacción lo modifica antes de que la primera termine).



3. Repeatable Read (Lectura Repetible)

- **Descripción:** Asegura que los datos leídos una vez durante la transacción no cambien si se leen de nuevo en la misma transacción. Bloquea los datos leídos para evitar que otros los modifiquen hasta que la transacción termine.
- **Problemas permitidos:** Evita lecturas sucias y lecturas no repetibles, pero aún permite lecturas fantasmas (nuevas filas pueden ser insertadas por otras transacciones en el rango consultado).

4. Serializable

- **Descripción:** Es el nivel de aislamiento más alto. Asegura que la ejecución de transacciones concurrentes sea equivalente a ejecutarlas secuencialmente una tras otra, evitando lecturas sucias, lecturas no repetibles y lecturas fantasma.
- **Problemas permitidos:** No permite que ocurra ningún problema, pero también es el más costoso en términos de rendimiento.

5. Snapshot (Microsoft SQL Server)

- **Descripción:** Permite a las transacciones ver una versión consistente de los datos en el momento en que la transacción comenzó, sin bloquear las filas que están siendo leídas. Saca una “foto” de los datos para cada transacción, la cual se almacena en caché.

6. Read Committed Snapshot (Microsoft SQL Server)

- **Descripción:** Similar a **Read Committed**, pero permite lecturas confirmadas sin bloquear, empleando el mismo mecanismo de la “foto” utilizado en snapshot.
- **Problemas permitidos:** Evita lecturas sucias y ofrece una mayor concurrencia que el modo **Read Committed** tradicional, pero aún permite lecturas no repetibles.

