

# Tarea 12. Niveles de aislamiento de bases de datos y propiedades ACID.

Lemus González, Javier Issac  
[javierissaclemusgonzalez@gmail.com](mailto:javierissaclemusgonzalez@gmail.com)  
 Universidad Nacional Autónoma de México

## I. INTRODUCCIÓN

Una transacción representa una unidad de trabajo ejecutada de manera integral sobre una base de datos. Consiste en una serie de declaraciones SQL que, aunque ejecutadas individualmente, son parte de un todo cohesivo. El principio fundamental de una transacción es que todas sus operaciones deben completarse con éxito; de lo contrario, ninguna se llevará a cabo. Esto se resume en el concepto de "todo o nada".

El uso de transacciones es crucial en escenarios donde es necesario garantizar la integridad de las operaciones compuestas por múltiples declaraciones SQL. Un ejemplo clásico es la actualización concurrente de saldos en dos cuentas bancarias durante una transferencia de fondos. En tales situaciones, las transacciones aseguran que, si alguna de las operaciones falla, el sistema revertirá todas las acciones previas, manteniendo así la consistencia de la base de datos.

Aunque es posible operar SQL sin el mecanismo de transacciones, su aplicación se vuelve indispensable cuando se requieren garantías de atomicidad y consistencia. El uso de transacciones no solo facilita el manejo de complejas secuencias de operaciones sino que también fortalece la fiabilidad y la integridad de los sistemas de gestión de bases de datos.

## II. PROPIEDADES ACID DE LAS TRANSACCIONES

Las transacciones en las bases de datos están diseñadas para garantizar la integridad y la fiabilidad de los datos mediante el cumplimiento de cuatro propiedades fundamentales, conocidas como ACID: Atomicidad, Consistencia, Aislamiento y Durabilidad.

### A. Atomicidad

La Atomicidad asegura que una transacción se ejecute en su totalidad o no se ejecute en absoluto. Esto significa que, o bien todas las operaciones dentro de la transacción se completan con éxito, o, en caso de fallo, todas se revierten, manteniendo la base de datos sin cambios. Por ejemplo, en una transferencia bancaria, ambas operaciones de actualización de cuentas deben ejecutarse juntas para que la transacción sea atómica.

### B. Consistencia

La Consistencia garantiza que cada transacción lleve la base de datos de un estado válido a otro, preservando la integridad de los datos. Las restricciones de la base de datos, como claves primarias y externas, se mantienen, asegurando que no haya corrupción ni pérdida de datos.

### C. Aislamiento

El Aislamiento asegura que las operaciones de una transacción no sean visibles para otras transacciones hasta que se hayan completado. Esto previene la interferencia entre transacciones simultáneas, garantizando que los resultados sean consistentes como si se ejecutaran secuencialmente.

### D. Durabilidad

La Durabilidad se refiere a la permanencia de los resultados de una transacción completada. Una vez que una transacción se ha comprometido, sus efectos son permanentes, incluso en caso de fallos del sistema. Esto se asegura mediante mecanismos como el almacenamiento en disco y las copias de seguridad.

Estas propiedades son esenciales para el manejo efectivo y seguro de las operaciones en sistemas de bases de datos, asegurando que los datos sean consistentes, aislados y duraderos, manteniendo así la integridad y la confiabilidad del sistema.

## III. NIVELES DE AISLAMIENTO EN SQL

Los niveles de aislamiento en SQL definen cómo las transacciones interactúan con los datos de otras transacciones, enfocándose en la visibilidad de los datos y el manejo de los "fenómenos de lectura". Estos fenómenos incluyen situaciones no deseadas como lecturas sucias, actualizaciones perdidas, lecturas no repetibles y lecturas fantasma, que pueden surgir durante la ejecución concurrente de transacciones.

### A. Fenómenos de Lectura

- **Lectura Sucia:** Ocurre cuando una transacción lee datos modificados por otra transacción que aún no se ha confirmado.
- **Actualización Perdida:** Sucede cuando los cambios realizados por una transacción son sobrescritos por otra transacción.
- **Lectura No Repetible:** Se da cuando una transacción lee la misma fila dos veces y obtiene resultados diferentes debido a las modificaciones por otras transacciones.

- Lectura Fantasma: Ocurre cuando los resultados de una consulta cambian debido a la inserción o eliminación de filas por otras transacciones.

#### *B. Niveles de Aislamiento*

- Leer No Comprometido (Read Uncommitted): Permite lecturas sucias, siendo el nivel más bajo de aislamiento.
- Leer Comprometido (Read Committed): Impide lecturas sucias permitiendo solo la lectura de datos confirmados. Es el nivel predeterminado en Oracle y SQL Server.
- Lecturas Repetibles (Repeatable Read): Asegura que si una transacción lee una fila, ninguna otra transacción puede modificarla hasta que la primera finalice, previniendo lecturas no repetibles. Predominante en MySQL.
- Serializable: El nivel más alto de aislamiento. Garantiza total aislamiento, impidiendo lecturas sucias, actualizaciones perdidas, lecturas no repetibles y lecturas fantasma. Es el estándar en PostgreSQL.

Cada nivel de aislamiento ofrece un compromiso entre el rendimiento y el grado de aislamiento, afectando directamente la concurrencia y la consistencia de las transacciones en la base de datos.

#### REFERENCIAS

- [1] WilliamDAssafMSFT, "SET TRANSACTION ISOLATION LEVEL (Transact-SQL) - SQL Server." <https://learn.microsoft.com/en-us/sql/t-sql/statements/set-transaction-isolation-level-transact-sql?view=sql-server-ver16>
- [2] Pulido Romero, E., Escobar Domínguez, Ó., & Núñez Pérez, J. Á., "Base de datos," Grupo Editorial Patria, 2019.