UNAM FACULTAD DE INGENIERIA

Bases de Datos

Tarea 07

Martínez García José Eduardo

DATABASE INSOLATION LEVELS AND ACID PROPERTIES

In a database management system, transaction isolation levels define the degree to which the operations in one transaction are isolated from the operations of other concurrent transactions. In other words, it defines how and when the changes made by one transaction are visible to others to assure data consistency and integrity. higher levels of isolation give higher accuracy, but possibly at a slower speed [1].

Isolation levels define the degree to which a transaction must be isolated from the data modifications made by any other transaction in the database system [1].

A transaction isolation level is defined by the following phenomena:

Dirty Read – A Dirty read is a situation when a transaction reads data that has not yet been committed [1].

Non Repeatable read – Non-repeatable read occurs when a transaction reads the same row twice and gets a different value each time [1].

Phantom Read – Phantom Read occurs when two same queries are executed, but the rows retrieved by the two, are different [1].

DATABASE INSOLATION LEVEL

The SQL standard defines four isolation levels:

1. **Read Uncommitted**

This is the lowest isolation level, where transactions can read data that other transactions have modified but not yet committed. This leads to possible dirty reads, as well as non-repeatable reads and phantom reads, since transactions are not isolated from each other [1].

2. **Read Committed**

At this level, a transaction can only read committed data, eliminating dirty reads. However, non-repeatable reads and phantom reads can still occur. The database applies locks on rows being read or written, preventing other transactions from modifying them during the operation.

3. **Repeatable Read**

This level ensures that any data read during a transaction remains the same throughout its execution, even if other transactions commit changes. It prevents dirty reads and non-repeatable reads by placing read and write locks on accessed rows. However, phantom reads — where new rows matching a query are added by other transactions — may still occur [1].

4. **Serializable**

This is the highest isolation level, where transactions are executed as if they occur one after another (serially). It prevents dirty reads, non-repeatable reads, and phantom reads, offering full isolation. This strict control ensures consistency but may reduce concurrency and performance [1].

ACID PROPERTIES

A transaction in DBMS refers to a sequence of operations performed as a single unit of work. These operations may involve reading or writing data to the database. To maintain data integrity, DBMS ensures that each transaction adheres to the ACID properties [2].

1. **Atomicity: "All or Nothing"**

Ensures that a transaction is atomic, it means that either the entire transaction completes fully or doesn't execute at all. There is no in-between state, transactions do not occur partially. If a transaction has multiple operations, and one of them fails, the whole transaction is rolled back, leaving the database unchanged. This avoids partial updates that can lead to inconsistency [2].

2. **Consistency: Maintaining Valid Data States**

ensures that a database remains in a valid state before and after a transaction. It guarantees that any transaction will take the database from one consistent state to another, maintaining the rules and constraints defined for the data. A transaction should only take the database from one valid state to another [2].

3. Isolation: Ensuring Concurrent Transactions Don't Interfere

ensures that multiple transactions can occur concurrently without leading to the inconsistency of the database state. Transactions occur independently without interference. Changes occurring in a particular transaction will not be visible to any other transaction until that particular change in that transaction is written to memory or has been committed [2].

4. Durability: Persisting Changes

This property ensures that once the transaction has completed execution, the updates and modifications to the database are stored in and written to disk and they persist even if a system failure occurs. These updates now become permanent and are stored in non-volatile memory. In the event of a failure, the DBMS can recover the database to the state it was in after the last committed transaction, ensuring that no data is lost [2].

References

- [1] "Chapter 8. Data types," PostgreSQL Documentation, Feb. 20, 2025. GeeksforGeeks, "Transaction isolation Levels in DBMS," *GeeksforGeeks*, Apr. 07, 2025. https://www.geeksforgeeks.org/transaction-isolation-levels-dbms/

- [2] GeeksforGeeks, "ACID properties in DBMS," *GeeksforGeeks*, Apr. 15, 2025. https://www.geeksforgeeks.org/acid-properties-in-dbms/