Transaction Management

* A transaction in a DBMS is a unit of work that comprises a series of operations (such as inserts, updates, deletes) on one or more database objects.
* A transaction is made up of one or more database modifications. Creating, updating, or deleting a record from a table, for example.
* To preserve data integrity and address database issues, it’s critical to keep track of these transactions. We can bundle SQL queries together and run them as a single transaction

Objectives of Transaction management

* To maintain data integrity and consistency even in the presence of concurrent access and potential failures.
* To provide atomicity, consistency, isolation, and durability (ACID properties) for transactions.

ACID Properties

* Used for maintaining the integrity of database transaction processing.

1. Atomicity

* A transaction is a single unit of operation. You either execute it entirely or do not execute it at all. There cannot be partial execution.

1. Consistency

* After any transaction is carried out in a database it should remain consistent. No transaction should affect the data residing in the database adversely.

1. Isolation

* When several transactions need to be conducted in a database at the same time, each transaction is treated as if it were a single transaction. As a result, the completion of a single transaction should have no bearing on the completion of additional transactions.
* Isolation ensures that the intermediate state of a transaction is not visible to other transactions until it is committed.

1. Durability

* Durability ensures that once a transaction is committed, the changes made to the database persist even in the event of system failures.
* In case of system failure or unexpected shutdown and changes made by a complete transaction are not written to the disk, during restart the changes should be remembered and restored.

Transaction states

1. Active state

* this is the state in which a transaction execution process begins. Operations such as read or write are performed on the database.

1. Partially committed

* The transaction has executed all of its operations, and it is about to be committed.

1. Committed stage

* After a transaction execution is completed successfully the transaction is in a committed state. All changes made to the database are permanently documented.

1. Failed state

* If a transaction is aborted while in the active state, or if one of the checks fails, the transaction is in the failed state.

1. Terminated state

* refers to the transaction leaving the system
* This state happens once the transaction leaving the system cannot be restarted once again.

A diagram of a process

Description automatically generated

Explanation

1. Once a transaction states execution, it becomes active. It can issue READ or WRITE operation.
2. Once the READ and WRITE operations complete, the transactions becomes partially committed state.
3. Next, some recovery protocols need to ensure that a system failure will not result in an inability to record changes in the transaction permanently. If this check is a success, the transaction commits and enters into the committed state.
4. If the check is a fail, the transaction goes to the Failed state.
5. If the transaction is aborted while it’s in the active state, it goes to the failed state. The transaction should be rolled back to undo the effect of its write operations on the database.
6. The terminated state refers to the transaction leaving the system.
7. Ice Breaker - Joaquin
8. Discussion:
9. History - Joaquin
10. Concepts/Overview - Joaquin
11. Transaction management in dbms – Joaquin
12. Properties of transaction in DBMS - Jai
13. Schedules in transaction management in DBMS - Jai
14. Benefits - Jai
15. Transaction states in dbms - Christy
16. Operations involved in transaction management in dbms - Christy
17. Rollback and recovery in transaction - Christy
18. Advantages and disadvantages - sy
19. Technology sample and application - sy
20. Application in diagram and programs - sy
21. Q&A
22. Lab Activity
23. Quiz