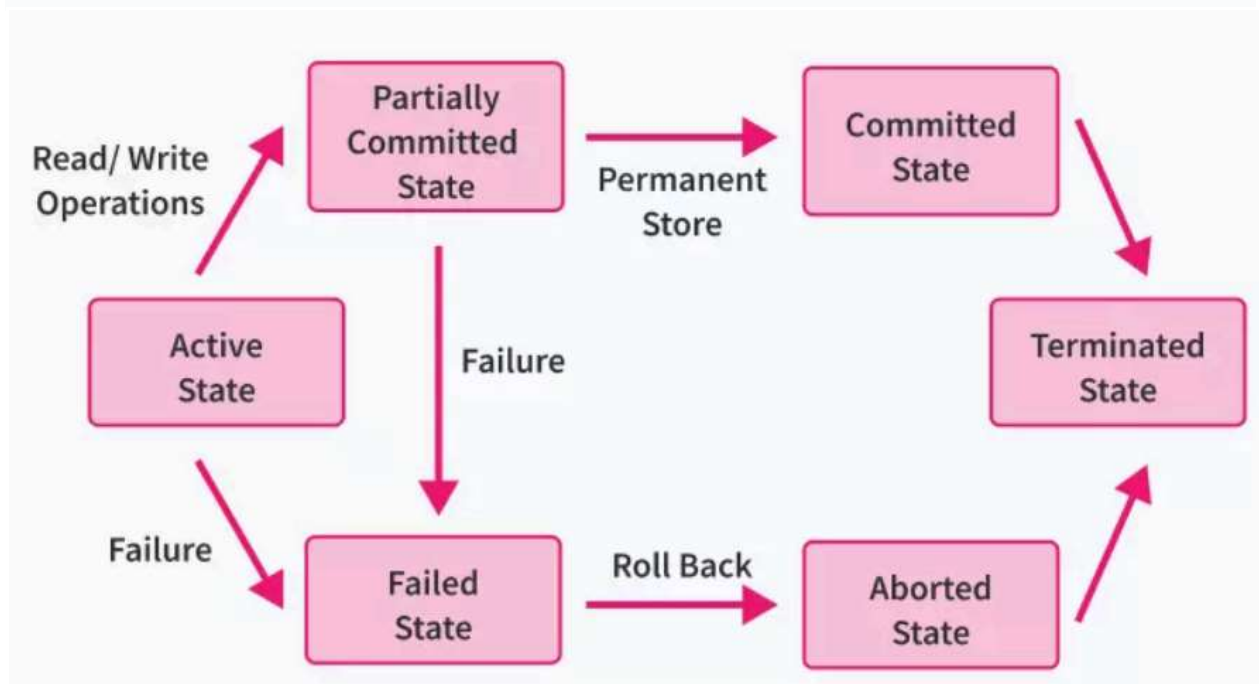


TRANSACTION CONCEPTS

A transaction is a single, logical unit of work that consists of one or more related tasks. A transaction is treated as a single, indivisible operation, which means that either all the tasks within the transaction are executed successfully, or none are.

Transaction States in DBMS

During the lifetime of a transaction, there are a lot of states to go through. These states update the operating system about the current state of the transaction and also tell the user about how to plan further processing of the transaction. These states decide the regulations which decide the fate of a transaction whether it will commit or abort.



The ROLLBACK statement undo the changes made by the current transaction. A transaction cannot undo changes after COMMIT execution.

Following are the different types of transaction States :

Active State: When the operations of a transaction are running then the transaction is said to be in active state. If all the read and write operations are

performed without any error then it progresses to the partially committed state, if somehow any operation fails, then it goes to a state known as failed state.

Partially Committed: After all the read and write operations are completed, the changes which were previously made in the main memory are now made permanent in the database, after which the state will progress to committed state but in case of a failure it will go to the failed state.

Failed State: If any operation during the transaction fails due to some software or hardware issues, then it goes to the failed state. The occurrence of a failure during a transaction makes a permanent change to data in the database. The changes made into the local memory data are rolled back to the previous consistent state.

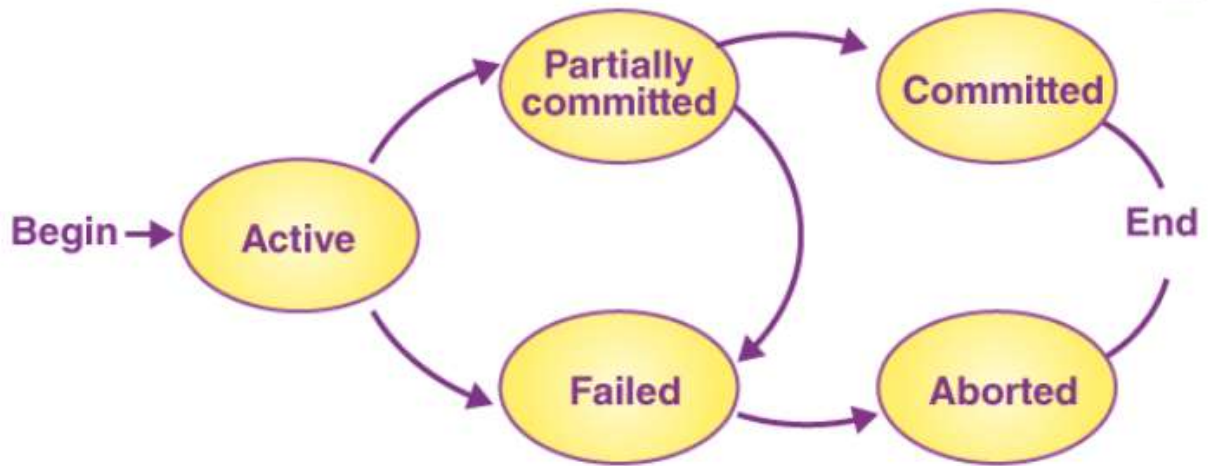
Aborted State: If the transaction fails during its execution, it goes from failed state to aborted state and because in the previous states all the changes were only made in the main memory, these uncommitted changes are either deleted or rolled back. The transaction at this point can restart and start afresh from the active state.

Committed State: If the transaction completes all sets of operations successfully, all the changes made during the partially committed state are permanently stored and the transaction is stated to be completed, thus the transaction can progress to finally get terminated in the terminated state.

Terminated State: If the transaction gets aborted after roll-back or the transaction comes from the committed state, then the database comes to a consistent state and is ready for further new transactions since the previous transaction is now terminated.

States of Transactions

In a database, a transaction can be in one of these states given below –



Active – This is the state in which a transaction is being executed. Thus, it is like the initial state of any given transaction.

Partially Committed – A transaction is in its partially committed state whenever it executes the final operation.

Failed – In case any check made by a database recovery system fails, then that transaction is in a failed state. Remember that a failed transaction can not proceed further.

Aborted – In case any check fails, leading the transaction to a failed state, the recovery manager then rolls all its write operations back on the database so that it can bring the DB (database) back to the original state (the state where it actually was prior to the transaction execution). The transactions in this state are known to be aborted. A DB recovery module can actually select one of these two operations after the abortion of a transaction –

Re-start

Kill the transaction

Committed – We can say that a transaction is committed in case it actually executes all of its operations successfully. In such a case, all of its effects are now established permanently on the DB system.