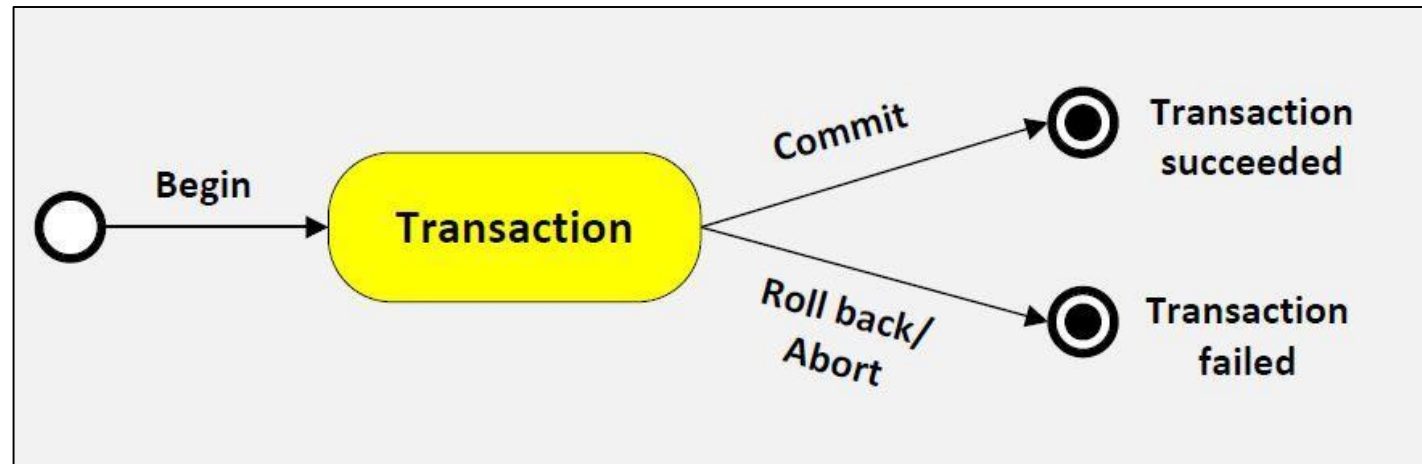


# Database Management System (BCSC – 1003)

## Topic: Transaction Processing



**Dr. Nikhil Govil**

Assistant Professor, Dept. of CEA, GLA University, Mathura.

# Topics to be covered



- Transaction
- Operations of Transaction
- Transaction Property
- State of Transaction

# Transaction

- The transaction is a set of logically related operation. It contains a group of tasks.
- A transaction is an action or series of actions. It is performed usually by a single user to perform operations for accessing the contents of the database.

Example: Suppose an employee of bank transfers Rs 800 from X's account to Y's account. This small transaction contains several low-level tasks:

X's A/c

```
Open_Account(X)
Old_Balance = X.balance
New_Balance = Old_Balance - 800
X.balance = New_Balance
Close_Account(X)
```

Y's A/c

```
Open_Account(Y)
Old_Balance = Y.balance
New_Balance = Old_Balance + 800
Y.balance = New_Balance
Close_Account(Y)
```

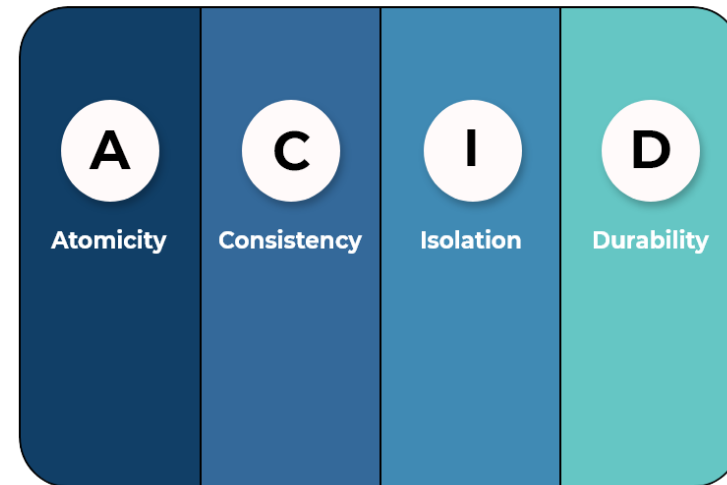
# Operations of Transaction

Following are the main operations of transaction:

- Read (X): Read operation is used to read the value of X from the database and stores it in a buffer in main memory.
- Write (X): Write operation is used to write the value back to the database from the buffer.
- Commit: It is used to save the work done permanently.
- Rollback: It is used to undo the work done.

# Transaction Property

- The transaction has the four properties.
- These are used to maintain consistency in a database, before and after the transaction.
- Property of Transaction are:
  - ❖ Atomicity
  - ❖ Consistency
  - ❖ Isolation
  - ❖ Durability



# Atomicity

- It states that all operations of the transaction take place at once if not, the transaction is aborted.
- There is no midway, i.e., the transaction cannot occur partially.
- Each transaction is treated as one unit and either run to completion or is not executed at all.
- Atomicity involves the following two operations:
  - ❖ Abort: If a transaction aborts then all the changes made are not visible.
  - ❖ Commit: If a transaction commits then all the changes made are visible.

# Consistency

- The integrity constraints are maintained so that the database is consistent before and after the transaction.
- The execution of a transaction will leave a database in either its prior stable state or a new stable state.
- The transaction is used to transform the database from one consistent state to another consistent state.

# Isolation

- It shows that the data which is used at the time of execution of a transaction cannot be used by the second transaction until the first one is completed.
- In isolation, if the transaction T1 is being executed and using the data item X, then that data item can't be accessed by any other transaction T2 until the transaction T1 ends.
- The concurrency control subsystem of the DBMS enforced the isolation property.



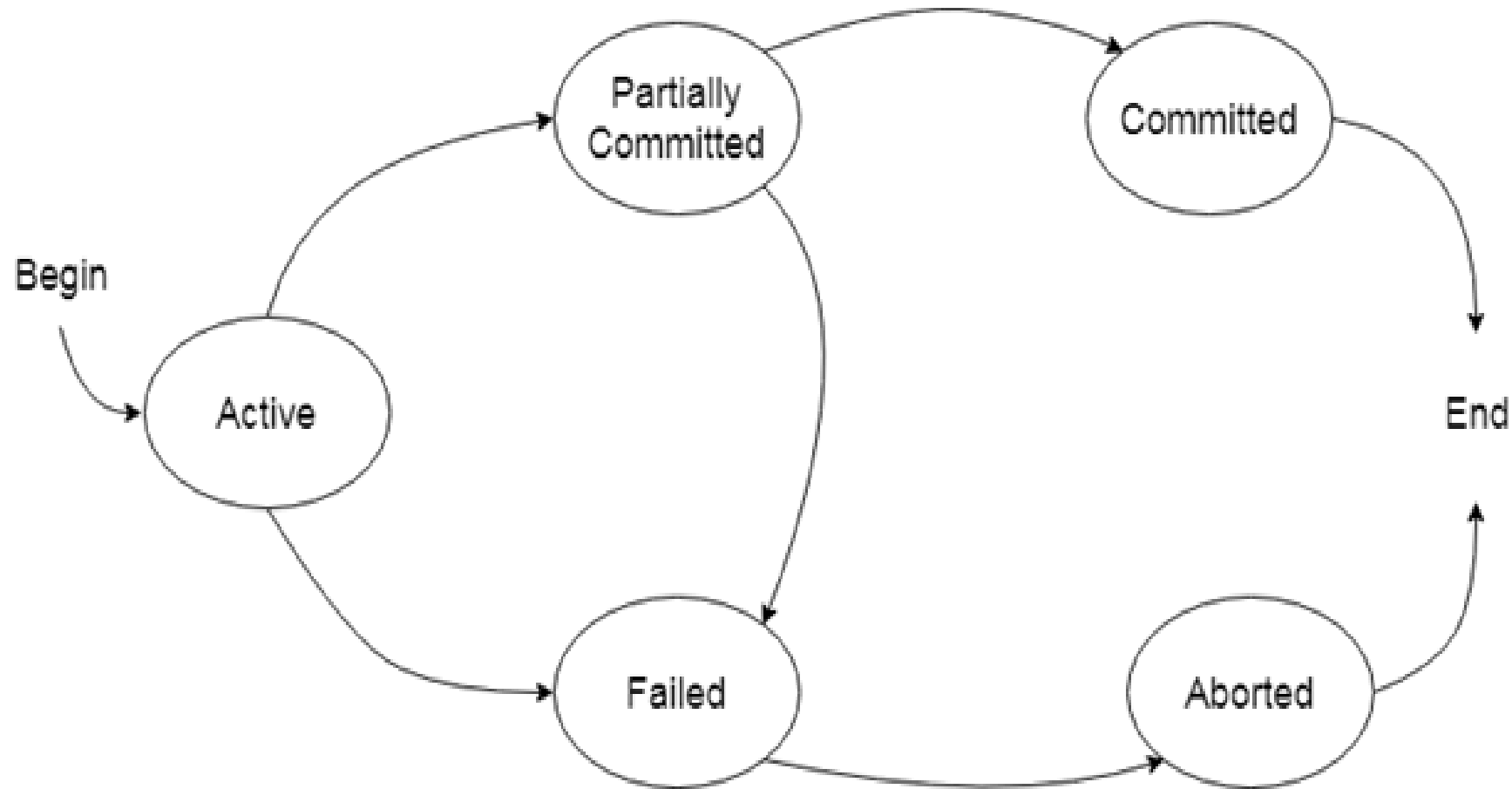
# Durability

- The durability property is used to indicate the performance of the database's consistent state. It states that the transaction made the permanent changes.
- They cannot be lost by the system failure. When a transaction is completed, then the database reaches a state known as the consistent state. That consistent state cannot be lost, even in the event of a system's failure.
- The recovery subsystem of the DBMS has the responsibility of Durability property.

# State of Transaction



In a database, the transaction can be in one of the following states:



# State of Transaction

## Active state:

- The active state is the first state of every transaction. In this state, the transaction is being executed.
- For example: Insertion or deletion or updating a record is done here. But all the records are still not saved to the database.

## Partially committed:

- In the partially committed state, a transaction executes its final operation, but the data is still not saved to the database.
- In the total mark calculation example, a final display of the total marks step is executed in this state.

# State of Transaction

## Committed:

- A transaction is said to be in a committed state if it executes all its operations successfully.
- In this state, all the effects are now permanently saved on the database system.

## Failed state:

- If any of the checks made by the database recovery system fails, then the transaction is said to be in the failed state.
- In the example of total mark calculation, if the database is not able to fire a query to fetch the marks, then the transaction will fail to execute.

# State of Transaction

## Aborted:

- If any of the checks fail and the transaction has reached a failed state then the database recovery system will make sure that the database is in its previous consistent state. If not then it will abort or roll back the transaction to bring the database into a consistent state.
- If the transaction fails in the middle of the transaction then before executing the transaction, all the executed transactions are rolled back to its consistent state.
- After aborting the transaction, the database recovery module will select one of the two operations:
  - ❖ Re-start the transaction
  - ❖ Kill the transaction

# References



- Korth, Silbertz and Sudarshan (1998), “Database Concepts”, 4th Edition, TMH.
- Elmasri and Navathe (2010), “Fundamentals of Database Systems”, 5th Edition, Addison Wesley.
- Date C J,” An Introduction to Database Systems”, 8th Edition, Addison Wesley.
- M. Tamer Oezsu, Patrick Valduriez (2011). “Principles of Distributed Database Systems”, 2nd Edition, Prentice Hall.
- <https://www.javatpoint.com/dbms-file-organization/> last accessed on 08 September' 2021.

*Thank  
you*

