Transaction Concepts – ACID Properties – Schedules – Serializability – Concurrency Control – Need for Concurrency – Locking Protocols – Two Phase Locking – Deadlock – Transaction Recovery - Save Points – Isolation Levels – SQL Facilities for Concurrency and Recovery

TRANSACTION RECOVERY

- **Recovery Algorithms**
  - Recovery algorithms are techniques to ensure database consistency and transaction atomicity and durability despite failures
  - Recovery algorithms have two parts
    1. Actions taken during normal transaction processing to ensure enough information exists to recover from failures
    2. Actions taken after a failure to recover the database contents to a state that ensures atomicity, consistency and durability
- Example
  - Begin transaction
  - Update Acc 1001{balance:=Balance-100}; If any error occurred then
  - Goto Undo;
  - End if;
  - Update Acc 1002{balance:=balance+100}; If any error occurred then
  - Goto undo;
  - End if;
  - Commit;
  - Goto finish;
  - Undo: rollback;
  - Finish: return;

Conn..

- **Requirement for recovery**
  - Implicit rollback
  - Message handling
  - Recovery log
  - Statement atomicity
  - No nested transaction
- **Transaction recovery**
- Database updates are kept in buffer in main memory and not physically written to disk until commit**.**
- **System recovery**
- Local failures –affect only the transaction which the failure has actually occurred. Global failures- affect all the transaction in progress at the time of failure.

Conn..

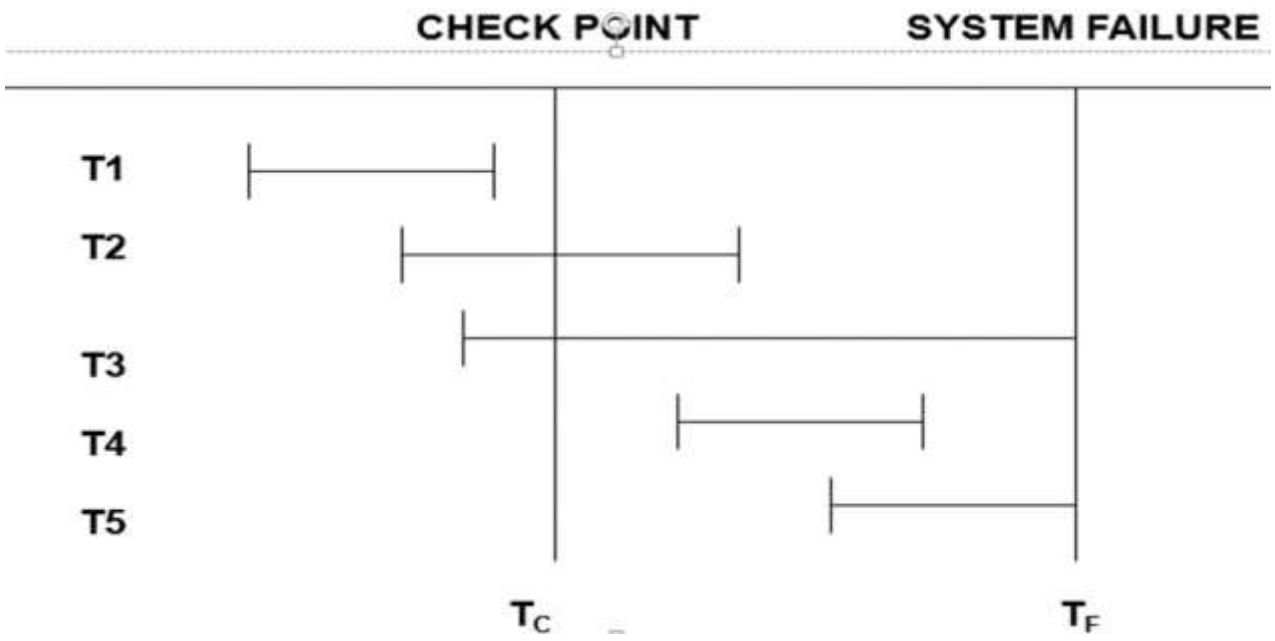
- System failure – do not physically damage the DB Eg: power shut down

Media failure-cause damage to the DB. Eg: head crash **ARIES Recovery**

Algorithm

- ARIES-Algorithm for Recovery and Isolation Exploiting Semantics
- ARIES recovery involves three passes
- Analysis pass: Determines the REDO and UNDO lists.
- Redo pass: Repeats history, redoing all actions from REDO List Undo pass: Rolls back all incomplete transactions

Conn..



Conn..

- The system failure occurred at time T_f , the most recent check point prior to the time T_f was taken at a time T_f
- Start with two list of transaction the UNDO and REDO list
- search forward through the log starting from check point.
- if begin transaction log record is found for transaction(T) add T to UNDO list.
- if commit log record is found for transaction(T),add T to REDO list
- when the end of log record is reached the UNDO and REDO list is identified

UNDO	REDO
T_3	T_2
T_5	T_4

Conn…

Thank You…….