# Database Management System – 43 (Database Recovery Techniques)

Ajay James
Asst. Prof in CSE
Government Engineering College Thrissur



## Outline

- Recovery concepts
- Checkpoints
- Fuzzy checkpointing
- Tansaction rollback
- NO-UNDO/REDO Recovery Based on Deferred Update

# Recovery Concepts

- Recovery process restores database to most recent consistent state before time of failure
- Information kept in system log
- Typical recovery strategies
  - Restore backed-up copy of database
    - Best in cases of extensive damage
  - Identify any changes that may cause inconsistency
    - Best in cases of non-catastrophic failure
    - Some operations may require redo



# Recovery Concepts

- Deferred update techniques
  - Do not physically update the database until after transaction commits
  - Undo is not needed; redo may be needed
- Immediate update techniques
  - Database may be updated by some operations of a transaction before it reaches commit point
  - Operations also recorded in log
  - Recovery still possible

## Checkpoints in the System Log and Fuzzy Checkpointing

- A [checkpoint, list of active transactions] record is written into the log periodically
- All transactions that have their [commit, T ] entries in the log before a [checkpoint] entry do not need to have their WRITE operations redone in case of a system crash
- List of transaction ids for active transactions at the time of the checkpoint is included in the checkpoint record



## Checkpoints in the System Log and Fuzzy Checkpointing

- Taking a checkpoint
  1. Suspend execution of all transactions temporarily
  2. Force-write all main memory buffers that have been modified to disk
  3. Write a checkpoint record to the log, and force write the log to the disk
  4. Resume executing transactions
- DBMS recovery manager decides on checkpoint interval

## Checkpoints in the System Log and Fuzzy Checkpointing

- Fuzzy checkpointing
  - System can resume transaction processing after a begin_checkpoint record is written to the log
  - Previous checkpoint record maintained until end_checkpoint record is written



## Transaction Rollback

- Transaction failure after update but before commit
  - Necessary to roll back the transaction
  - Old data values restored using undo-type log entries
- Cascading rollback
  - If transaction T is rolled back, any transaction S that has read value of item written by T must also be rolled back
  - Almost all recovery mechanisms designed to avoid this

## NO-UNDO/REDO Recovery Based on Deferred Update

- Deferred update
  - Postpone updates to the database on disk until the transaction completes successfully and reaches its commit point
  - Redo-type log entries are needed
  - Undo-type log entries not necessary
  - Can only be used for short transactions and transactions that change few items
    - Buffer space an issue with longer transactions



## NO-UNDO/REDO Recovery Based on Deferred Update

- Deferred update protocol
  - Transaction cannot change the database on disk until it reaches its commit point
  - All buffers changed by the transaction must be pinned until the transaction commits (no-steal policy)
- Transaction does not reach its commit point until all its REDO-type log entries are recorded in log and log buffer is force-written to disk

# Reference


- Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education 6th edition and 7th edition




# Thank you