SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-IOT Including CS&BCT

COURSE NAME : 19SB504 DATABASE MANAGEMENT SYSTEMS

III YEAR / V SEMESTER

Unit V- CONCURRENCY CONTROL AND RECOVERY SYSTEM

Topic : **ADVANCED RECOVERY TECHNIQUES**

Objective:
Advanced recovery techniques in Database Management Systems (DBMS) are designed to ensure the consistency and integrity of the database in the event of failures or errors.

Traditional recovery techniques like undo logging and redo logging are essential, but advanced techniques provide additional features and efficiency improvements.

There are nine advanced recovery techniques:

1.**Checkpoints:**

Fuzzy Checkpointing:

Instead of taking a consistent snapshot of the entire database, fuzzy checkpoints allow transactions to continue while the checkpoint is being taken.

This helps reduce the downtime associated with traditional checkpoints.

2. Write-Ahead Logging (WAL):

Group Commit:

Instead of committing each transaction individually, group commit allows multiple transactions to be committed together.

This reduces the number of disk writes, improving performance.

3. Shadow Paging:

Copy-on-write:

In shadow paging, modifications are made to a separate copy of the page rather than the actual page.

Only the modified pages are copied to the main storage when a transaction commits.

This helps in reducing the overhead of copying entire pages.

4. Log Sequence Number (LSN) Based Recovery:

LSN Ordering:

Transactions are assigned a unique LSN, and the log records are ordered based on these numbers.

During recovery, logs can be replayed in the correct order, ensuring that the database reaches a consistent state.

5. Write-Behind Logging:

Asynchronous Commit:

Instead of waiting for the log to be written to disk before a transaction is considered committed, asynchronous commit allows the transaction to be considered committed once the log record is in memory.

This can improve the response time of transactions.

6.Quiescent Checkpoints:

Database Snapshot:

A quiescent checkpoint involves temporarily halting database activity to take a consistent snapshot of the database.

This snapshot is used during recovery to bring the database to a consistent state.

7. Immediate Consistency Techniques:

Fast-Path Recovery:

Rather than rolling back transactions during recovery, fast-path recovery discards incomplete transactions and applies committed transactions.

This reduces the time needed for recovery.

8. Point-in-Time Recovery:

Incremental Recovery:

Instead of recovering the entire database, incremental recovery allows the recovery of only the changes made since the last backup.

This reduces the recovery time for large databases.

9. Distributed Database Recovery:

Two-Phase Commit (2PC):

Ensures that all nodes in a distributed database either commit or abort a transaction together, maintaining consistency across the distributed system.

Thank you ......