

Tutorial 11: Database Recovery Techniques

CS3402 Database Systems

Question 1

- Given the read and write operations for 4 transactions and the system log before a system crash, describe the recovery based on the **deferred update** recovery strategy.

T_1
read_item(A)
read_item(D)
write_item(D)

T_2
read_item(B)
write_item(B)
read_item(D)
write_item(D)

T_3
read_item(A)
write_item(A)
read_item(C)
write_item(C)

T_4
read_item(B)
write_item(B)
read_item(A)
write_item(A)

[start_transaction, T_1]
[write_item, T_1 , D, 20]
[commit, T_1]
[checkpoint]
[start_transaction, T_4]
[write_item, T_4 , B, 15]
[write_item, T_4 , A, 20]
[commit, T_4]
[start_transaction, T_2]
[write_item, T_2 , B, 12]
[start_transaction, T_3]
[write_item, T_3 , A, 30]
[write_item, T_2 , D, 25]

Question 1 (Answer)

- **Redo-logging (deferred update):** Log → COMMIT → change
- There is no need to redo the write_item operations of T_1 because it is committed before the last checkpoint.
- T_2 and T_3 are ignored because they did not reach their commit points.
- T_4 is redone because its commit point is after the last system checkpoint.

[start_transaction, T_1]
[write_item, T_1 , D, 20]
[commit, T_1]
[checkpoint]
[start_transaction, T_4]
[write_item, T_4 , B, 15]
[write_item, T_4 , A, 20]
[commit, T_4]
[start_transaction, T_2]
[write_item, T_2 , B, 12]
[start_transaction, T_3]
[write_item, T_3 , A, 30]
[write_item, T_2 , D, 25]

Question 2

- Given the read and write operations for 4 transactions and the system log before a system crash, describe the recovery based on the **immediate update** recovery strategy.

T_1
read_item(A)
read_item(D)
write_item(D)

T_2
read_item(B)
write_item(B)
read_item(D)
write_item(D)

T_3
read_item(A)
write_item(A)
read_item(C)
write_item(C)

T_4
read_item(B)
write_item(B)
read_item(A)
write_item(A)

[start_transaction, T_1]
[write_item, T_1 , D , 20]
[commit, T_1]
[checkpoint]
[start_transaction, T_4]
[write_item, T_4 , B , 15]
[write_item, T_4 , A , 20]
[commit, T_4]
[start_transaction, T_2]
[write_item, T_2 , B , 12]
[start_transaction, T_3]
[write_item, T_3 , A , 30]
[write_item, T_2 , D , 25]

Question 2 (Answer)

- **Undo-logging (immediate update):** Log → change → COMMIT
- There is no need to undo the write_item operations of T_1 because it is committed before the last checkpoint.
- There is no need to undo any write_item for T_4 because it has a commit record.
- T_2 and T_3 are undone because they did not reach their commit points.

[start_transaction, T_1]
[write_item, T_1 , D , 20]
[commit, T_1]
[checkpoint]
[start_transaction, T_4]
[write_item, T_4 , B , 15]
[write_item, T_4 , A , 20]
[commit, T_4]
[start_transaction, T_2]
[write_item, T_2 , B , 12]
[start_transaction, T_3]
[write_item, T_3 , A , 30]
[write_item, T_2 , D , 25]