Durability

A successful transaction contains this operation.

update customer set balance = balance + 100;

The above operation causes the following actions to take place.

a) Write data to database file
b) Read data from database file into database buffer
c) Write data to log buffer
d) Write data to the rollback segment
e) Write data to log file
f) Update data in data buffer

1. In which sequence will the above actions be executed?

B -> C -> D -> F -> A ->E

2. Why is data written to the log buffer and then to the log file asynchronously

The reasons are to improve performance for the system and availability for the users.

+ Writing to disk is relatively slower than writing to memory, by buffering the log in the buffer, the database can reduce the number of disk writes and improve the performance of the system.

+ Asynchronous writing also allows the database system to group entries together and write them to disk in batches, which will reduce overhead for I/O operations.

3. Assume a successful transaction is committed.

After which point (from the actions above) will the DBMS have to honour the transactioneven if a system or media failure occurs shortly afterwards.

Once the transaction has been written to the log file, it is considered durable and the DBMS has to honor, meaning that changes are now persistent, the transaction even if a system or media failure occurs shortly afterwards.

4. What event ensures that the log buffer is flushed to the log file ?

Checkpoint is used as a background process that periodically flushes the contents of the log buffer to the log file.

5. Transaction A has performed the following operations:

- Insert
- Insert
- Insert
- Update

Assume that each of the above operations has physically written to the log file. However, the transaction has not yet been completed.

Which one of the following options best describes what has happened to the database file?

a) Some of the operations above, will have caused changes to the database file

b) None of the operations above, will have caused changes to the database file
This is the correct answer as the transaction is first written to the database buffer in memory and not yet to be written to the database file

c) All of the operations above, will have caused changes to the database file

d) Any of the above is possible

6. What is a checkpoint?

A Checkpoint is a background process that writes all the modified data from the database buffer to the disk, or other persistent storage.

7. True or false?

- **All transactions must be completed before a checkpoint can take place**
  This is **FALSE** as checkpoint will write all the modified from the database buffer to persistent storage, meaning that any transactions that have been committed up to the point when checkpoint occurs will be included in the check point, even if it is not completed.
- **A transaction can be partially completed when a checkpoint take place**
  According to the explanation above, this sentence will be **TRUE**.

8. Consider the transaction log entries shown below. All transactions consist of inserts into tables named DEPOSIT, WITHDRAWAL and INTEREST rows into a database. Some transactions involve only one database operation. Some transactions others involve several operations.

   For each log entry, the transaction ID, the time, the action performed, the table and row ID affected and the row before/after images are recorded (details not shown). Log entries for checkpoints, transaction starts, commits and rollbacks are also recorded as shown. All transactions occurred in the given order.

Tran ID	Action	Table Affected	Rowid	Before	After
1	START				
2	START				
1	INSERT	DEPOSIT	300		---/---
1	INSERT	WITHDRAWL	200		---/---
2	INSERT	INTEREST	100		---/---
2	INSERT	INTEREST	101		---/---
1	COMMIT				
3	START				
3	INSERT	DEPOSIT	301		---/---
4	START				
4	INSERT	DEPOSIT	600		
	CHECKPOINT				
3	INSERT	WITHDRAWL	201		---/---
5	START				
2	INSERT	INTEREST	102		---/---
5	INSERT	DEPOSIT	302		---/---
3	ROLLBACK				
5	COMMIT				
2	INSERT	INTEREST	103		---/---
6	START				
2	COMMIT	INTEREST	104		---/---
6	INSERT	WITHDRAWL	202		---/---

end of log

9. What is a system failure?

Assume that as system failure occurred after the final entry in the above log file was made.When a system failure occurs, the DBMS examines the log file.

A System failure occurs when an unexpected event caused a computer system or application to stop functioning properly or to crash it entirely.

10. Consider transaction 1.

- **Was it completed before the checkpoint?**
  Yes, It is completed before the checkpoint.
- **Were all of the logged actions of transaction 1 written to the database file?**
  Yes, as it is completed before the checkpoint.
- **What should happen to this transaction Undo / Redo / No Action?**
  As it is completed and written to the database, no action would happened

11. **Consider transaction 2.**

- **Was it completed before the checkpoint?**

  No, it is committed after the checkpoint.
- **Were any of the actions of transaction 2 logged before the checkpoint written to thedatabase file?**

  Yes, those are the two INSERT statements.
- **Were any of the actions of transaction 2 logged after the checkpoint written to thedatabase file?**

  Yes, those are the two INSERT statements.
- **What should happen to this transaction Undo / Redo / No Action?**

  Redo should happen.

12. Consider transaction 3.

- **Was it completed before the checkpoint?**
  No, it is rolled back after the checkpoint.
- **Were any of the actions of transaction 3 logged before the checkpoint written to thedatabase file?**
  Yes, that is an INSERT statement.
- **Were any of the actions of transaction 3 logged after the checkpoint written to thedatabase file prior to the system failure occurring.**
  No, it has been rolled back.
- **What should happen to this transaction Undo / Redo / No Action?**
  Redo should happen.

13. Consider transaction 4.

- **Was it completed before the checkpoint?**
  No, it is yet to be committed.
- **Were any of the actions of transaction 4 logged before the checkpoint written to thedatabase file?**
  No, it is yet to be committed.
- **What should happen to this transaction Undo / Redo / No Action?**
  As it is not committed, it should be Undo.

14. Consider transaction 5.

- **Was it started after the checkpoint?**
  Yes, it was.
- **The log shows that this transaction was rolled-back**
  - **At the time of the systems failure, could any of the actions performed by thistransaction been written to the database file?**
    No, as it started before the checkpoint.
  - **Were any of the actions of this transaction written to the database file?**
    As the transaction started after the checkpoint and it is subsequently rolled-back, it will only be written to the database buffer not the database file.
- **What should happen to this transaction Undo / Redo / No Action?**
  As it is not completed before the checkpoint, it should be Undo.

15. Consider transaction 6.

- **What should happen to this transaction Undo / Redo / No Action?**
  As it is not completed before the checkpoint, it should be

16. How does a media failure differ from a systems failure

Media failure is more specific toward a hardware storage device failure, such as hard disk, or persistent storage as it becomes corrupted or is damaged. System failure, on the other hand, refers to a general, broader term of failure, such as software error, network failure, or power outage.

17. Assume that a media failure has occurred.

It is necessary to revert to the **backup** that was done just prior to the start of this log.
If a media failure occurred and the data on the affected storage device is no longer recoverable, it may be necessary to revert to the backup that was done just prior to the start of this log.

18. How often should a checkpoint occur?

It may depend on the specific system and its configuration as well as the policy and nature of the business operating that system. In general checkpoints should occur frequently enough to ensure that the amount of data needs to be recovered in case of a failure without affecting the performance of the system too heavily.

19. What transactions need to be

- **Redone**

  Transactions that were committed before the point of failure or if its changes were not yet to be written to the database file need to be redone.
- **Undone**

  Transactions that were not committed, meaning that it is still in its process, and its process spans after the checkpoint should be undone, otherwise it will leave the database in an inconsistent state.

20. List 3 distinct events that can cause the DBMS to fail. Specify what action could be taken bythe recovery manager to repair the database.

+ Media failure: such as a disk crash or power outage can cause the DBMS to fail. In this case, the recovery process can take place by restoring from the recent backup and redoing the transaction log.

+ System failure: such as a software error, or network error can cause the DBMS to fail. In this case, the recovery process can take place by rolling back any incomplete transactions and redoing the transaction log to recover any changes made before the failure.

+ User error: such as accidentally deleting or modifying of data can cause the DBMS to fail. In this case, the recovery process can take place by restoring the transaction log since the backup was taken.

21. Describe the state transactions during a backup.

During a backup, transaction is progress will be temporarily suspended, while transactions that have already been committed will be included in the backup.
Transaction that are still in progress at the time of the back up will not be included in the backup, as their changes have not been committed to the database.

Lab Tasks

Continue with Assignment work