Crash recovery

Transaction is a collection of actions that transform a database from a consistent state to another consistent state; during the execution the database might be inconsistent.

Properties of a transaction

The ACID properties of a transaction are:

Atomicity: a transaction is treated as a single/atomic unit of operation and is either executed completely or not at all.

Consistency: a transaction preserves DB consistency. It transforms DB from its consistent state to consistent state.

Isolation: Execution of one transaction is isolated from that of other transactions.

Durability: if a transaction commits, its effect persists. If a transaction has been reported back to the user as complete, the resulting changes to the database survive subsequent hardware and software failures.

A transaction can be interrupted before running to completion for variety of reasons e.g. system crashes, disk failure, logical errors, system errors, etc. A DBMS must ensure that the changes made by such incomplete transactions are removed from the DB. For example, if the DBMS is in the middle of transferring money from account **A** to account **B**, and has debited the first account but not yet credited the second account when the crash occurs, the money debited from account A must be restored when the system comes back up after the crash.

The **recovery manager** of a DBMS is responsible for ensuring **atomicity** by undoing (rolling back) the actions of transactions that do not commit and **durability** by making sure that all actions of committed transactions survive system crashes and media failures i.e. corrupt disk.

The aim of **crash recovery** is to restore the database to the most recent consistent state which existed prior to the failure/crash.

Introduction to ARIES

The ARIES (Algorithm for Recovery and Isolation Exploiting Semantics) is a database recovery algorithm and has three phases:

Analysis phase: It scans the log (history of executions by the DBMS) forward (from the most recent checkpoint) to identify all active transactions and dirty pages in the buffer pool at the time of the crash.

Redo phase: This phase returns the database to the state at the time the DB crash occurred. Database history is repeated to reconstruct state at crash.

Undo phase: When the system restarts after the crash, the list of active transactions (uncommitted at the time database crashed) identified in phase 1 is rolled back (reversed) individually and restores the database to the consistent state that existed before the start of transaction.

Other recovery related data structure

In addition to the log, the following two tables contain important recovery related information:

Transaction table: this table contains one entry for each active transaction. The entry contains (among other things) the transaction id, status, etc. The status can be that it is in progress, committed or is aborted.

Dirty page table: contains one entry for each dirty page in the buffer pool, i.e. pages with changes that are not yet reflected on disk.

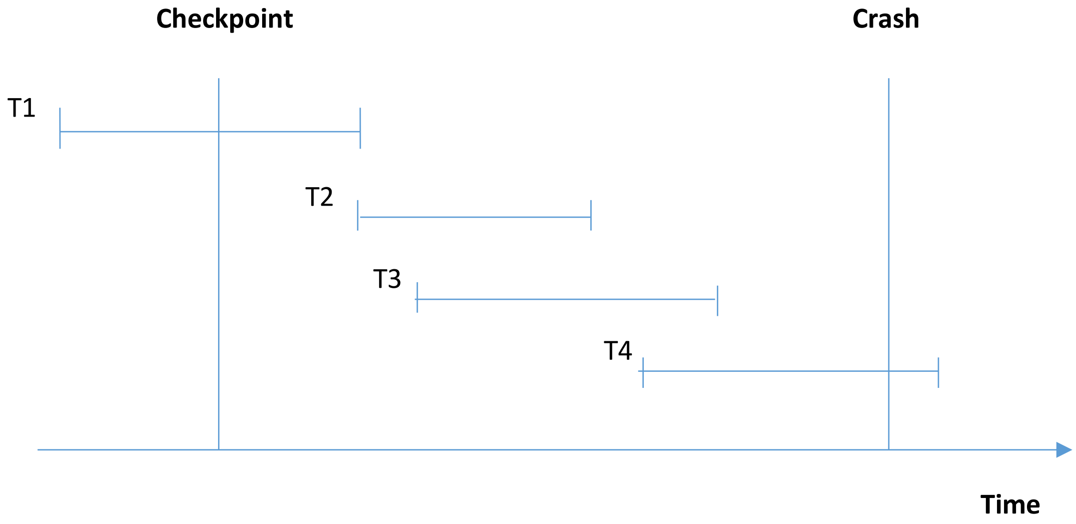
Write Ahead Logging (WAL) protocol

WAL is one of the principles behind ARIES recovery algorithm. In a system using WAL, all modifications to the database object are first recorded in the log, the record in the log must be written to stable storage before the change to the database object is written to disk(before they are applied on the database).

The purpose of this can be illustrated by an example. Imagine a program that is in the middle of performing some operation when the machine it is running on loses power. Upon restart, that program might well need to know whether the operation it was performing, succeeded or failed. If a write-ahead-log was used, the program could check this log and compare what it was supposed to be doing when it unexpectedly lost power to what was actually done. On the basis of this comparison, the program could decide to undo what it had started, complete what it had started, or keep things as they are.

Check pointing

Keeping and maintaining the logs in real time may fill out all the memory space available in the system. As time passes log file may be too big to be handled at all. **Checkpoint** is a mechanism where all the previous logs are removed from the system and stored permanently in storage disk. Checkpoint declares a point before which the DBMS was in a consistent state and all transactions were committed. Periodic check pointing can reduce the time needed to recover from a crash.



Lock: A mechanism used to control access to a database objects.

Shared lock on an object can be held by two transactions at the same time. If a transaction needs to read a database object, a shared lock is required.

Exclusive lock on an object ensures that no other transaction hold any lock on this object. If a transaction needs to write (modify or change) a database object, an exclusive lock is required.

Media recovery: this is the process of restoring the entire disk files/folders after a crash. Disk failure can occur as a result of formation of bad sectors, disk head crash, or any other failures which destroys all or parts of disk storage.