#ASLI ENGINEERING

How does a database guarantees reliability?

SWIPE

BY

ARPIT BHAYANI

Write-ahead logging

For any database system, Reliability is super-important

↓

does everything to guarantee reliable operations

Persistent Database

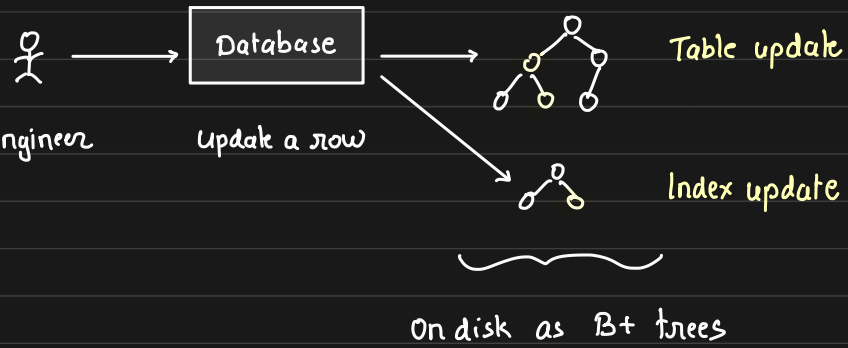
MySQL, PostgreSQL

Upon commit:

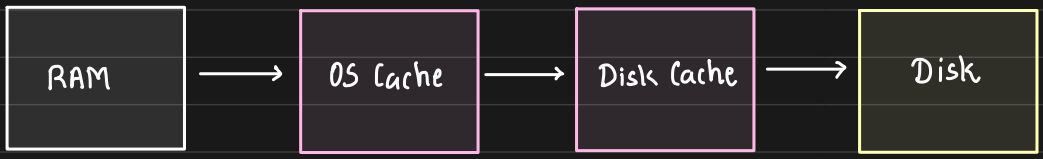
- data / updates are stored on disk (Non-volatile storage)

↓ Power loss ↓ OS failure ↓ Hardware failure

Non-volatile storage safe from



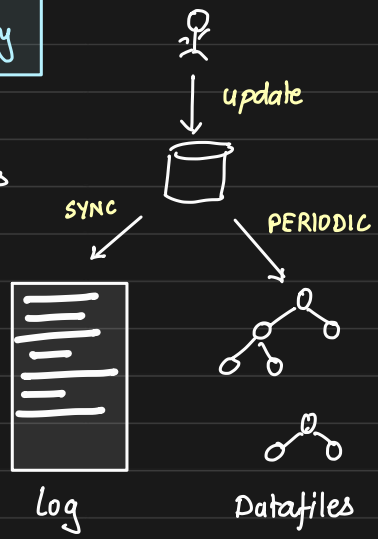
Disk writes are complicated



Write Ahead logging → Standard method for ensuring

Data Integrity

Core idea: Before making changes to actual datafiles (tables and indexes), log those changes describing the changes.



Flow: Update triggered on DB
- log the entry in WAL
- makes changes in table & indexes

Advantages of WAL

- we do not need to flush the data changes on every commit
- in case of crash we can recover by replaying the logs
* - reduce the number of disk writes

WAL file is sequential so, cost of logging the changes is significantly lower than the cost of changing data files

log → 1 disk write
data file → update table
update index
tree rebalance

- Point-in-time recovery is possible with WAL

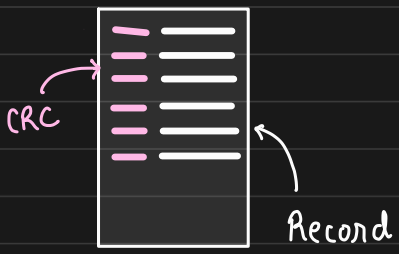
 ← Apply WAL on a snapshot

Data integrity in WAL

Every individual record in WAL is CRC-32 protected

↓

We could tell if record content are correct
CRC is checked during crash recovery,
and replication



WAL is by default enabled by all the databases.

Insert position

↓

Log Sequence Number (LSN)

[Byte offset into logs]

16 MB files

Segment - 000000001