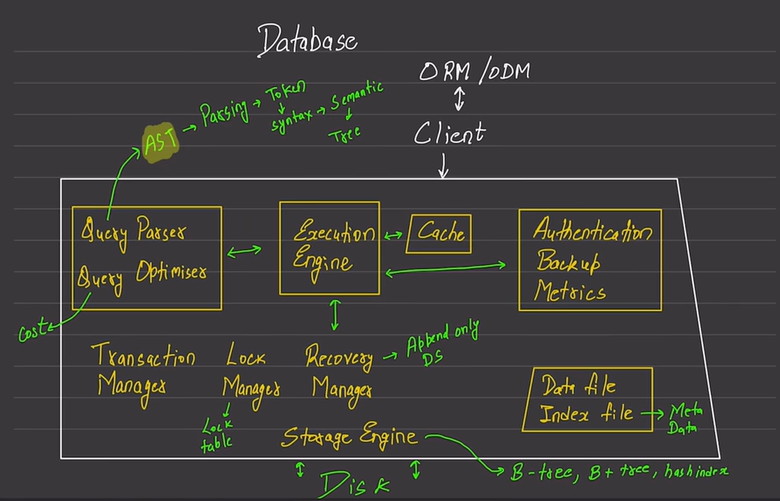
-Data is always store and retrieved from disk

-it is a costly operation

-not single threaded like Javascript, uses maximum concurrency as possible



ORM

-talks to client

-simplifies interaction with DB

-better to talk to client directly, so that at least one layer is skipped

Query Parser

-converts the text query (select \* from..) to AST (tree)

-in between it converts the text to tokens , checks the syntax..

Query optimiser

-after query is being parsed, it is optimised

-optimises the query

-cuts cost of query by checking and reducing the query into a smaller one if possible

Cache

-retrieving data from disk is costly

- Therefore data is cached

-if data is present in cache, data is retrieved from cache rather from the disk, and does not go to disk

Storage engine

-more powerful the storage engine, more powerful the database is

-b-tree , b+tree , hash index etc..

Transaction manager

-ensures the property ‘if it happens, it happens, nothing takes place partially’

Lock manager

-if we wanna write data, lock manager gives access to only that portion of the disk

-ensures no two process writes on the same portion at a given time

-Allows two/more processes to read data from same portion of the disk

Recovery manager

-Append only data structure

-goes to the time where everything was alright

H2 does not support the concept of schemas