



#ASLI ENGINEERING

Picking the right database

SWIPE

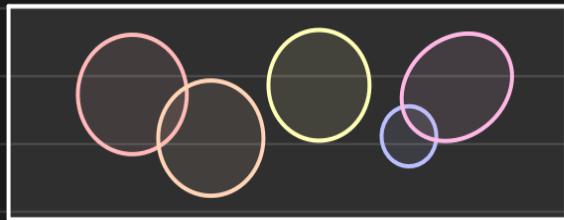
BY

ARPIT BHAYANI

Picking the right database

It is not a fight, so no need to pick a side

a database is designed to solve a particular problem really well.



→ Each kind of database picks a segment with a slight overlap

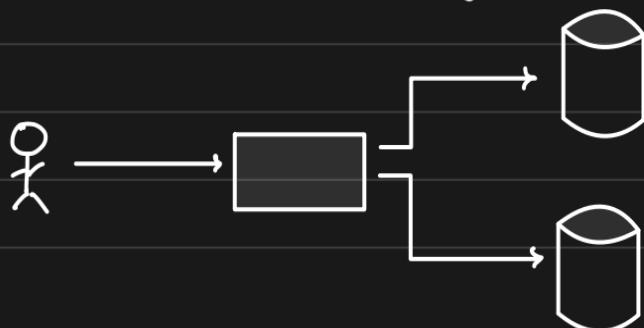
Common Misconception: Picking Non-relational DB because relational Databases do not scale

Why non-relational DBs scale

- There are no relations and constraints
- Data is modelled to be sharded
 - ↳ split across multiple nodes

if we relax the above ↑ on relational DB, we can scale it too!!

- do not use foreign key check
- do not use cross shard transaction
- do manual sharding



Does this mean, no DB is different?

No! every single database has some peculiar properties and guarantees
and if you need those, you pick that DB

How does this help in designing system?

While designing any system,

do not jump to a particular DB right away

1. understand what data you are storing
2. understand how much of data you will be storing
3. understand how you will be accessing the data
4. what kind of queries you will be firing
5. any special feature you expect

e.g. Expiration

How to pick the right DB? [Not exhaustive, but you'll get the idea]

if data can fit on a single node

You need strong consistency? ↳ data correctness is critical

↳ go for relational database

You need complex queries, aggregations

↳ go for relational database

Your access is KV based but need it to be really fast

↳ go for Redis

You need advanced data structures & algorithms

↳ go for Redis

if data cannot fit on one node

You have expertise in SQL & can do manual sharding

↳ drop constraint & go for relational DB

You have simple KV based access

↳ go for KV store like DDB, MongoDB etc

if you require sophisticated graph algorithms

↳ go for graph DB like Neo4j

if you have nothing specific, but want to future proof

↳ go for document DB like MongoDB