Dynamo architecture analysis

NOTE – ① Based on paper published by AWS Team.
② AWS service DynamoDB architecture is somewhat different from who is presented in paper.

PART I – Dynamo Background

Why dynamo? → Amazon's storage needs were fulfilled by RDBMS.

But most of services don't require all capabilities of RDBMS like complex queries.

Simple query on key is requirement.

To run such system, you require expensive hardware & experienced admin

Requirements

① Query model → simple read/write operations uniquely identified by key.

② ACID → weaker consistency with higher availability.

③ Efficiency → work on commodity hardware.

Design considerations

① Data replication

CAP Theorem → Partition Tolerance, strong Consistency and high availability cannot be achieved simultaneously.

Difficult to achieve strong consistency along with maximum availability.

due to server/network faults

↳ Can be ↑ by adding more replicas.

as there are multiple replicas, it could lead to conflicting changes.

① when should conflicts be resolved ?

↳ conflicts are resolved on reads and writes are never rejected.

↑ watch "Logical Clocks" video on channel.

② who will resolve?

application or database

↑ last write wins

↑ can be as per use-case

② Incremental scalability → scale out without any impact

③ Symmetry → leaderless

↑ all nodes perform same function

④ Decentralization

no centralised control.

⑤ Heterogeneity

work distribution should be proportional to capability of node.

m4.4xlarge gets more work than m4.large

Request routing between nodes

↳ request routing is avoided through multiple nodes.

multi hop

↑ Can increase response time

- routing info is maintained locally by nodes for direct interaction with node.

Zero hop Distributed Hash Table (DHT)

We'll be discussing System Architechure in Part II of this series.

Subscribe for more

YouTube - MsDeep Singh

Happy Learning ☺