

No SQL Database

not only sql database, complements RDBMS

1. Natively scalable
2. Dynamic schema

Distributed Systems=to solve storage and computation problem of data, you are using multiple system instead of using single machine.

1. Make use of multiple systems (commodity H/W, inexpensive system)
2. Each machine will be connected to each other over the network
3. All the complexity should be abstracted from the end user

Failure

1. commodity H/W
2. Dependent on N/W

Solution for distributed database = Replication (store data in multiple machine)

CAP THEOREM

C=consistency (if there are multiple people trying to call at the same time for the same data, for the same key and response will be same then system is consistent)

A=availability (if an end user is trying to make call to the system, they should always respond)

P= partition tolerance (store data copies in multiple location over the network)

If we make an distributed system out of C, A and P only two of the following are guaranteed

Types of NoSQL Database-

1. Key Value database-

Use Cases- caching, cluster management in distributed systems, real time analytics

Challenges- lack of query support, data modelling complexity, limited data type

Example- Redis

2. Document database-store data in the forms of document (JSON, BSON)

Use Cases- semi structured data, flexibility, best of both

Challenges- consistency, Aggregation, transaction

Example- MongoDB

3. Graph database- collection of edges and vertices

Use Cases- complex relationship, recommendation system, fraud detection

Challenges- data modelling, query speed

Example- Neo4j

