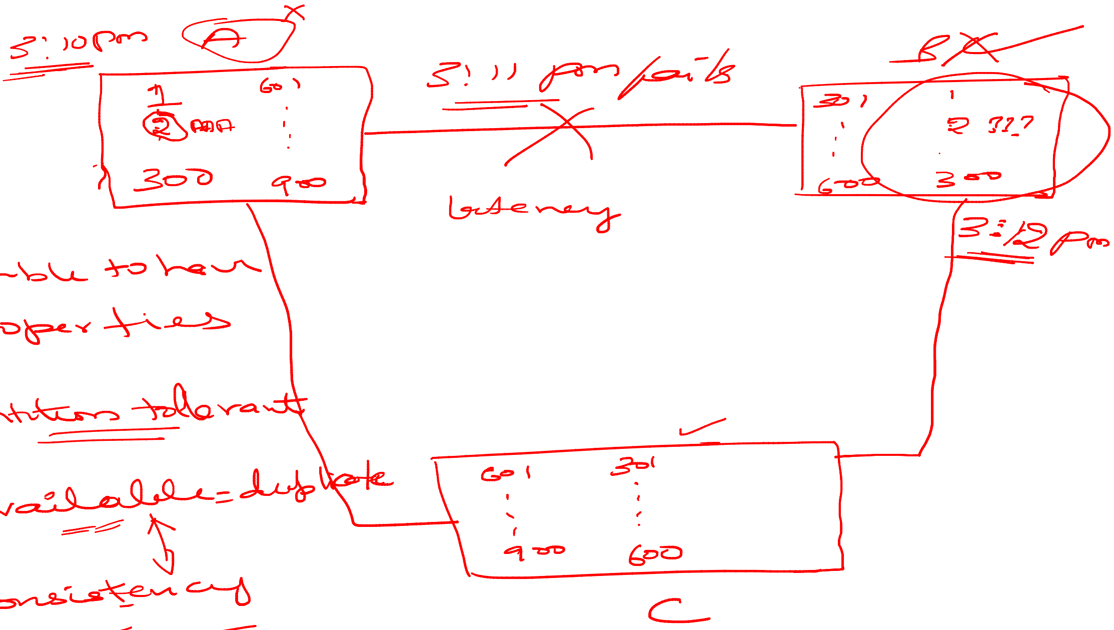
# CAP Theorem

In the context of NOSQL

# Distributed systems and NOSQL

- we are witnessing a strong and increasing desire to scale systems *out*
- This is accomplished through adding additional commodity hardware to a system to handle the increased load.
- As a result of this scaling strategy, an additional penalty of complexity is incurred in the system.
- This is where the CAP theorem comes into play.

Distributed systems => Partition of Data.



Desirable to have 3 Properties

1. Partition tolerant

2. Available = duplicate

3. Consistency

Employee

| id | name |
|---|---|
| 1 | |
| 2 | ~~AAB~~ ~~AAA~~ |
| ⋮ | |
| 900 | |

- We need a distributed database system having such features:
- – **Fault tolerance**
- – **High availability**
- – **Consistency**
- – **Scalability**

**Which is impossible!!!**

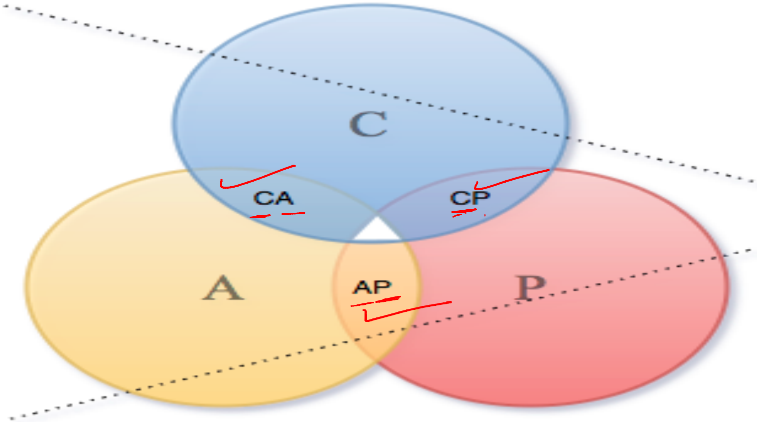
**According to CAP theorem**

Consistency

Availability

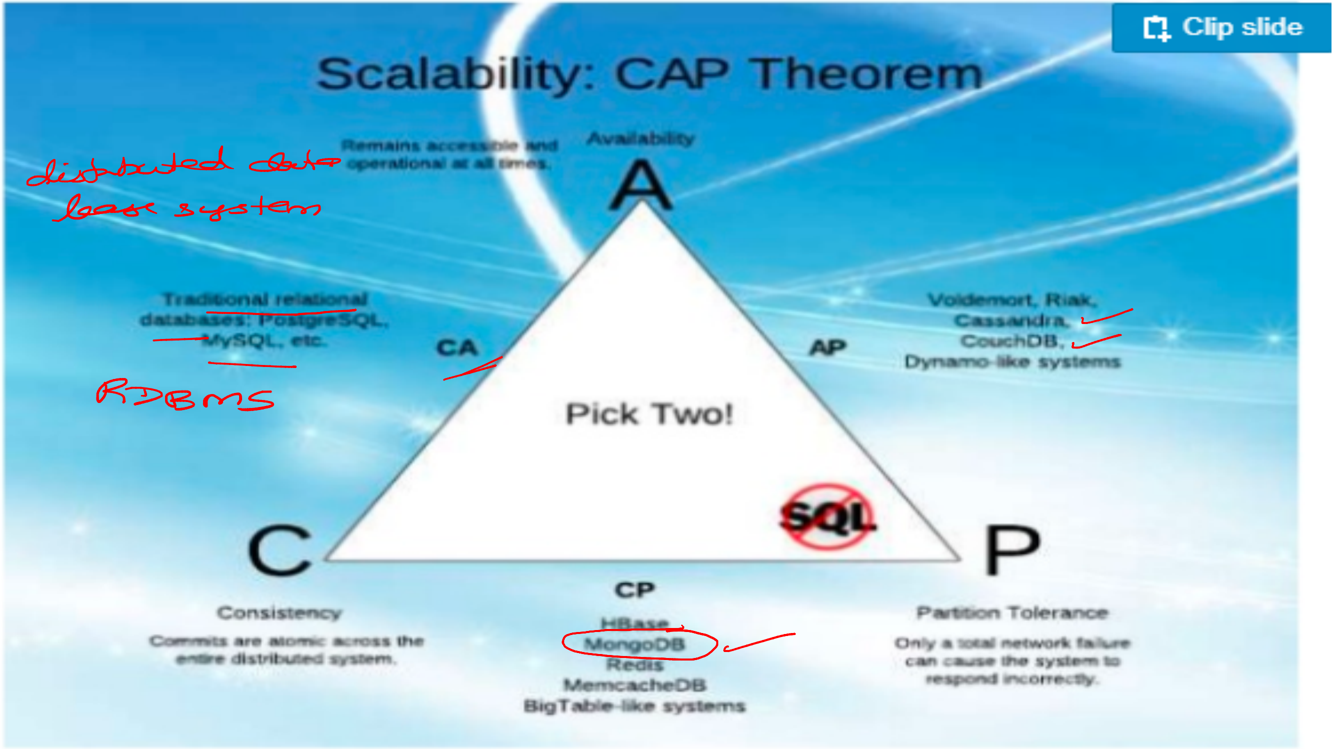
Partition Tolerance

- The CAP Theorem states that, in a distributed system (a collection of interconnected nodes that share data.), you can only have two out of the following three guarantees across a write/read pair: Consistency, Availability, and Partition Tolerance - one of them must be sacrificed.


C
CA
CP
A
AP
P

- **Consistency -** A read is guaranteed to return the most recent write for a given client.
- **Availability** - A non-failing node will return a reasonable response within a reasonable amount of time (no error or timeout).
- **Partition Tolerance -** The system will continue to function when network partitions errors occur.

- The CAP theorem categorizes systems into three categories:
- CP (Consistent and Partition Tolerant) — At first glance, the CP category is confusing, i.e., a system that is consistent and partition tolerant but never available. CP is referring to a category of systems where availability is sacrificed only in the case of a network partition.
- CA (Consistent and Available) — CA systems are consistent and available systems in the absence of any network partition. Often a single node's DB servers are categorized as CA systems. Single node DB servers do not need to deal with partition tolerance and are thus considered CA systems. The only hole in this theory is that single node DB systems are not a network of shared data systems and thus do not fall under the preview of CAP.
- AP (Available and Partition Tolerant) — These are systems that are available and partition tolerant but cannot guarantee consistency.



# Scalability: CAP Theorem

distributed data base system

Availability

Remains accessible and operational at all times.

A

CA

Traditional relational databases: PostgreSQL, MySQL, etc.

RDBMS

AP

Voldemort, Riak, Cassandra, ✓ CouchDB, ✓ Dynamo-like systems

Pick Two!

SQL

C

Consistency

Commits are atomic across the entire distributed system.

CP

HBase MongoDB ✓ Redis MemcacheDB BigTable-like systems

P

Partition Tolerance

Only a total network failure can cause the system to respond incorrectly.