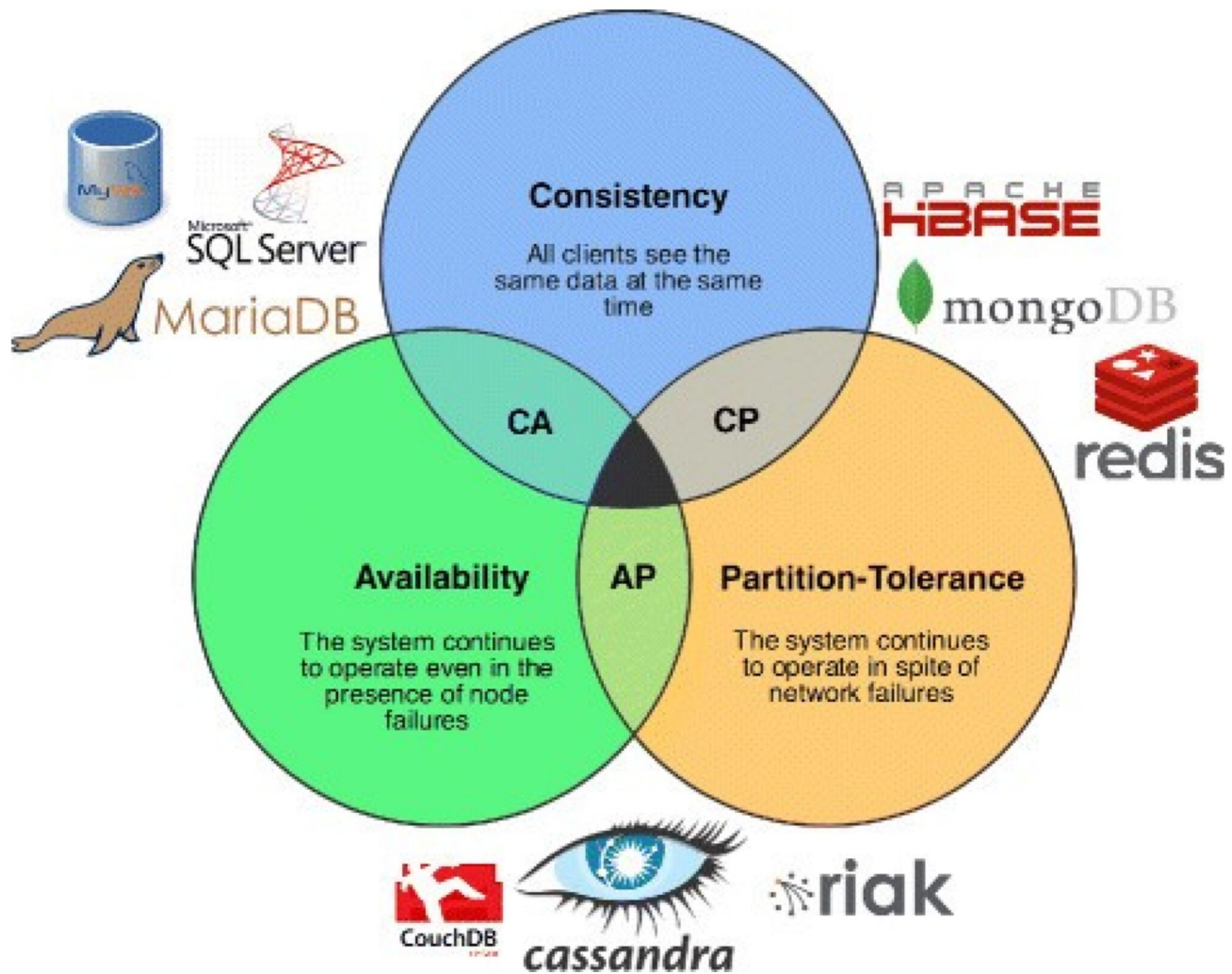# CAP THEOREM
(Brewer's Theorem)

CAP — Consistency, Availability, Partition Tolerance

→ CAP theorem states that it is impossible for a distributed system to offer more than two out of three following properties:

Simply put,

→ Consistency : Every read should receive the recent write or an error.

→ Availabilty : Every request will receive(non-error) response, without the gurantee that the response contains the most recent write.

→ Partition Tolerance : The system(cluster) should continue to operate despite any number of communication breakdowns or network failures between nodes. (i.e, a lost connection or temporarily delayed connections between nodes).

Network failures are inevitable in distributed systems , so the system should either choose one of the following in case of network partition :

→ Choosing consistency at the cost of availability :

   In this case, an error is returned (say data out-of-date error or data unavailable error) if a particular information is not guranateed to be up-to-date.

→ Choosing availability at the cost of consistency :

   In this case, the system will manage to return the most recent version of data even though it is not guaranteed to be the up-to-date.

NOTE :   Database systems designed with traditional ACID guarantees such as RDBMS choose consistency over availability, whereas systems designed around the BASE guarantees, such as NoSQL for example, choose availability over consistency.

PACELC THEOREM

PACELEC theorem is an extension of **CAP** theorem.
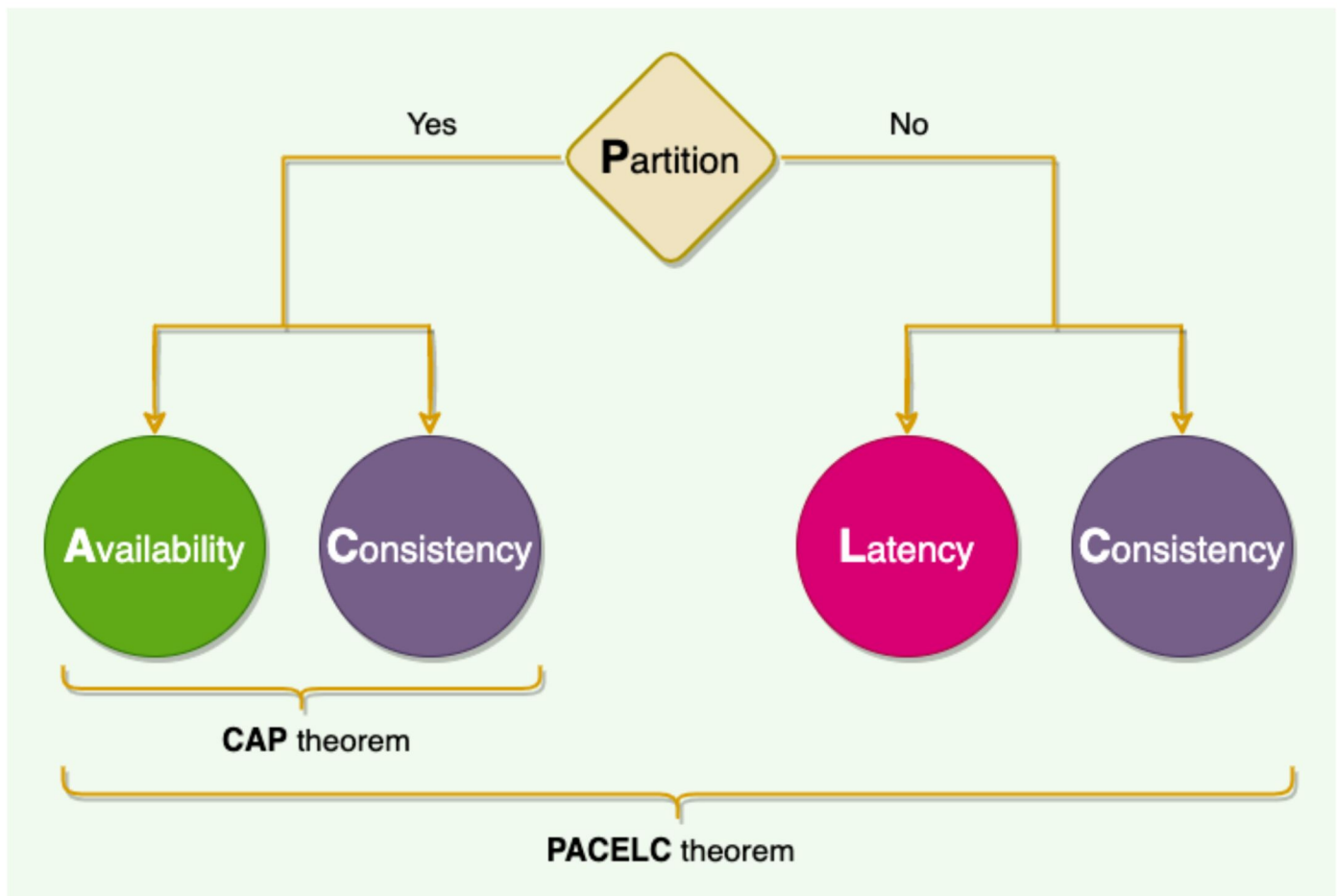
→ PACELC theorem states that in a distributed system :

In case of network partition(P) the system trades off between availability(A) and consistency(C) **Else(E)** the system has to tradeoff between latency(L) and consistency(C).

➡ CAP theorem does not address how a system behaves when there is no partition.

i.e, Let's say the system is serving a request but it's taking 10 mins to return a response. This latency is unacceptable.

➡ PACELC addresses this **key limitation** of CAP theorem.

➡ It states that when the system is functioning normally (without any partition) it either have to choose :

➡ To stay consistent at all times by ensuring data is updated sucessfully (which obviously increases latency).

➡ To return the available data at the moment thereby improving the latency without the guarantee that the data has the most recent updates (non-consistent data).

Santosh Kumar Mishra
SDE @Microsoft

Follow for more

 iamsantoshmishra

 iamsantoshmishra

 Interview Cafe

 Interview Cafe Notes