CAP Theorem

What is CAP Theorem?

CAP Theorem is used to explain the Non-Functional requirements and their relationship for a distributed system. The CAP in CAP Theorem stands for:

C: Consistency
A: Availability
P: Partition Tolerance

Let's revise what these terms mean.

Consistency: Consistency is uniformity in data. When a user requests data and the system reverts back with the same data irrespective of the geographical location, time etc. For a consistent system, the server at which data is updated or changed should effectively replicate the new data to all the nodes before the user reads the data from any node.

Availability: Availability is the probability of whether a system will work as needed when the user wants to make a request. If an application responds every time the user makes a request, then the application is an available application.
Example: Google has a very supportive system and is 100% available/very high availability.

Partition Tolerance: Failures are frequent in distributed systems. During a failure, the system should not go down, and the system must handle the fault gracefully. This attribute of a system is known as partition tolerance.

The CAP Theorem states that it is possible to attain only two properties and the third would be always compromised. The system requirements should define which two properties should be chosen over the rest.
Example:

- The system designer can select Consistency and Partition Tolerance but the availability would be compromised then.
- The system designer can select Partition Tolerance and Availability but the consistency would be compromised then.

- The system designer can select Availability and Consistency but the Partition Tolerance would be compromised then.

Out of all these three desirable properties, Partition Tolerance should always be given priority else during a network breakdown, all the data would be lost. The designer can select or prioritize between availability and consistency.

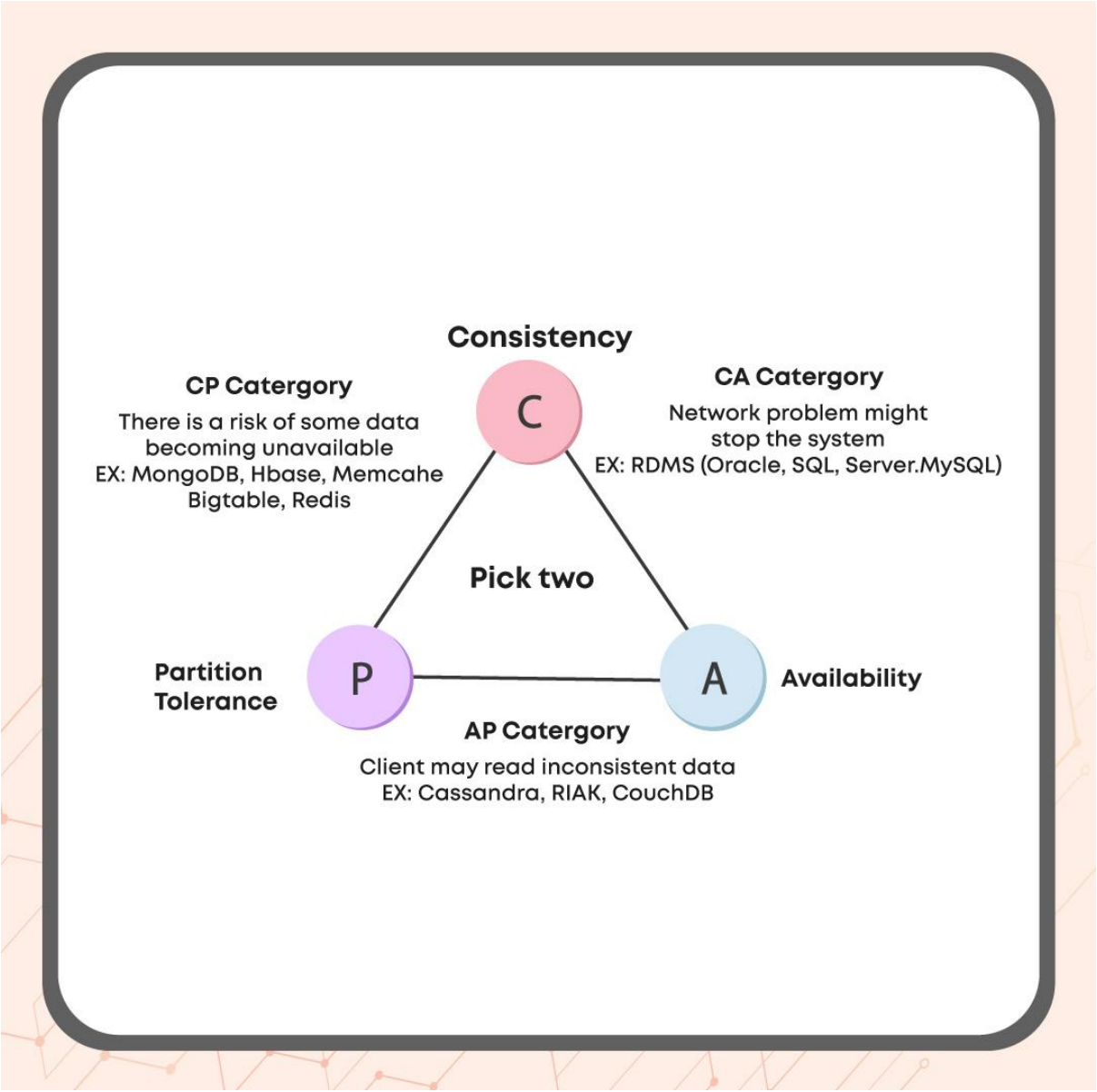

Fig: CAP Theorem