### What is Cassandra

Apache Cassandra is **highly scalable**, high performance, **distributed NoSQL database**. Cassandra is designed to handle huge amount of data across many commodity servers, providing **high availability without a single point of failure**.

Cassandra has a distributed architecture which is capable to handle a huge amount of data. Data is placed on different machines with more than one replication factor to attain a high availability without a single point of failure.

## Reasons behind its popularity

Cassandra is an Apache product. It is an open source, distributed and decentralized/distributed storage system (database). It is used to manage very large amounts of structured data spread out across the world. It provides high availability with no single point of failure.

## Important Points of Cassandra

* Cassandra is a column-oriented database.
* Cassandra is scalable, consistent, and fault-tolerant.
* Cassandra's distribution design is based on Amazon's Dynamo and its data model on Google's Bigtable.
* Cassandra is created at Facebook. It is totally different from relational database management systems.
* Cassandra follows a Dynamo-style replication model with no single point of failure, but adds a more powerful "column family" data model.
* Cassandra is being used by some of the biggest companies like Facebook, Twitter, Cisco, Rackspace, ebay, Twitter, Netflix, and more.

## Features of Cassandra

There are a lot of outstanding technical features which makes Cassandra very popular. Following is a list of some popular features of Cassandra:

## High Scalability

Cassandra is highly scalable which facilitates you to add more hardware to attach more customers and more data as per requirement.

## Rigid Architecture

Cassandra has not a single point of failure and it is continuously available for business-critical applications that cannot afford a failure.

## Fast Linear-scale Performance

Cassandra is linearly scalable. It increases your throughput because it facilitates you to increase the number of nodes in the cluster. Therefore, it maintains a quick response time.

## Fault tolerant

Cassandra is fault tolerant. Suppose, there are 4 nodes in a cluster, here each node has a copy of same data. If one node is no longer serving then other three nodes can served as per request.

## Flexible Data Storage

Cassandra supports all possible data formats like structured, semi-structured, and unstructured. It facilitates you to make changes to your data structures according to your need.

## Easy Data Distribution

Data distribution in Cassandra is very easy because it provides the flexibility to distribute data where you need by replicating data across multiple data centers.

## Transaction Support

Cassandra supports properties like Atomicity, Consistency, Isolation, and Durability (ACID).

## Fast writes

Cassandra was designed to run on cheap commodity hardware. It performs blazingly fast writes and can store hundreds of terabytes of data, without sacrificing the read efficiency.

## Cassandra Architecture

Cassandra was designed to handle big data workloads across multiple nodes without a single point of failure. It has a peer-to-peer distributed system across its nodes, and data is distributed among all the nodes in a cluster.

* In Cassandra, each node is independent and at the same time interconnected to other nodes. All the nodes in a cluster play the same role.
* Every node in a cluster can accept read and write requests, regardless of where the data is actually located in the cluster.
* In the case of failure of one node, Read/Write requests can be served from other nodes in the network.

## Summary

* Apache Cassandra is **highly scalable distributed NoSQL database.**
* It provides **high availability without a single point of failure**.
* Data is placed on different machines with more than one replication factor to attain a **high availability without a single point of failure.**
* Cassandra is a **column-oriented** database.
* Cassandra is scalable, consistent, and fault-tolerant.
* Cassandra's distribution design is based on Amazon's **Dynamo** and its data model on Google's **Bigtable**.
* Cassandra follows a Dynamo-style replication model with no single point of failure, but adds a more powerful "column family" data model.

## Fault tolerant

Cassandra is fault tolerant. Suppose, there are 4 nodes in a cluster, here each node has a copy of same data. If one node is no longer serving then other three nodes can served as per request.