Roll No. 171347

Discuss about the architecture and properties of Cassandra.

Ans:

The Apache Cassandra architecture is built to store enormous volumes of data with scalability, availability, and dependability.

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

All the nodes in a cluster play the same role. Each node is independent and at the same time interconnected to other nodes.

Each node in a cluster can accept read and write requests, regardless of where the data is actually located in the cluster.

When a node goes down, read/write requests can be served from other nodes in the network.

The architecture of Cassandra can be illustrated as follows:

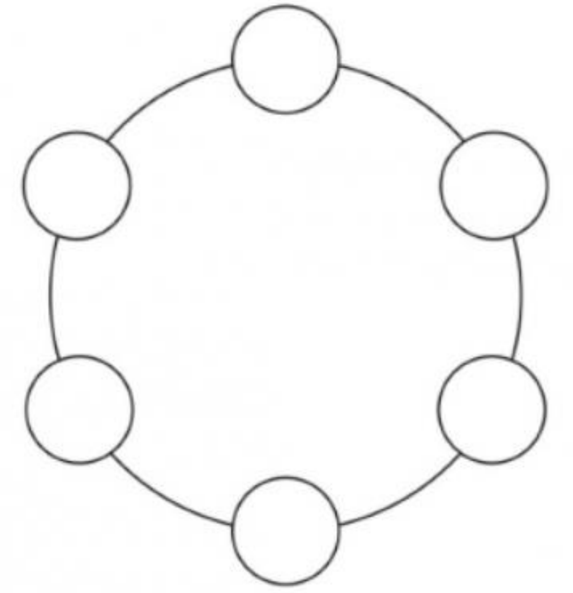

Figure 1. Architecture of Cassandra

The architecture of Cassandra greatly contributes to its being a database that scales and performs with continuous availability. Rather than using a legacy of RDBMS master-slave or a manual and difficult-to-maintain sharded design, Cassandra has a masterless “ring” distributed architecture that is elegant, and easy to set up and maintain.

In Cassandra, all nodes are the same; there is no concept of a master node, with all nodes communicating with each other via a gossip protocol.

Cassandra’s built-for-scale architecture means that it is capable of handling large amounts of data and thousands of concurrent users/operations per second, across multiple data centers, as easily as it can manage much smaller amounts of data and user traffic. To add more capacity, you simply add new nodes in an online fashion to an existing cluster.

Cassandra’s architecture also means that, unlike other master-slave or sharded systems, it has no single point of failure and therefore offers true continuous availability and uptime.

Writing and Reading Data

One of Cassandra’s hallmarks is its fast I/O operation capability for both writing and reading data.

Data is written to Cassandra in a way that provides both full data durability and high performance. From a high level perspective, data written to a Cassandra node is first recorded in a commit log and then written to a memory-based structure called a memtable. When a memtable's size exceeds a configurable threshold, the data is flushed to disk and written to an SStable (sorted strings table), which is immutable.

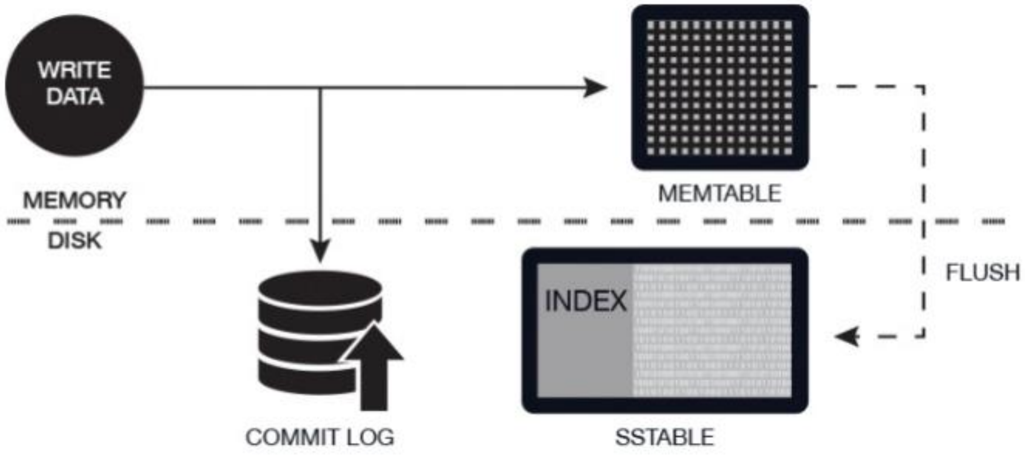

Figure 2. The Cassandra Write Path

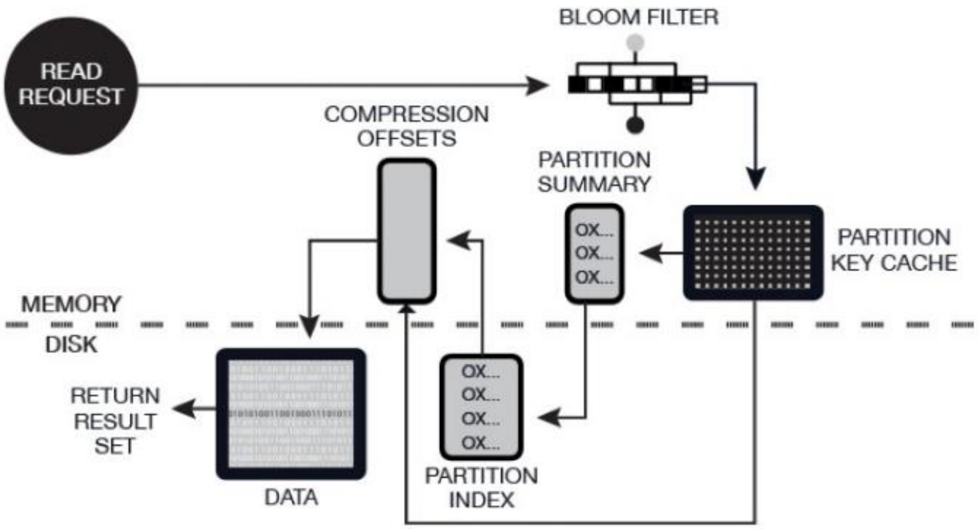

Figure 3. The Cassandra Read Path

There are a lot of outstanding technical features which makes Cassandra very popular. The properties of Cassandra are as enlisted below:

1. **High Scalability**

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

2. **Rigid Architecture**

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

3. **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.

4. **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.

5. **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.

6. **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.

7. **Transaction Support**

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

8. **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.