CASSANDRA INTRODUCTION: WHAT IS APACHE CASSANDRA?

Apache Cassandra is a distributed database management system that is built to handle large amounts of data across multiple data centers and the cloud. It's highly scalable, offers high availability, and has no single point of failure. Written in Java, it's a NoSQL database offering many things that other NoSQL and relational databases cannot.

Cassandra was originally developed at Facebook for their inbox search feature. Facebook open-sourced it in 2008, and Cassandra became part of the Apache Incubator in 2009. Since early 2010, it has been a top-level Apache project. It's currently a key part of the Apache Software Foundation and can be used by anyone wanting to benefit from it.

Cassandra stands out among database systems and offers some advantages that other systems cannot. Its ability to handle high volumes makes it particularly beneficial for major corporations. As a result, it's currently being used by many large businesses including Apple, Facebook, Instagram, Uber, Spotify, Twitter, Cisco, Rackspace, eBay, and Netflix.

To fully appreciate what Apache Cassandra is and what it can do, it's helpful to first understand NoSQL databases and to then look more specifically at Cassandra's architecture and capabilities. Doing so provides a good introduction to the system and can help to determine if it's right for your business.

What is a NoSQL Database?

A NoSQL, often referred to as "not only SQL", database is one that stores and retrieves data without requiring data to be stored in tabular format. Unlike relational databases, which require a tabular format, NoSQL databases allow for unstructured data. This type of database offers a simple design,

horizontal scaling, and extensive control over availability. It doesn't require fixed schema, it allows for easy replication, it has a simple API, it's generally consistent, and it can handle large amounts of data.

That said, there are pros and cons of using this type of database. While NoSQL databases offer many benefits, they also have drawbacks. Generally, they only support simply query language, are just "eventually consistent," and don't support transactions.

Nevertheless, they are effective with huge amounts of data and offer easy, horizontal scaling, making this type of system a good fit for many large businesses. Apache Cassandra, Apache HBase, and MongoDB are some of the most popular and effective NoSQL databases.

What Makes Apache Cassandra Unique?

Cassandra is one of the most efficient and widely-used NoSQL databases. One of the key benefits of this system is that it offers highly-available service and no single point of failure. This is key for businesses that can afford to have their system go down or to lose data. With no single point of failure, it offers truly consistent access and availability.

Another key benefit of Cassandra is the massive volume of data that the system can handle. It can effectively and efficiently handle huge amounts of data across multiple servers. Plus, it is able to fast write huge amounts of data without affecting the read efficiency. Cassandra offers users "blazingly fast writes," and the speed or accuracy is unaffected by large volumes of data. It is just as fast and as accurate for large volumes of data as it is for smaller volumes.

Another reason that so many enterprises utilize Cassandra is its horizontal scalability. Its structure allows users to meet sudden increases in demand, as it allows users to simply add more hardware to accommodate additional customers and data. This makes it easy to scale without shutdowns or major adjustments needed. Additionally, its linear scalability is one of the things that helps to maintain the system's quick response time.

Some other benefits of Cassandra include:

- Flexible Data Storage - Cassandra can handle structured, semi-structured, and unstructured data, giving users flexibility with data storage.
- Flexible Data Distribution - Cassandra uses multiple data centers, which allows for easy data distribution wherever or whenever needed.
- Supports ACID - The properties of ACID (atomicity, consistency, isolation, and durability) are supported by Cassandra.

Cleary, Apache Cassandra offers some discrete benefits that other NoSQL and relational databases cannot. With continuous availability, operational simplicity, easy data distribution across multiple data centers, and an ability to handle massive amounts of volume, it is the database of choice for many enterprises.

How does Cassandra Work?

Apache Cassandra is a peer-to-peer system. Its distribution design is modeled on Amazon's Dynamo, and its data model is based on Google's Big Table. The basic architecture consists of a cluster of nodes, any and all of which can accept a read or write request. This is a key aspect of its

architecture, as there are no master nodes. Instead, all nodes communicate equally.

While nodes are the specific location where data lives on a cluster, the cluster is the complete set of data centers where all data is stored for processing. Related nodes are grouped together in data centers. This type of structure is built for scalability and when additional space is needed, nodes can simply be added. The result is that the system is easy to expand, built for volume, and made to handle concurrent users across an entire system.

Its structure also allows for data protection. To help ensure data integrity, Cassandra has a commit log. This is a backup method and all data is written to the commit log to ensure data is not lost. The data is then indexed and written to a memtable. The memtable is simply a data structure in the memory where Cassandra writes. There is one active memtable per table.

When memtables reach their threshold, they are flushed on a disk and become immutable SSTables. More simply, this means that when the commit log is full, it triggers a flush where the contents of memtables are written to SSTables. The commit log is an important aspect of Cassandra's architecture because it offers a failsafe method to protect data and to provide data integrity.

Who Should Use Cassandra?

If you need to store and manage large amounts of data across many servers, Cassandra could be a good solution for your business. It's ideal for businesses that can't afford for data to be lost or that can't have their database down due to the outage of a single server. Further, it's also easy to use and easy to scale, making it ideal for businesses that are consistently growing.

At its core, Apache Cassandra's structure is "built-for-scale" and can handle large amounts of data and concurrent users across a system. It lets major corporations store massive amounts of data in a decentralized system. Yet, despite the decentralization, it still allows users to have control and access to data.

And, data is always accessible. With no single point of failure, the system offers true continuous availability, avoiding downtimes and data loss. Additionally, because it can be scaled by simply adding new nodes, there is constant uptime and no need to shut the system down to accommodate more customers or more data. Given these benefits, it's not surprising that so many major companies utilize Apache Cassandra.