**MANASI**

Hello everyone my name is Manasi Patil, and I welcome you all for the group discussion on google cloud and sql vs nosql database. I hope you all in the pinkest of your health bcoz we are going to have some fun today along with my group members Uma Chaitanya and Bhavin

so let us move further and start,

before knowing about the google cloud we need to understand about the cloud and the situation before cloud came into the picture

so, what appended back,

suppose you want to host a website, you have to buy stack of servers, and we all know the server are very expensive so we ended up paying lot of money

next was the issue of traffic

now as we all know if you are hosting a website, we are dealing with traffic that is not constant throughout the day

other thing was you have to monitor and maintain the severs

**UMA**

yeah, this is a really big problem, troubleshooting problems can be tedious and may conflict with your business goals

**BHAVIN**

yes, like for the necessary growth which completes your goals and make the appropriate decisions right for you but at same time you also need to focus on the infrastructure as Manasi has mentioned and you had to maintain those servers due to which you cannot focus more on your business so either you had to do multitasking or you need to hire more people for that purpose so again you ended up paying more money.

CHAITANYA

I would like to add one point here that as Manasi has discussed traffic on server is not constant, and since it varies you are not certain about the patterns, say for an example I need to host a website and for that purpose I reserved 2 petabytes of total memory or space based on traffic, but what if the traffic is very high then it will consume all of my 2 petabytes, and what if the traffic is very low for a certain hours of the day, I’m actually not utilizing my server so I end up paying more money on server than I should be.

**BHAVIN**

Yeah, because of these things they become an issue for you as you paying more money, and you will not get sufficient time to take decisions, which is understandable more than in one way, and makes more trouble in monitoring and mentioning.

CHAITANYA

apart from that one important thing we need to consider here the amount of data that is being generated and that was being generated then, then it was okay but now a days when you take look at it the amount of data that is generated now is enormous, and this is another reason why cloud has become so important

**UMA**

Also, we know that everything is going online nowadays, we shop online, we transfer money online, we buy food online, even we are having this discussion online, and many other things that means we have lot of data that is generated every day and that data is the digital data. And we need space to store this huge data and to maintain it, so yes, all these issues your cost, monitoring, maintaining, providing sufficient space everything was taken care by cloud.

Hence

"The cloud" refers to servers that are accessed over the Internet, and the software and databases that run on those servers. Cloud servers are located in data centres all over the world. also, the cloud enables users to access the same files and applications from almost any device, because the computing and storage takes place on servers in a data centre, instead of locally on the user device.

CHAITANYA

This is why a user can log in to their Instagram account on a new phone after their old phone breaks and still find their old account in place, with all their photos, videos, and conversation history.

**MANASI**

I think the concept of cloud is pretty clear. It is time to discuss about the google cloud

**Google Cloud Platform** (**GCP**), offered by Google, is a suite of cloud computing services that runs on the same infrastructure that Google uses internally for its end-user products, such as Google Search, Gmail, Google Drive, and YouTube.

In April 2008, Google announced App Engine, a platform for developing and hosting web applications in Google-managed data centres, which was the first cloud computing service from the company. The service became generally available in November 2011. Since the announcement of App Engine, Google added multiple cloud services to the platform.

**BHAVIN**

**And Here I would like to introduced you guys what is SQL databases?**

Which is nothing but commonly referred as a relational database because it’s based on the relational model introduced in the early 70s by Edgar F. Codd. This relational model defines a methodology for organizing structured data into tables with columns and rows, also it defines the relationships between those tables. And, there’s a lot more to know about relational databases —and I’ve been fairly loose with its terminology—but the important point to know is that the relational database soon became the de facto standard for storing and managing data in organizations of all sizes after its introduction and remains a prevalent technology to this day. The language is also used to store, manipulate, and retrieve data from those tables.

**UMA**

Listen Everyone should know that SQL has been adopted by both the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO) and is well known and widely supported by developers around the world.

CHAITANYA

Yeah, Guys I guess that’s enough info about SQL Now let me tell all of you about NoSQL

**NoSQL**

When people use the term “NoSQL database,” they typically use it to refer to any non-relational database. Some say the term “NoSQL” stands for “non-SQL” while others say it stands for “not only SQL.” Either way, most agree that NoSQL databases are databases that store data in a format other than relational tables.

Each NoSQL database has its own unique features. At a high level, many NoSQL databases have the following features:

* [Flexible schemas](https://docs.mongodb.com/manual/core/data-modeling-introduction/#flexible-schema)
* [Horizontal scaling](https://www.mongodb.com/basics/scaling)
* [Fast queries due to the data model](https://docs.mongodb.com/manual/core/data-modeling-introduction/#document-structure)
* [Ease of use for developers](https://www.mongodb.com/why-use-mongodb)

**UMA**

Also, there are 4Types of NoSQL databases

Over time, four majors [types of NoSQL databases](https://www.mongodb.com/scale/types-of-nosql-databases) emerged:

document databases,

key-value databases,

wide-column stores,

and graph databases.

So, **Document databases** store data in documents, right?

**MANASI**

**Yes, right right!**

**Document databases** store data in documents similar to JSON (JavaScript Object Notation) objects. In which Each document contains pairs of fields and values. And The values can typically be a variety of types including things like strings, numbers, Booleans, arrays, or objects.

* **Key-value databases** are a simpler type of database where each item contains keys and values.
* **Wide-column stores** store data in tables, rows, and dynamic columns.
* **Graph databases** store data in nodes and edges. Nodes typically store information about people, places, and things, while edges store information about the relationships between the nodes.

CHAITANYA

**Till now we discuss sql and non sql but now I would like to talk about difference between** google cloud SQL database vs google cloud NoSQL data base.

SQL databases are primarily called as Relational Databases (RDBMS); whereas NoSQL database are primarily called as non-relational or distributed database.

Also, SQL databases defines and manipulates data based structured query language (SQL). Seeing from a side this language is extremely powerful. SQL is one of the most versatile and widely-used options available which makes it a safe choice especially for great complex queries.

**MANASI**

But from other side it can be restrictive. SQL requires you to use predefined schemas to determine the structure of your data before you work with it.

Also, all of your data must follow the same structure. This can require significant up-front preparation which means that a change in the structure would be both difficult and disruptive to your whole system.

CHAITANYA

A NoSQL database has dynamic schema for unstructured data. Data is stored in many ways which means it can be document-oriented, column-oriented, graph-based or organized as a Key Value store. This flexibility means that documents can be created without having defined structure first. Also, each document can have its own unique structure. The syntax varies from database to database, and you can add fields as you go.

**BHAVIN**

Yes right, also In SQL and NoSQL, there is difference in their scalability,

In almost all situations, SQL databases are vertically scalable. That means, you can increase the load on a single server by increasing things like RAM, CPU or SSD. But on the other hand, NoSQL databases are horizontally scalable. This means that you handle more traffic by sharing, or adding more servers in your NoSQL database. It is like adding more floors to the same building versus adding more buildings to the neighborhood. Which makes NoSQL larger and more powerful database, and now we are making these databases as preferred choice for large or ever-changing data sets.

**UMA**

We must know there’s difference in there structure , as SQL databases are table-based on the other hand NoSQL databases are either key-value pairs, graph databases or wide-column stores. This makes relational SQL databases a better option for applications that require multi-row transactions such as an accounting system that were built for a relational structure.

**MANASI**

Yeah, absolutely right Bhavin, do you know they follows different properties likeSQL databases follow ACID properties Yes (i.e., Atomicity, Consistency, Isolation and Durability) whereas the NoSQL database follows the Brewers CAP theorem (i.e., Consistency, Availability and Partition tolerance).

Also, a lot of independent consultations are there who can help you with SQL database for a very large-scale deployments but for some NoSQL database you still have to rely on community support and only limited outside experts are available for setting up and deploying your large-scale NoSQL deployments.

Some examples of SQL databases include PostgreSQL, MySQL, Oracle and Microsoft SQL Server. NoSQL database examples include Redis, Raven DB Cassandra, MongoDB, Bigtable, HBase, Neo4j and CouchDB.

**UMA**

So as my group members discuss about their difference now the question is Which one is better SQL or NoSQL

The decision between SQL and NoSQL will depend largely on the workloads you plan to support and the structure and amount of data. However, you should also consider the differences in the database products themselves, such as maturity, stability, licensing fees, vendor support, and the extent and participation of the developer communities.

**Consider SQL databases when…**

* Your data is highly structured, and that structure doesn’t change frequently
* You support transaction-oriented systems such as accounting or financial applications
* You require a high degree of data integrity and security
* You routinely perform complex queries, including ad hoc requests
* You don’t require the scale-out capabilities that NoSQL offers
* So now lets see where to consider NoSQL databases.

Bhavin

**And now I will tell you guys when to use NoSQL databases…**

* When You’re working with large amounts of unstructured or semi-structured data that doesn’t fit the relational model, then you should consider for NoSQL databases.
* Whenever you require the flexibility for your database or want more choices for the data model, you can use NoSQL databases.
* If you want to multiple your database system across various locations you might need NoSQL database for that.
* Also, when you want to streamline development and avoid the overhead of a more structured approach or your applications might don’t require the level of data integrity which is offered by SQL databases, you can go for NoSQL.

**Manasi**

Again, these are just guidelines. You should consider each situation individually, taking into account the shape of your data and workload requirements. At the same time, keep in mind that you’re not limited to one database type over the other. Many organizations have implemented both SQL and NoSQL database systems to meet their different requirements, making it possible to get the best of both worlds.