When to pick a NoSQL Database?

In this lesson, you will discover when to choose a NoSQL database over any other kind of database.

We'll cover the following

- Handling a large number of read write operations
- Flexibility with data modeling
- Eventual consistency over strong consistency
- Running data analytics

Handling a large number of read write operations#

Lean towards *NoSQL* databases when you need to scale fast. When do you generally need to scale fast?

The answer is when there are a large number of read-write operations on your website and when dealing with a large amount of data. Since NoSQL databases have the ability to add nodes on the fly, they can handle more concurrent traffic and bigger amounts of data with minimal latency.

Flexibility with data modeling#

modeling#

The second cue is during the initial phases of development when you are not sure about the data model, or the database design, and things are expected to change at a rapid pace. *NoSQL* databases offer us more flexibility.

Eventual consistency over strong consistency#

It's preferable to pick *NoSQL* databases when it's okay for us to give up *strong consistency* and when we do not require *transactions*.

A good example of this is a social networking website like Twitter. When a celebrity's tweet blows up and everyone is liking and re-tweeting it from around the world, does it matter if the count of *likes* goes up or down a bit for a short while?

The celebrity definitely would not care if instead of the actual 5 million 500 *likes*, the system shows the *like* count as 5 million 250 for a short while.

When a large application is deployed on hundreds of servers spread across the globe, the geographically distributed nodes take some time to reach a global consensus.

Until they reach a consensus, the value of the entity is inconsistent. The value of the entity eventually becomes consistent after a short while. This is what *eventual consistency* is.

However, the inconsistency does not mean that there is any sort of data loss. It just means that the data takes a short while to travel across the globe via the internet cables under the ocean to reach a global consensus and become consistent.

and become consistent.

We experience this behavior all the time, especially on YouTube. Often you might see a video with 10 views and 15 likes. How is this even possible?

It's not. The actual views are already more than the likes. It's just the count of views is inconsistent and takes a short while to update. I will discuss *eventual consistency* in more detail further down the course.

Running data analytics#

NoSQL databases also fit best for *data analytics* use cases, where we have to deal with an influx of massive amounts of data.

There are dedicated databases for use cases like this such as *Time-Series databases*, *Wide-Column*, *Document Oriented* etc. I'll talk about each of them later on in the course.

Right now, let's take a look into the performance comparison of SQL and NoSQL tech.

← Back

Features of NoSQL Databases

Next →

Is NoSQL More Performant Than SQL?

Mark as Completed

Report an Issue