

SQL vs NoSQL

Visual Research Story 1

By Carlos E. Belardo

Date: 07/12/19

Adobe Spark Link:

<https://spark.adobe.com/page/QS9dVR947i9QZ/>



Relational Vs. Non-Relational (or SQL Vs. NoSQL)

One of the main differences between SQL and NoSQL is how the data is stored in the database. When storing data SQL uses separate tables for its database, each table stores its own type of data. When using NoSQL data is not stored and organized in tables but is stored and organized in documents. When using complex queries SQL is better for such applications since it is able to be used since they are more powerful. NoSQL is preferred for big sets of data.

Examples of SQL management systems : MySQL, MS-Server, and Oracle.

Examples of NoSQL management systems : MongoDB, CouchDB, and Redis.

A large, dense cloud of "twitter" text in various sizes and colors (blue, white, grey) on a dark blue background. The word "twitter" is repeated numerous times in a non-linear, overlapping arrangement, creating a sense of motion and frequency.

Three NoSQL Twitter Features

HBase: Twitter uses HBase for its people search feature. HBase is used on top of Hadoop to store data into clusters which SQL is not able to do. With Twitter's scale being so big for analytics a spokesman for Twitter said "MySQL isn't efficient at doing analytics at the scale that Twitter needs". For its people search HBase uses complex user calculations and uses custom indexing for searches. The Pros of HBase are fast processing, large data sets, scalability, and consistency to name a few. The disadvantages are there is a possible point of failure, there is no handling for joins in the database, and there is no built-in authentication or permissions when using HBase.

FlockDB: Twitter uses FlockDB to see who follows who. For example checking to see who is following you in your followers list. With the data from FlockDB you can also see who you follow back since it is used to show intersections in the database. Some of FlockDB's pros are it is a much simpler graph database, it is designed for real-time, low-latency, and high throughput environments. The disadvantage of FlockDB is it was only designed to handle Twitter's single followers following feed.

Cassandra: Cassandra is used for tweets. The reason Twitter switched to Cassandra was because of its flexible

schema and it has high write throughput. The advantages of using Cassandra is there is no single or centralized point of failure, it is suited for multiple data-center deployment, redundancy, failover and disaster recovery. The disadvantages are there are no concept of joins connections and when retrieving data there is no way to organize it such as using a GROUP BY or ORDER BY,



Facebook Feature

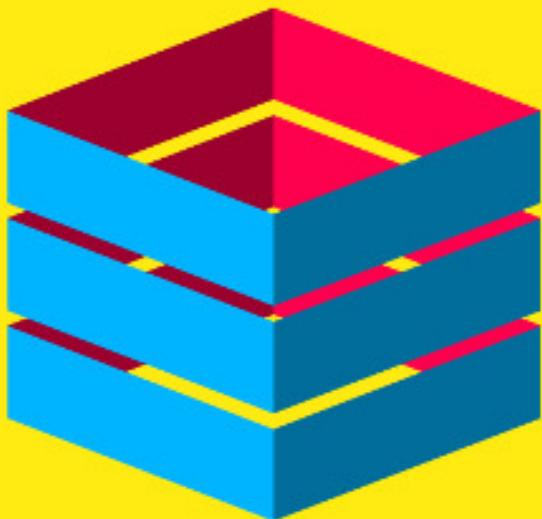
How To

One Facebook feature Facebook uses a relational database for is wall post. The way Facebook uses MySQL is as a generic engine for data manipulation. MySQL advantage is it is very efficient as a storage engine. One disadvantage of MySQL for Facebook is scalability. When applications start and do not expect a very fast growth rate and then encounter it they are not prepared for the issues that come along with MySQL and its scalability issues.

facebook

Four NoSQL Database Types

- Key-Value Store: It has a big hash table of keys and values.
- Document based Store : Stores documents made up of tagged elements.
- Column based store: Each Storage block contains data from only one column.
- Graph Based: Network database that uses edges and nodes to represent data and store data.



nosql



mong
membase

ORACLE
Coherence

lasticsearch.

Ap
S

Five NoSQL Databases

- **Apache Cassandra :** Cassandra is a database management system that is designed to handle large amounts of data across multiple servers. Apache Cassandra is an open source database that can handle large amounts of data across the world. The pros of Cassandra are its ability to be scaled massively, it is one of the few database systems that can process data in real time, while generating high performance and maintaining high availability.
- **MongoDB :** MongoDB organizes all of data with the use of JSON documents as the row, fields as the columns, and it is all grouped into collections as the tables. It is used for things like Blogs, content management, Maintaining location-based data, Mobile and social networking sites, and E commerce product catalogs. Pros of MongoDB are High performance, High availability, High scalability, and flexibility to name a few. The cons of MongoDB are there are no functions to bind logic and it is not ACID compliant.
- **Neo4j :** Neo4j is a graph-based database. It is used for managing of data reliably and effectively. Another use is storing data from social networks and recommendation sites and storing them in nodes and displaying them in graphs. Some of the pros for Neo4j are it is efficient with large data sets, it is capable of inserting or updating through a UI, and it provides a graphical representation of data sets. Some cons of

Neo4j are the servers for a large data set need to be well configured, It is tough to allocate memory, and the graphs can sometimes be complex.

- **Hadoop** : Hadoop is designed to help with huge amounts of data and limit financial and processing-time. Hadoop is used to store and manage vast amounts of data cheaply and efficiently. Some pros for Hadoop are it is cost effective, It can accept a variety of data, and its ease of use. Some of the cons are since Hadoop is written in Java it is vulnerable to security breaches, it only supports batch processing, and it has issues with large numbers of small files.
- **OrientDB** : OrientDB is a database that uses graphs and documents. It is known as a hybrid database because of this. It was designed to be a mix of both MongoDB and Neo4j. OrientDB is used by many companies such as Kyocera, Accenture, and Comcast to name a few. Some of the pros for OrientDB are it has support for clustering, it supports SQL and other query languages, and its fast performance. Some cons of OrientDB are it is not well known, Bulk inserts may cause an out of memory crash, and it could be a learning curve for some.



Weather App (Two NoSQL Solutions)

One NoSQL Database that could be used for a weather app is MongoDB. The reason MongoDB would work well for a weather based app is because of its ability to quickly add features to its range of Internet based weather information services. Also with MongoDB's ease of development and fast response times it would be perfectly suited for a weather app for real-time updates.

Another NoSQL database that could be used for a weather app would be Apaches Cassandra. Cassandra would be a good database management system to use for a weather app because of its ability to handle thousands of transactions per second. Also Cassandra can be used to distribute data globally and that data is easily accessible.



References

Dataflair Team. (2018, September 14). HBase Pros and Cons | Problems with HBase. Retrieved from <https://dataflair.training/blogs/hbase-pros-and-cons/>

Weil, K. (2010, April 21). NoSQL at Twitter (NoSQL EU 2010) . Retrieved from
https://www.slideshare.net/kevinweil/nosql-at-twitter-nosql-eu-2010/121-FlockDB_Realtime_distributed_Intersection_Temporal

Cassandra: Pros & Cons! (2012, September). Retrieved from <https://marsmedia.info/en/blog/cassandra-pros-cons-and-model>

Pachel, P. (2011, December 15). Why Facebook Uses MySQL for Timeline. Retrieved from
<https://mashable.com/2011/12/15/facebook-timeline-mysql/>

Harris, D. (2011, July 7). Facebook trapped in MySQL “fate worse than death.” Retrieved from
<https://gigaom.com/2011/07/07/facebook-trapped-in-mysql-fate-worse-than-death/>

Kumar, G. (n.d.). Exploring the Different Types of NoSQL Databases Part ii. Retrieved from <https://www.3pillarglobal.com/insights/exploring-the-different-types-of-nosql-databases>

Jayaram, P. (2016, November 30). When to Use (and Not to Use) MongoDB. Retrieved from <https://dzone.com/articles/why-mongodb>

Hussain, A. (2014, October 31). Top 5 NoSQL Databases. Retrieved from <https://hub.packtpub.com/top-5-nosql-databases/>

Jackson, J. (2014, October 14). The Weather Channel forecasts heavy NoSQL ahead. Retrieved from <https://www.pcworld.com/article/2833772/the-weather-channel-forecasts-heavy-nosql-ahead.html>

Henschen, D. (2014, October 29). NoSQL Database Choices: Weather Co. CIO's Advice. Retrieved from <https://www.informationweek.com/big-data/software-platforms/nosql-database-choices-weather-co-cios-advice/a/d-id/1317052>