Overview of NoSQL databases in Cloud Computing

Eshaq Rahmani[1]

[1]Agile Software Design Higher Diploma,Technological University of Shannon

May 17, 2025

Abstract

With increasing volumes of data, organizations are implementing new technological solutions to efficiently store and process this data. NoSQL databases are solutions that show improved performance over the traditional relational databases, in particular for big data problems. In this paper, we aim to give an overview of the prominent types of NoSQL databases today, their use cases and discuss advantages and disadvantages.

Keywords— ***NoSQL databases, Big data, Data models***

1 Introduction

With the proliferation of computer devices and internet access, data generated every day is increasing at a rapid rate. In comparison to traditionally structured and clean data, however, the big data generated today is categorized as unstructured, semi-structured and structured data. As a result, relational databases management systems (RDBMS) has increasingly been proven inefficient in storing and maintaining such data, where systems performance degrades rapidly with increasing data volumes and complexity[1][2].

NoSQL or "Not only SQL" are distributed databases that has gained popularity due to its improved performance over the conventional SQL databases. Unlike RDBMS, NoSQL generally support only simple queries, are non-relational and horizontally scalable[7]. NoSQL typically use a sharding model, where data and load is distributed across multiple machines. As a result, sharding can increase capacity and throughput horizontally by reducing individual server operations[8]. Hence, NoSQL databases can take advantage of low cost commodity hardware and increase data access efficiency by utilizing a non-relational, distributed system[7].

Tech giants such as Facebook, Google and Amazon have their own distributed systems for data storage. For instance, Facebook's Cassandra [1], Google's BigTable [3], Amazon's Dynamo [4], EBay's MongoDB are examples of organizations that created NoSQL databases to solve their specific big data problem. Adopting NoSQL means organization can efficiently analyze massive amounts of data and provide near real-time information on areas such as user behavior or predictive analysis, and many other important development areas[5].

In this paper, we outline the four major data models of non-relational databases, provide an overview of the popular and available NoSQL databases of each model, and study their advantages and disadvantages. The four major NoSQL data models are: Key-value stores, Document store, Column-family and Graph databases.

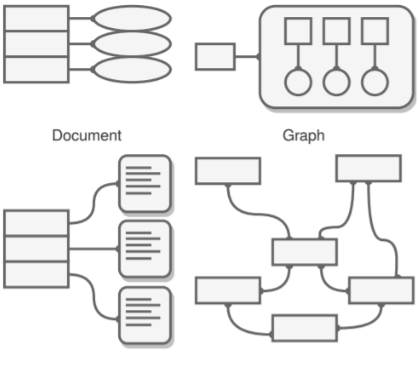

Figure 1: Structure of the four major types of NoSQL databases.

2 Key-Value Store

Key-value stored model is the simplest NoSQL database. The database mainly use distributed hash tables to create and store a pair of string formatted keys and their associated data value. Thus, a "key-value" pair is formed:

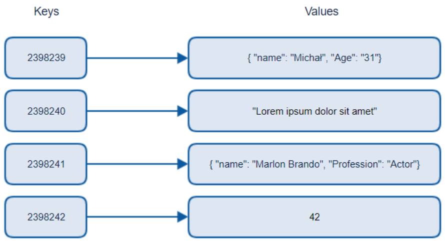
Keys
Values
2398239
{ "name": "Michal", "Age": "31"}
2398240
"Lorem ipsum dolor sit amet"
2398241
{ "name": "Marlon Brando", "Profession": "Actor"}
2398242
42

Figure 2: Key/Value pair in databases.

The model is efficient and offers high concurrency and scalability as well as high rates of read/insert operations. Drawbacks of the model are a lack of schema (redundancy), consistency and are also not suited for complex datasets due to its simple structure. However modern key/value models are designed such to favor the advantages listed over robust schema and consistency[10]. Among the variety of the open source key-value stores we discuss two notable databases that exist today: Amazon DynamoDB and Redis.

2.1 Amazon DynamoDB

Amazon DynamoDB is a fully managed, fast and predictable performing NoSQL database (key-value and document model) service with no-limit scalability. The service allows automatic replication of all data items (stored in SSDs[1]) across 3 Availability zones providing built-in high durability and data availability[4]. In particular, the database has a peer-to-peer architecture unlike the traditional master-slave architecture use consistent hashing. The disadvantages are deployability (only on AWS[2]),limitations on queries and no relational operations.

Amazon DynamoDB and key-value databases in general are suitable for web applications and e-commerce due to the simplicity and speed of operations[4]. Therefore, the popular database has many clients including streaming service Netflix [12] and the online learning application Duolingo[13].

2.2 Redis

Redis is an open source, fast, in memory data store which is in popular us today as a database, cache, queue and message broker[15]. The model is based on an advanced key-value architecture, offering a vast variety of data structure storage. In contrast to other key-value stores, Redis also provides lists, sets, hashes, range queries, bitmaps among others.

Main advantage of Redis is very fast response times and enabling millions of requests per second, achieving this using an in-memory dataset. Other advantages include ease-of-use (simple code), high availability, high scalability and replication[15]. A major drawback of Redis is that the size of data set is limited to the main memory available. Other disadvantages include performance degradation in large write/delete operations and security[16].

Due to its fast performance Redis is suitable for real-time applications in industries such as gaming, ad-tech, IoT and healthcare. Moreover, Redis is a great choice for in-memory cache and can be implemented as a complementary solutions to existing databases. Web page caching and query results caching are popular examples of caching with Redis. Therefore, many well known companies such as GitHub, Twitter, Pinterest and so forth that use Redis for their web applications[17].

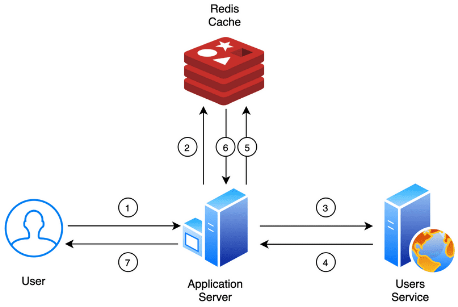

Figure 3: Redis Caching [17].

3 Document Store

Document store databases, as evident by the name, store data in the form of documents in a collection. The model is slightly more complex version of key-value databases, but data is stored in document forms such as PDF, XML, JSON and are accessed by a unique key. The key can be in the format string, path string or URI string.

Document store is schemaless, where fields of any length and variety can be contained in each document. Although more complex, the

[1] Solid State Drives
[2] Amazon Web Services

database model offers high performance, horizontal scaling and schema flexibility [16]. Drawbacks include inconsistency due to a more complex design, atomicity weakness as the model is non-relational and security[16].

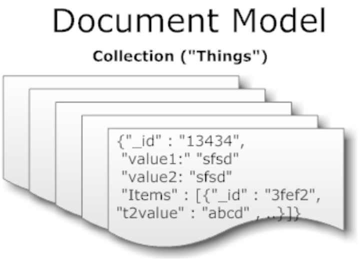

Figure 4: Example of document model database items.

A notable provider that use document store is MongoDB, which is discussed below. Others include CouchDB, RavenDB and many more.

3.1 MongoDB

MongoDB is written in C/C++ and uses Binary JSON format, and it is a schema-free document database. A driver is used to connect the database which is a benefit for developers [14]. It allows for easy replication, MapReduce and clustering. Other benefits are easy query language, ease of deployment, good documentation and many user clients. Disadvantages can include less flexible queries, no transactions and limited availability due to primary master server shutdown in a cluster[16]. Therefore, as per the CAP theorem[3], the use of MongoDB scarifies availability and chooses Consistency and Partitioning.

MongoDB is well suited for applications such as real-time analysis, web, content management system etc. The NoSQL database is currently used by many big companies such as The Guardian, BARCLAYS, Forbes and many others[18].

4 Column Model

Column model is the type most similar to relational databases. In this database the column is the smallest instance of data and it is a tuple containing a name, value and a timestamp. A unique row key is the specific identifier of a particular row. Column store have efficient storage as they are excellent at compression of data, which means a single column can hold large amounts of information[10]. The data model is therefore suitable for data mining and analytical applications where common operations can be performed quickly on large amounts data[16].

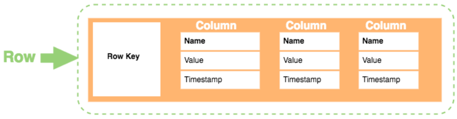

Figure 5: Column store data structure.

Examples of column-store databases include Casandra, BigTable, HBase etc. Below we will discuss Facebook's Cassandra and Google's BigTable database.

4.1 BigTable

BigTable is a high performance column-store database built on the Google File System (GFS) in C/C++. The database has three major components: a library link to clients, a master server and many small tablet servers. The master server tasks include schema changes, assigning tablets etc and tablet servers are used to manage tablets (similar to tables in RDBMS). The database offer fault tolerance, consistency and persistence.

BigTable is used in over sixty projects at Google, including web indexing, Google Earth, Google Analytics, Personalized Search and increasing [21]. The database is suitable for applications that need very high throughput and scalability and as an efficient storage engine for stream processing/analytics and machine-learning applications [21]. Examples include time-series data, marketing, finance, IoT etc [22].

4.2 Cassandra

Cassandra was developed and released by Apache Software Foundations in 2008. The database is based on both Google BigTable and Amazon's Dynamo model, involving concepts of key-values store and column store[1]. A table in Cassandra is a multi dimensional map indexed by a key, and columns are grouped together to form column families or super-columns. It has features such as dynamic schema, high availability, partition tolerance and persistence. A major disadvantage of Cassandra is that reads are comparatively slower than writes[1]. However, a big advantage is no single point of failure to increase fault tolerance.

Primarily used by Facebook, Apache Cassandra is now an open source database trusted by thousands of companies[19]. It is used for real

[3] CAP theorem: https://www.ibm.com/cloud/learn/cap-theorem

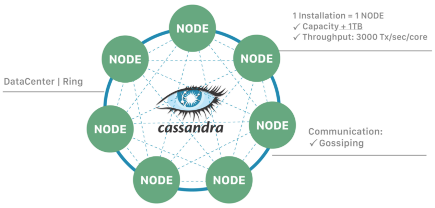

Figure 6: Cassandra: distributed database consisting of nodes[20].

time data analysis, banking and finance, online retail etc[1]. Facebook, for example, used Cassandra for their Inbox Search for efficient storing, indexing and searching messages[1].

5 Graph Store

Graph store models is based on graph theory that use a structure (node spaces) composed of nodes, edges and properties. The database mainly focuses on relationship between data where the node represents an object with a unique id, the edge describes the relationships between two nodes and the property represents a key-value pair attached to both the nodes and edges [23]. Furthermore, the model uses index free adjacency technique, meaning each node directly references its adjacent nodes. This in particular is cheaper and more efficient than normal indexing (increases traversal efficiency), as query time is proportional to graph searched[24].

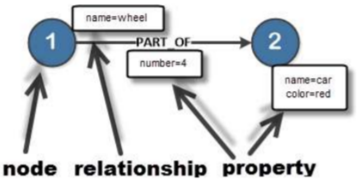

Figure 7: Graph store concept.

Graph store provides fast performance, ACID compliance and rollback support, and in particular excel at handling dynamic semi-structured data due to its use of graph theory. A few major drawbacks of the model include data structure and constraint limitations, difficulty in horizontal scalability and the model has no uniform query language[23][24][16].

The graph database is suitable for many variety of use cases, such as big data analytics in social media applications, semantic search for administration purposes, fraud detection etc. A major graph store database is Neo4j, which we will discuss below.

5.1 Neo4j

Neo4j is a robust, high performance and embedded persistence NoSQL database that is efficient in handling semi-structured data. The database was developed in Java by Neo Technology and was released in 2007. Neo4j has the largest graph community, offers high performance read/write operations, index free adjaceny to shorten read time and is very user-friendly [24]. Drawbacks of Neo4j, stated by their own research [24], are requiring a learning curve for usage, better tool support for using node spaces and difficulty in executing arbitrary queries on the database.

As the database is based on graph theory, Neo4j is used by many organizations for complex social network data, recommendation etc. Therefore, companies such as Adobe, Cisco, Mozilla and Lufthansa use Neo4j as a part of their applications[25].

Conclusion

NoSQL databases are modern technologies for handling and processing large volumes of data. The paper describes four major types of databases and gives examples of technologies implemented by organizations today. Although self-evident, each tool performs better for each specific issue needed to be solved. Future work can include an extensive performance analysis of the different NoSQL databases and compare them.

References

[1] A. Lakshman and P. Malik, “Cassandra: a decentralized structured storage system,” A C M SIGOPS Operating Systems Review, vol. 44, no. 2, pp. 35-40, 2010

[2] J. Dean and S. Ghemawat, “Mapreduce: simplified data processing on large clusters,” Communications o f the ACM, vol. 51, no. 1, pp. 107-113, 2008

[3] P. Zikopoulos, C. Eaton et al., Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media, 2011.

[4] G.DeCandia, D.Hastorun, M.Jampani, G.Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, “Dynamo: amazon’ s highly available key-value store,” ACM SIGOPS operating systems review, vol. 41, no. 6, pp. 205-220, 2007

[5] Ali, A., Qadir, J., Rasool, R.u. et al. Big data for development: applications and techniques. 1, 2 (2016). https://doi.org/10.1186/s41044-016-0002-4

[6] S. Chakraborty, S. Paul and K. M. Azharul Hasan, "Performance Comparison for Data Retrieval from NoSQL and SQL Databases: A Case Study for COVID-19 Genome Sequence Dataset," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021, pp. 324-328, doi: 10.1109/ICREST51555.2021.9331044.

[7] MongoDB. 2021. NoSQL vs SQL Databases. [online] Available at: <https://www.mongodb.com/nosql-explained/nosql-vs-sql> [Accessed 3 December 2021].

[8] MongoDB. 2021. NoSQL vs SQL Databases. [online] Available at: <https://docs.mongodb.com/manual/sharding/> [Accessed 3 December 2021].

[9] Clarence J M Tauro, Aravindh S, Shreeharsha A. B, “Comparative Study of the New Generation, Agile, Scalable, High Performance NOSQL Databases”, International Journal of Computer Applications (0975 – 888) Volume 48– No.20, June 2012 doi:10.5120/7461- 0336

[10] S. Kalid, A. Syed, A. Mohammad and M. N. Halgamuge, "Big-data NoSQL databases: A comparison and analysis of “Big-Table”, “DynamoDB”, and “Cassandra”," 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA), 2017, pp. 89-93, doi: 10.1109/ICBDA.2017.8078782.

[11] Varia, J. and Mathew, S., 2014. Overview of amazon web services. Amazon Web Services, 105.

[12] Amazon Web Services, Inc. 2021. Duolingo Case Study-DynamoDB Case Study. [online] Available at: <https://aws.amazon.com/solutions/case-studies/duolingo-case-study-dynamodb/> [Accessed 4 December 2021].

[13] The Courage of Innovation: A Conversation with Vernā Myers, V., 2021. Netflix Innovator. [online] Amazon Web Services, Inc. Available at: <https://aws.amazon.com/solutions/case-studies/netflix/> [Accessed 4 December 2021].

[14] Burtica, Ruxandra Mocanu, Eleonora Andreica, Mugurel Tapus, Nicolae. (2012). "Practical application and evaluation of no-SQL databases in Cloud Computing". 1-6. 10.1109/SysCon.2012.6189510.

[15] Amazon Web Services, Inc. 2021. Redis: in-memory data store. How it works and why you should use it. [online] Available at: <https://aws.amazon.com/redis/> [Accessed 4 December 2021].

[16] P. P. Srivastava, S. Goyal and A. Kumar, "Analysis of various NoSql database," 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015, pp. 539-544, doi: 10.1109/ICGCIoT.2015.7380523.

[17] 2021. [online] Available at: <https://redis.io/topics/whos-using-redis> [Accessed 4 December 2021].

[18] MongoDB. 2021. Our Customers. [online] Available at: <https://www.mongodb.com/who-uses-mongodb> [Accessed 4 December 2021].

[19] Apache Cassandra. 2021. Apache Cassandra | Apache Cassandra Documentation. [online] Available at: <https://cassandra.apache.org/index.html> [Accessed 5 December 2021].

[20] Apache Cassandra. 2021. Apache Cassandra | Apache Cassandra Documentation. [online] Available at: <https://cassandra.apache.org/cassandra-basics.html> [Accessed 5 December 2021].

[21] Chang, F., Dean, J., Ghemawat, S., Hsieh, W.C., Wallach, D.A., Burrows, M., Chandra, T., Fikes, A. and Gruber, R.E., 2008. Bigtable: A distributed storage system for structured data. ACM Transactions on Computer Systems (TOCS), 26(2), pp.1-26.

[22] Google Cloud. 2021. Overview of Bigtable | Cloud Bigtable Documentation | Google Cloud. [online] Available at: <https://cloud.google.com/bigtable/docs/overview> [Accessed 5 December 2021].

[23] Venkatraman, S., Fahd, K., Kaspi, S. and Venkatraman, R., 2016. SQL versus NoSQL movement with big data analytics. International Journal of Information Technology and Computer Science, 8(12), pp.59-66.

[24] Dist.neo4j.org. 2021. [online] Available at: <http://dist.neo4j.org/neo-technology-introduction.pdf> [Accessed 5 December 2021].

[25] Neo4j Graph Database Platform. 2021. Neo4j Customers. [online] Available at: <https://neo4j.com/customers/> [Accessed 5 December 2021].