CONTEMPORARY APPROACHES TO DATA STORAGE AND PROCESSING

Nikolay Yanev

University of Mining and Geology “St. Ivan Rilski”, 1700 Sofia; nikolay.yanev@mgu.bg

ABSTRACT. The purpose of this article is to present modern approaches to data storage and processing. The NoSQL and NewSQL technologies are considered in which the focus shifts from common solutions (traditional relational database management systems) to individual ones. Attention is also paid to the data format so that they could be integrated into the information systems.

Keywords: information system, database, SQL, NoSQL, NewSQL

СЪВРЕМЕННИ ПОДХОДИ ПРИ СЪХРАНЕНИЕ И ОБРАБОТКА НА ДАННИ

Николай Янев

Минно-геоложки университет “Св. Иван Рилски“, 1700 София

РЕЗЮМЕ. Целта на настоящата статия е да се представят съвременните подходи при съхранение и обработка на данни Разглеждат се технологиите NoSQL и NewSQL при които фокусът се измества от общи решения (традиционни RDBMS) към индивидуални такива. Обръща се внимание и на формата на данните с оглед интегрирането им в информационни системи.

Ключови думи: информационна система, бази данни, SQL, NoSQL, NewSQL

Introduction

Contemporary databases operate with different data models. The aim is to represent the described real objects as accurately as possible. At the same time, the data form should allow their on-line, real-time processing (Fig. 1).

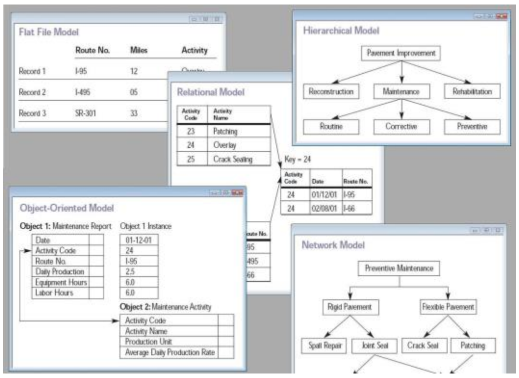

Fig. 1. Database models

In general, the evolution of database management systems (DBMS) can be described in three stages:

- Navigation systems − those were used in the 1960s and represented hierarchical and network models of data description;
- Relational - those were created in the 1970s and are used to this day. They are based on set theory and on relational algebra. The objects are described in the form of two-dimensional tables allowing for connections (relations) between them. They use the SQL programming language;
- Post-relational − this category comprises a wide variety of data description methods. The object-oriented model was introduced in the 1980s, and the NoSQL and the NewSQL models have become popular in the recent decade.

The subject of this article is the final stage, and in particular the nature and application of the NoSQL and NewSQL databases and the major software tools and data formats used with them. While traditional relational database management systems are general-purpose (i.e. they provide uniform solutions to different types of problem), NoSQL and NewSQL solutions are intended for a specific problem, such as short-term Online Transaction Processing (OLTP) operations.

IDC predicts that the total amount of global data will grow from 33 ZB in 2018 to 175 ZB in 2025; the forecast is for an annual growth rate of about 60% (Reinsel, 2018). Cisco's forecast is similar: in 2022, the traffic generated will be higher than the total traffic during the first 32 years since the launch of the Internet (Cisco, 2019).

There are other remarkable stats for the year 2025:

- The storage industry will ship 42 ZB of capacity over the next seven years;
- 90 ZB of data will be created on Internet of Things (IoT) devices by 2025;
- By 2025, 49% of the data will be stored in public cloud environments;
- Nearly 30 percent of the data generated will be consumed in real-time by 2025.

The factors underlying such predictions are as follows:

- The ever-increasing computing power and data storage capacity of modern computers and smart devices;
- Promoting the Internet. The advent of Web 2.0 has enabled the passive user to become active, generating network distributed data;
- The increasingly accessible services offered by contemporary data centres and cloud computing;
- With the advent of Internet of Things (IoT) and Internet of Everything (IoE), it is not just the users who generate information; a huge part of the items we handle on a daily basis, too, begin to generate significant volumes of data that we can employ for various purposes;
- And last but not least, the business needs: industry needs more and more information to manage and analyse in order to maximise business benefits.

Inevitably, these trends result in a revision of classical data management methods. Collecting, storing, analysing, sharing, and visualising data, including unstructured and semi-structured data, has become increasingly difficult with the traditional tools and approaches. Thus, the need to effectively manage large data centres and cloud systems has led to the establishment of new methods and approaches in data modelling.

Mining industry is no exception to these trends. Besides, most of the tasks in the modern mining industry are characterised by a high degree of indefiniteness, non-linearity and multifactoriality (Eftimov et al., 2011) which hinders the application of classical analytical methods (makes them impossible to apply) to solve such tasks.

Approaches for the implementation of database management systems

SQL

The relational model was offered in 1970 by Dr. Edgar Codd. It was introduced massively in the 1980s and 1990s and has been dominant globally to this very day. According to DB-Engines Ranking (2019), four of the top five most popular data management systems are relational.

The relational model is based on set theory and relational algebra, and this rigorous mathematical basis leads to its main advantages: efficiency, simplicity and intuition (Codd, 1970).

The relational database management systems (RDBMS) describe the objects uniformly: through the rows of a two-dimensional table (Fig. 2). Each table contains unordered rows and named columns. The different tables can be linked together.

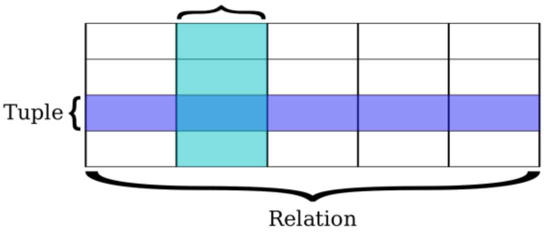

Fig. 2. Major elements in a relational table

The popularity of RDBMS is largely due to the fact that they allow multiple users to work simultaneously without compromising data integrity. This is achieved through strict adherence to ACID (Atomicity, Consistency, Isolation, and Durability) rules that describe the requirements for transactional systems (Gray, 1981):

- Atomicity – This is the ability of a DBMS to implement multiple commands as one. Database transactions must follow the all-or-nothing rule; if one part of the transaction fails, the whole transaction fails;
- Consistency - Each transaction changes the database from one state with consistent information to another such state; before and after the transaction the database must maintain its integrity;
- Isolation - This is a requirement that information in an operation which is being executed and has not been completed is not accessible; if the transactions are executed concurrently, none of them must affect the other. The transactions must be executed in complete isolation as if no other transaction was being performed. Contemporary databases adhere to this rule with some circumvention, through several different types of isolation level, and compromise is allowed in the name of reducing the number of deadlocks (READ UNCOMMITED isolation level);
- Durability − Once verified as successfully completed, a transaction must be retained in the database even in the case of a power outage, or power crash.

The RDBMS use SQL as their basis. Most of the major developers of such systems (Oracle, IBM, Microsoft) have created their own SQL-based languages. Oracle, DB2, SQL Server, MySQL PostgreSQL, Access, etc. are popular RDBMS. Hence, another essential advantage of a RDBMS: SQL is standardised and there is a great deal of overlap among SQL implementations in various databases.

The most significant drawback of the relational model is the inability to scale horizontally due to the use of a relatively static object description scheme. Deterioration of the performance of a RDBMS is also observed with a significant increase in workload and the volume of work data.

NoSQL

Relational databases were not designed to handle the scale (Big Data), flexibility (web applications such as blogs, social networks, etc.) and real-time operation that are required by modern applications. In addition, they do not take full advantage of the low cost of storage devices, nor of the high performance of the machines we have at our disposal nowadays.

NoSQL encompasses a wide variety of database technologies that have been developed in response to the increasing amount of data stored for users, objects and products, the frequency with which this data is accessed, as well as the need of high performance in their processing.

The first NoSQL software appeared in the early 21st century: MongoDB (2009), Redis (2009), Cassandra (2008), etc. Today there is a wide variety of data models used in NoSQL systems. The most popular are shown in Figure 3:

- Key-value: here, information is stored in records of the “key-value” type and complex data structures, including XML, can be stored as “value”. The search is performed via a key. Dynamo, Riak, Azure, Redis, Cache are such NoSQL databases;

- Document: the work data and related information are stored in documents, most often in the XML or JSON formats. This model resembles the key-value model, with the “value” being the document itself. Such models are MongoDB, CouchDB, Raven, BaseX, etc.;
- Wide column stores: again, a "key" is used, but this may point to a family of columns. Each record can have a different number of columns and can be placed in other columns called super columns. BigTable, Hbase, Cassandra, Accumulo are popular examples of column family database software;
- Graph: this works with graph structures. Data is modelled as a network of links between particular elements. Neo4J, Allegro, Virtuoso, Bigdata are such models;
- Multidimensional: Globals, SciDB, Minim DB.

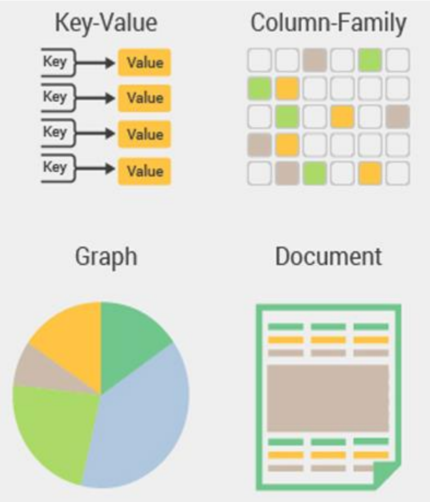

Fig. 3. Popular NoSQL models of data

Among the main advantages of NoSQL databases are: flexibility − they do not work with static schemes; scalability − they also allow for horizontal scaling; facilitated database transfer across multiple servers.

The biggest drawback to NoSQL systems is that they are not transitive.

Typically, NoSQL databases are used in distributed architecture systems, where the focus is on performance with the processing of large amounts of data. In such systems, the CAP Theorem (Brewer's Theorem) is observed (Brewer, 2000): "Up to two of the following categories can be guaranteed in a distributed system:

- Consistency (C): all database clients see the same information, even with competitive updates;
- Availability (A): all database clients can access any version of the information;
- Partition tolerance (P): The database can be partitioned over multiple servers.

The simultaneous provision of all three guarantees is impossible (Fig. 4).

The theorem proves that only two of these three pillars can be used to create such a system. In other words, we can have a system of high consistency and expandability, or a system of high data availability and expandability, or a system of high consistency and high availability but expandability.

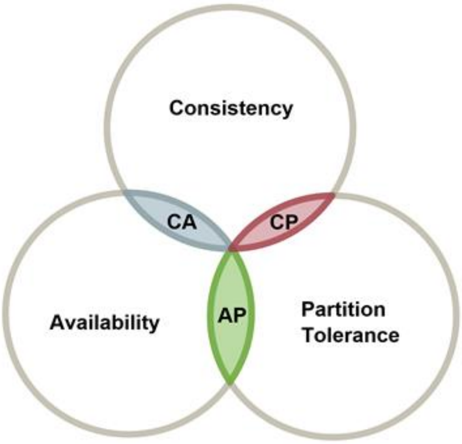

Fig. 4. The CAP theorem

Most NoSQL databases operate on the BASE (Basically Available, Soft-state, Eventual consistency) principle: choosing availability and partitioning at the expense of consistency and looking for the fastest and most reliable synchronisation among individual servers.

Numerous comparative analyses on the performance of RDBMS and NoSQL have been published. As a whole, NoSQL systems perform better when recording, deleting, and updating Bid Data sets.

NoSQL databases have limited application in specific areas; yet, the fact that they are used by IT giants like Google, Facebook, Amazon, and LinkedIn is a proof of their potential.

NewSQL

NewSQL databases have emerged in the past few years. The term NewSQL was proposed by Aslett (2010). These are databases that combine the benefits of SQL and NoSQL databases (Figure 5). NewSQL are horizontally and vertically extensible and transitive.

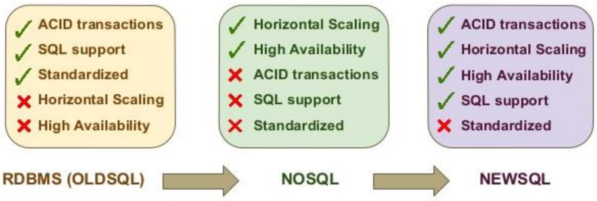

Fig. 5. Comparison between SQL, NoSQL, and NewSQL

The products described as NewSQL databases are very diverse. HANA was created to be a business reporting powerhouse that could also handle a modest transactional workload, a perfect fit for SAP deployments. Hekaton adds sophisticated in-memory processing to the more traditional Microsoft SQL Server. Both systems are non-clustering for now, and both are designed to replace or enhance OldSQL deployments directly. NuoDB set out to be a cluster-first SQL database with a focus on cloud-ops: it runs on many nodes across many data centres and lets the underlying system manage data locality and consistency for you. This comes at a cost in performance and consistency for arbitrary workloads. Other systems, such as MemSQL, focus on clustered analytics. Distributed with MySQL compatibility, MemSQL often offers faster OLAP analytics than all-in-one OldSQL systems, with higher concurrency and the ability to update data as it is being analysed. VoltDB, the most mature of these systems, combines streaming analytics, strong ACID guarantees and native clustering. This allows VoltDB to be the system-of-record for data-intensive applications, while offering an integrated high-throughput, low-latency ingestion engine. It is a great choice for policy enforcement, fraud/anomaly detection, or other fast-decisioning apps (Piekos, 2015).

As a summary of the above, we can classify three major types of NewSQL databases:

- New architectures: databases that were designed to operate in a distributed cluster (Google Spanner, Clustrix, VoltDB, MemSQL);
- SQL engines: highly optimised storage engines for SQL (MySQL Cluster, Infobright, TokuDB);
- Transparent sharding: they provide a sharding middleware layer to automatically split databases across multiple nodes (ScaleBase).

Very often, NewSQL databases are used for partial solutions within the context of RDBMS or NoSQL systems. The ultimate goal of NewSQL is to deliver a high performance, highly available solution to handle modern data, without compromising on data consistency and high-speed transaction capabilities.

Conclusion

Although relational databases are one of the oldest technologies used in the IT industry, they are widely used nowadays. However, with the increase in the volume of data processed, and especially of those distributed in the web environment, some disadvantages of RDBMS have come to the fore that make them inapplicable in modern data storage and analysis systems, particularly when it comes to real-time processing of large arrays of data. Large companies increasingly prefer non-relational approaches when describing such data. If NewSQL databases still offer partial solutions, NoSQL has already established itself in certain areas as a better solution than classic RDBMS.

The trend of increasing the impact of NoSQL databases is also evident from the data on *DB-Engines Ranking* (2019) presented in Figure 6.

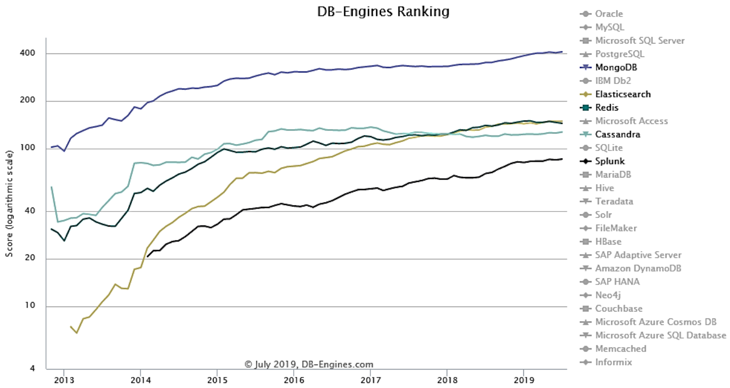

Fig. 6. Graph reflecting the popularity of the NoSQL systems used most often

The software used in the mining industry is characterised by diversity and specifics, both in terms of the type of mineral deposits and of the compliance with the requirements of the particular company (Kutzarov et al, 2012; Anastasova et al., 2016). This is a prerequisite for integrating NoSQL data models in it due to their flexibility.

References

Anastasova, Y., D. Anastasov. 2016. The use of modern information technologies in the education of students from the University of Mining and Geology “St. Ivan Rilski”. − *5th International Scientific and Technical Conference “Technologies and Practices in Underground Mining and Mining”*, 201−204 (in Bulgarian with English abstract).

Aslett, M. 2010. What we talk about when we talk about NewSQL. 451 Group − *https://blogs.the451group.com/information_management/2011/04/06/what-we-talk-about-when-we-talk-about-newsql/*

Brewer, E. A. 2000. Towards robust distributed systems. − *Proceedings of the XIX Annual ACM Symposium on Principles of Distributed Computing. Portland, OR*: ACM. *19*, 7.

Cisco. 2019. *Cisco Visual Networking Index: Forecast and Trends, 2017–2022. White Paper.*

Codd, E. F. 1970. A relational model of data for large shared data banks. − *Communications of the ACM, 13,* 6, 377–387.

DB-Engines Ranking. 2019. *https://db-engines.com/en/ranking.*

Eftimov, Z., D. Anastasov. 2011. Scientific aspects in formation of quality of ore in extraction stage. − *22nd World Mining Congress, I, Istanbul,* 181−185.

Gray, J. 1981. The transaction concept: virtues and limitations. − *Proceedings of the 7th International Conference on Very Large Databases,* 144−154.

Kutzarov, K., D. Anastasov, Z. Eftimov. 2012. Principles for the application of mining software for the planning of mining processes for the extraction of underground natural resources. − *Journal of Mining and Geology*, 2−3, 56−58 (in Bulgarian with English abstract).

Piekos, J. 2015. SQL vs. NoSQL vs. NewSQL: Finding the Right Solution. *https://dataconomy.com/2015/08/sql-vs-nosql-vs-newsql-finding-the-right-solution/.*

Reinsel, D., J. Gantz, J. Rydning. 2018. The Digitisation of the World from Edge to Core. − *IDC White Paper #US44413318, Sponsored by Seagate.*