Годишник на секция "Информатика"
Съюз на учените в България
Том 11, 2021, 1–14

Annual of "Informatics" Section
Union of Scientists in Bulgaria
Volume 11, 2021, 1–14

OVERVIEW OF BIG DATA SYSTEMS

Veselina Bureva

Laboratory of Intelligent Systems, "Prof. Dr. Assen Zlatarov" University,
1 "Prof. Yakimov" Blvd., Burgas 8010, Bulgaria
e-mail: vbureva@btu.bg

Abstract: The big data methodology is described. The purpose of the appearance of big data is investigated. The database overview is made to analyze the evolution of management systems and tool for analyzing. The most used Big Data architectures are presented. The software tools for Big Data analytics are discussed.
Keywords: Big data, Hadoop, MapReduce, NoSQL, NewSQL.

1 Characteristics of Big Data

In the current investigation a brief overview of Big Data systems is presented. Big Data is an interdisciplinary field including distributed computing, databases, computer architecture, system administration, artificial intelligence. Big Data can be considered as field of data science having the aim to collect, store, extract, process and analyze the huge datasets located in different places (data clusters) [6]. Big Data characteristics are determined by presenting the notation for Big Data V's. *Volume* describes massive amount of datasets. *Variety* refers to the multiple formats and types of data and their sources. *Velocity* is related to the frequency of data generating and processing. These three characteristics are fundamental. Their meaning can be easily understood using the Gartner's Vector model [2] presented in Fig. 1.

Additional characteristics are introduced in the literature later. *Veracity* is related to data quality ratability of data and authentication. *Value* is determined as the usefulness of data for an enterprise. *Variability* presented the way of big data changing model, format transformation and integration. *Validity* refers to the ways of cleaning the datasets. *Vulnerability* presents the privacy and security of the data. *Volatility* defines clear rules for data exchange and availability.

Visualization presents graphs or appropriate view for easy understanding of any users [1, 4, 10, 15, 17, 22, 28, 35].

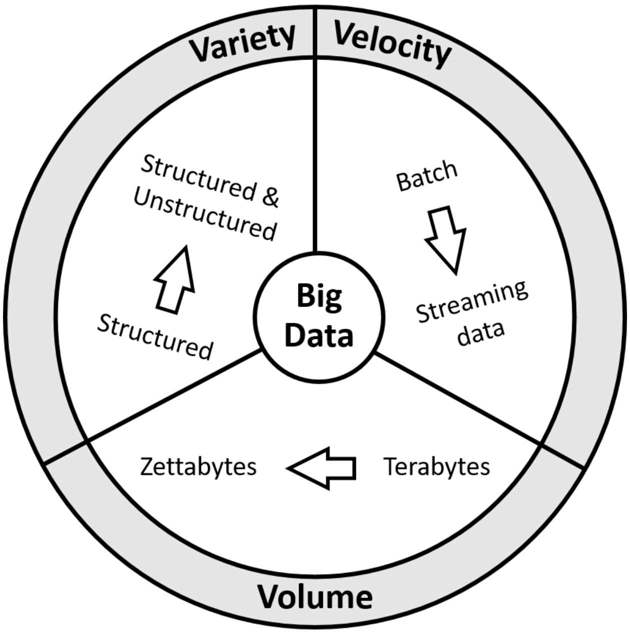

Figure 1. The Gartner's Vector model

Data warehouses were the powerful tool for data management analysis before cloud computing appearance. The data amount increases and includes different formats of data for storing. In addition to structured data large amounts of semi-structured and unstructured datasets are provided. The data storages in the Big Data era are investigated in Section 2. Traditional data warehouses are not enough to solve the problem effectively. The Big Data systems are introduced as a tool for solving the problems and limitations of data warehouses in the new reality of data formats and complex situations. The author has the aim to present the transition from data warehousing to Big Data and need of using the recent technologies of hardware and parallel and distributed computations for its realization. The Big Data solutions include large scale applications on data centers with huge amount of servers. Distributed data management techniques are used for applying fast and complex analyzing tasks. The management of big data application is performed by distributed storage layer. Typically the datasets are separated in the nodes of a shared-nothing cluster [4, 27]. The architectures of Big Data systems are presented in Section 3. Big data Analytics are discussed in Section 4.

2 From data warehouse to Big Data systems: Overview of data models and big data storage systems

The notation Big Data is recently defined having the aim to present the fast processing and analyzing of huge amounts of data frequently provided from different sources. The variety of new data formats that have to be stored and processed in data warehouses make the ETL processes difficult for data analysts. In the last years the volume of the data is incremented. Data warehouses are stored the data using ETL. The requirements for storing and processing huge amounts of data in different formats have to be executed using more powerful tools. The variety of datasets leads to complete different approaches for storing the data. Big Data systems have the capabilities to store structured, semi-structured and unstructured datasets. The task of physical data integration, known as a step from data warehousing, is complex and difficult to be ensured in big data technologies. The difference in big data storages according to data warehouses is the way of storing the data. The datasets are saved in their native format, i.e. the storage can be schema-on-read. The big data requirements for data storage leads to the definition of new type of data warehouse that store datasets in its native format. The storage for different type of data files is introduced as "data lake". The big data and data lake technologies are supported by the notation of NoSQL databases. In the next sections several types of databases used in big data application will be discussed. The NoSQL and NewSQL databases are introduced as part of big data technologies for storing the input information [27].

2.1 Relational databases: The inheritance of data warehouses

The growth of the technology innovations provides technical solutions supporting the data storages and data analyzing engines. Transactional databases are well known solutions for numerous business applications in the past. Flexible aggregations provide to the users useful knowledge from their historical datasets. The complex queries can be executed to present user defined reports. The relational databases require the normalized datasets for data storage. The information is stored in tables with the appropriate primary keys and relationships. The standard ACID transactions (Atomic, Consistent, Independent, and Durable) are supported. The database schema represents the datasets organization. Relational databases are the main data source for data warehousing. In the recent years the information grows and the type of the data sources change. The standard data warehouses have to be modernized to provide flexible solutions for different types of datasets. Transactional systems can be used in Big Data applications but not cover all the data types existing in the Big Data analyses. They are not enough to provide the variety of Big Data. The constraints of relational model are established firstly by *Google*. Then they introduce the GFS (Google File System) and a new hardware and software platform.

2.2 Big Data Systems

The cloud computing appearance presents new direction for data storage development tools. The datasets are not only data warehouse based - the web based systems provides new type of information that have to be processed. The area of cloud applications includes webmails, online files storage, social networking sites and online business applications. The cloud computing model needs of available network connection to provide to the users access to information and computer resources in each time moment at any place [15]. The organizations begin the investigations on social networks to achieve useful data for user behavior. The development of the new technologies related to generating huge amounts of different type of data and fast processing requires the stable data storage engines. The parallel and distributed systems are effectively involved in the investigation processes. The first Big Data innovation is considered the *PageRank* algorithm which is introduced in 1996 by *Google*. The concept uses the search engines for indexing on keywords within web pages. In 2005 Google change the hardware architecture of the data center to the *Google Modular Data Center*. *Google* focuses to the massively parallelizing and distributing processing across very large numbers of commodity servers. Thereafter, they introduce *Google Modular Data Center* and *Google Software Stack* in 2006. The Google software platform contains three main layers: Google File System (GFS), MapReduce and BigTable. GFS is a distributed cluster file system [14] that provides the capabilities for accessing the Google data center disks [12]. The architecture of GFS is presented in the Fig. 2 [15]. GFS is constructed from a master server and multiple chunk servers. Files are separated in several chunks with fixed size. Chunk servers store chunks on local disks. Each chunk is replicated on multiple chunk servers. The master contains all file system metadata [12, 14].

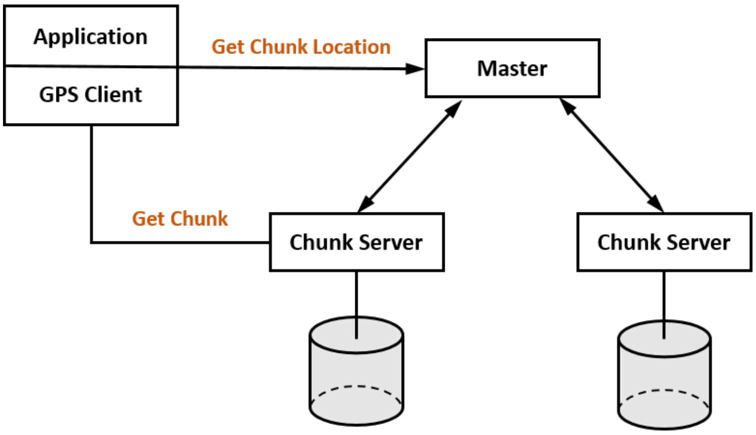

Figure 2. Google File System Architecture

MapReduce is investigated in 2004 as distributed processing framework for huge amounts of data [9, 35]. MapReduce paradigm is presented in Fig. 3. The programming model consist two main user-defined functions *map*() and *reduce*().The information is stored in files using key-value format. The *map*() function navigate through each record to calculate the intermediate ⟨key, value⟩ pairs. Thereafter the *reduce*() function computes a combined result by summarizing all the values that have the same unique key. Depending on the used platform several modifications of the MapReduce can be investigated. BigTable is presented in 2006 as database system of *Google* [7]. These three layers provide the basics for Big Data systems. The most popular big data architectures are presented in Section 3.

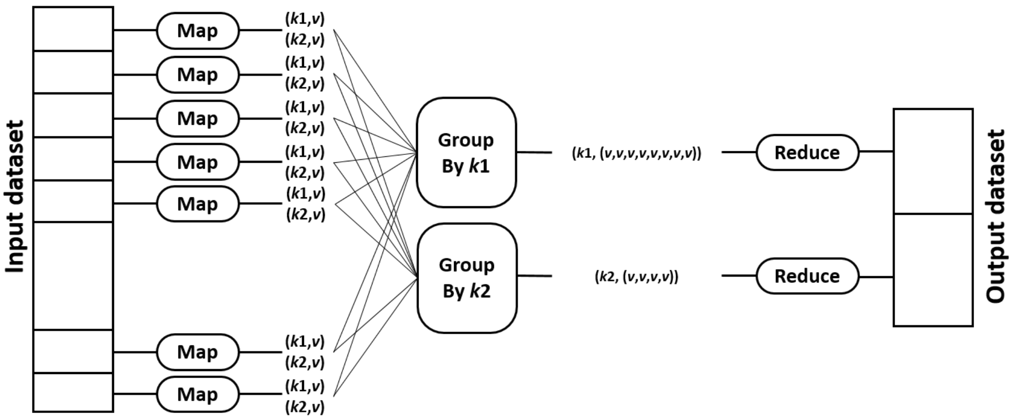

Figure 3. MapReduce execution process

2.3 Big Data adaptation of NoSQL databases

The data warehouses are excellent decision for storing and manipulating structured data but are not stable to process complex data, huge data, semi-structured or unstructured data as web logs, images, video, and etc. Nowadays the datasets are collected by the social networks (*Facebook*, *LinkedIn*), *Google* and *Amazon* user status data, mobile data, data streams, sensors data (Internet of Things gadgets, Smart Home measurements), e-mails. Therefore the type of the input information is different from the standard format of transactional stores. The challenge of Big Data is to store and analyze different types of datasets for optimal time. The need of Big Data methodologies are related to the appearance of the NoSQL databases. NoSql databases are systems that are non-relational, highly scalable, having dynamic schemas, and providing fast processing of large volumes of data [6]. The document-oriented storages, column-oriented storages, key-value storages and graph based storage provide datasets in different formats.

The introduction of NoSQL databases is considered as the third revolution of the database systems after the second step of the revolution based on the transactional systems as relational model and object-relational model, and the first revolution of the databases investigating the hierarchical model and the network model. NoSQL databases make attention on the presence or absence of the SQL language – the most non-relational systems do not support SQL. NoSQL Databases do not require schema. The support of nested data implementation and retrieval is provided. The datasets are automatically divided to the servers. NoSQL databases are used in web space allowing sharing and replication. NoSQL database do not use the ACID (atomicity, consistency, isolation and durability) transaction guarantees to achieve performance [20]. NoSQL databases satisfy the CAP theorem introduced in 2000 by Dr. Eric Brewer and proved by Gilbert and Lynch in 2002 (Fig. 4).

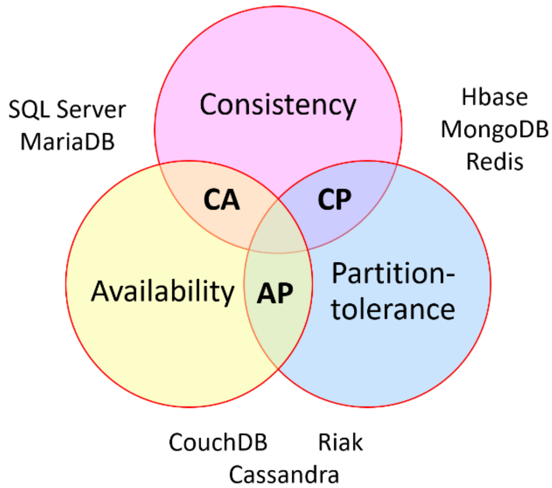

Figure 4. CAP theorem

The CAP theorem includes the important properties of distributed systems known as Consistency, Availability and Partition-tolerance. Consistency presents the same identical view of data for each user. Availability is applied in the time when it is determined a failure but the database continues to work. Partition tolerance provides a way for supporting operations between two segments of the distributed system in the time when the network is corrupted [5, 13, 14]. The NoSQL databases can use the BASE model (Basically Available, Soft state, Eventual consistency). NoSQL databases storage management usually use memory (RAM), solid-state drives (SSDs) and spinning disk drives (HDDs) for performance purposes [11]. NoSQL databases are classified in several groups: document databases, column-oriented databases, graph databases, key-value databases [33]. Brief notation of each type is presented below:

- ***Document databases*** use JSON, BSON, or XML documents. The information is stored in separate documents with corresponding keys. The relations between two documents can be made by document embedding or linking by document identifiers. The querying is faster than relational databases. The lack of a fixed schema before inserting the data. Examples of document databases are *CouchDB* and *MongoDB* [14].
- ***Graph databases*** adopt the notations from graph theory. They are focused on the relationship between data elements. Nodes represent entities. Links or relationships are used to present the connections between elements. Both nodes and relationships have properties. Languages as SPARQL and Gremlin are used in graph databases. Example for graph database is Neo4J [14, 23].
- ***Column-oriented databases*** – the information is saved in columnar format despite of rows. The columns forms blocks. Columnar databases store the datasets in massively distributed architectures. They can quickly aggregate the value of selected column that makes it perfect for data analytics and data mining. *Apache Parquet* and *Google*'s *BigTable* are examples for columnar databases [23].
- ***Key-value databases*** – the information is presented as key-value pair. The key has corresponding value or group of values. It is based on hash table. The keys have to be unique. Examples of key-value databases are *Redis*, *Riak* and *Amazon DynamoDB* [14].

2.4 NewSQL databases: The agreement between the relational databases and NoSQL databases

The term NewSQL was coined in 2011 by Matthew Aslett in his business analysis report discussing the rise of new database systems [3]. The term NewSQL is presented as new type of databases that try to achieve the same scalability of NoSQL databases but still keep the relational model (with SQL) and transaction support. NewSQL databases are relational systems that combine the OLTP, high performance and scalability of NoSQL. The main NewSQL systems contain new architectures, transparent sharding middleware, SQL engines and Database-as-a-Service (DBaaS). Sharding (partitioning) of the NewSQL databases scale out by dividing the database into separate subsets known as partitions or shards. The tables are horizontally spitted based on the column values. Related parts from different tables are joined to create shard. Secondary indexes are used. NewSQL database use concurrency control to ensure simultaneous transactions. Crash Recovery is included to provide recovering the data in the case of crashes [19, 34].

2.5 In-memory databases

The information in traditional databases is stored on disk. In-memory databases (IMDB) store the datasets entirely in main memory. In-memory databases can use RAM disk or SSD drive. Obviously, the advantage of in-memory databases is that they work faster than databases with disk storage. The complete explanation of the in-memory databases is presented in [18, 19]. In-memory databases can be classified depending on the used data model: in-memory relational database, in-memory column-based storage, in-memory online transaction processing (in-memory OLTP), in-memory NoSQL databases. The system *TimesTen* of *Oracle* is an in-memory relational system. *HANA* database of *SAP* provides column-based storage loaded into memory. The OLTP database *Starcounter* provides own object oriented data manipulation language based around NewSQL [18]. *VoltDB* data platform is known as a NewSQL in-memory database. *Apache Ignite* is an example for distributed in-memory database. In addition to In-Memory Databases (IMDB) the In-Memory Data Grids (IMDG) is frequently used term in the literature for describing the technologies applied in real-time computing and big data applications. While IMDB stores and retrieves data placed in a computer's main memory the IMDG is optimized for analytical processing of large amount of data.

3 Big Data Architectures

Big Data architectures are introduced to support Big Data systems. Depending on the area of application and type of the input datasets different architectures can be applied. Here, the most popular architectures for big data systems are discussed. There are Lambda Architecture, Kappa Architecture, Microservice-based Architecture, Zeta Architecture [16].

- *Lambda Architecture* **–** Lambda architecture is proposed by Nathan Marz in 2011. It supports the batch and real time processing. Lambda Architecture is fault-tolerant, scalable and generic data processing architecture. The dataflow has two paths. The architecture is constructed using three layers: batch layer (cold path), speed layer (hot path) and serving layer. Batch layer is presented by distributed file system which stores the historical data. It is used for storing and computing. Speed layer analyze the datasets in real-time. It executes small batch-processes on data according to the selected time window. The serving layer merges the outputs from the batch and speed processing layer to provide fast interactive analyses by users. Popular platforms using lambda architecture are *Twitter*, *Spotify* [16, 21, 32]. Fig. 5 presents Lambda architecture.
- ***Kappa architecture* –** The kappa architecture was coined by Jay Kreps in 2014 as simplification of lambda architecture. Only one path is used – stream processing system. It contains two layers stream processing layer and serving layer. Kappa architecture performs data stream processing, real-time processing, or processing of live discrete events. Examples are IoT events, social networks, log files or transaction processing systems. Popular platforms using kappa architecture are *LinkedIn* and *Yahoo* [16, 32]. Fig. 6 presents Kappa architecture.

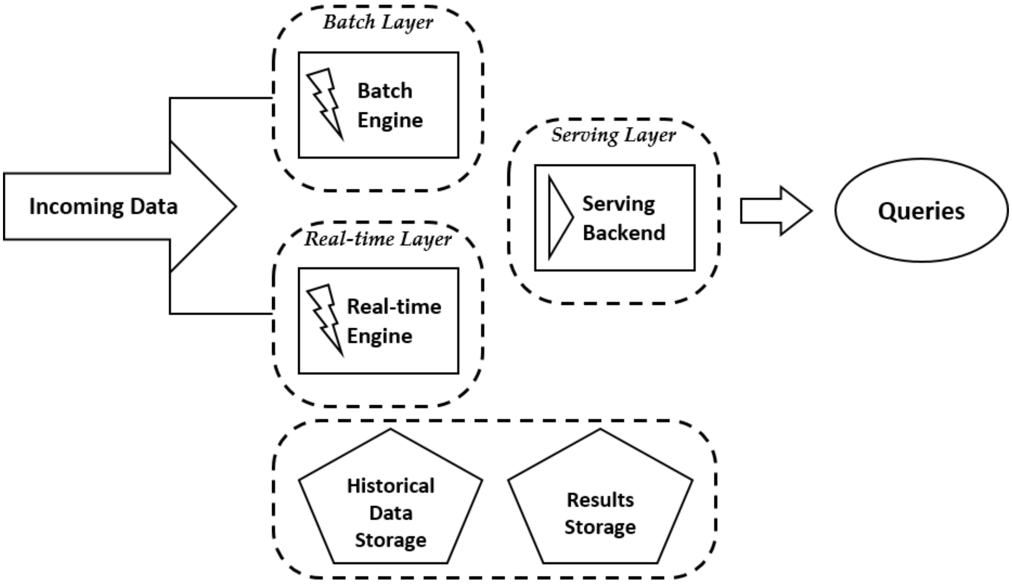

Figure 5. Lambda Architecture

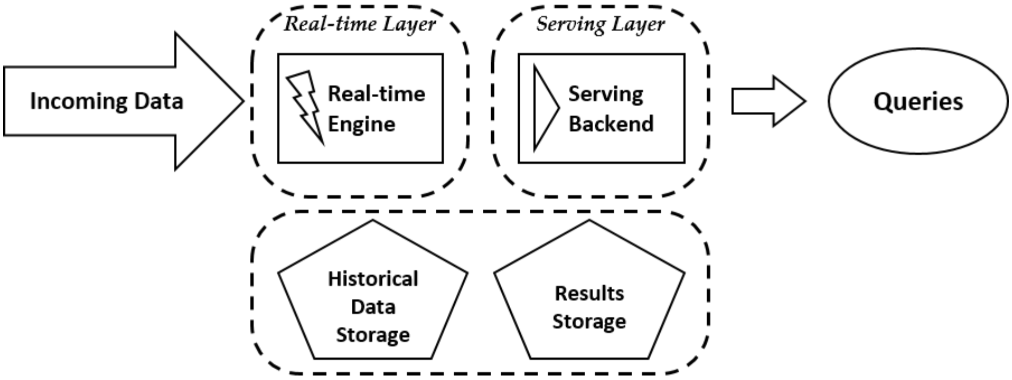

Figure 6. Kappa Architecture

- ***Zeta architecture* –** Zeta architecture is introduced by Jim Scott in 2015. It is architecture that allows scalable integration of business data. Zeta architecture contains seven components: Distributed file system, Real-time data storage, Pluggable execution engine, Deployment management system, Solution architecture, Enterprise applications and Dynamic and global resource management [16, 32].
- ***Microservices-based architecture* –** the microservices-based architecture is known approach for implementing software applications. It divides big data system into

many services called microservices that can run independently. All the elements of big data architecture are treated as services, deployable on a cluster. Different services could use different technology stacks. A popular platform using the microservices-based architecture is *Amazon* [16, 32].

4 Big Data Analytics

Big Data analytics is a process exploring data to extract hidden patterns and previously unknown trends. The received knowledge helps the process of decision making in given enterprise. There are three different types of big data analytics: descriptive, predictive and prescriptive (Fig. 7).

- ***Descriptive Analytics –*** it present an overview of events in the past by analyzing historical data. The descriptive analytics present the unreadable for the user data in appropriate format. This branch of data analytics adopts some simple statistical methods used in business intelligence as regression and correlation [31].
- ***Predictive Analytics –*** the field contain methods predicting future trends and events based on the huge data from different sources. Predictive analytics include all the processes of prediction, forecasting and scoring. Here, the user can perform predictions in different scientific areas [31].
- ***Prescriptive Analytics –*** the prescriptive analytics help the scientist to evaluate the impact of different possible decisions. The techniques for prescriptive analytics include optimization, numerical modeling, and simulation [31].

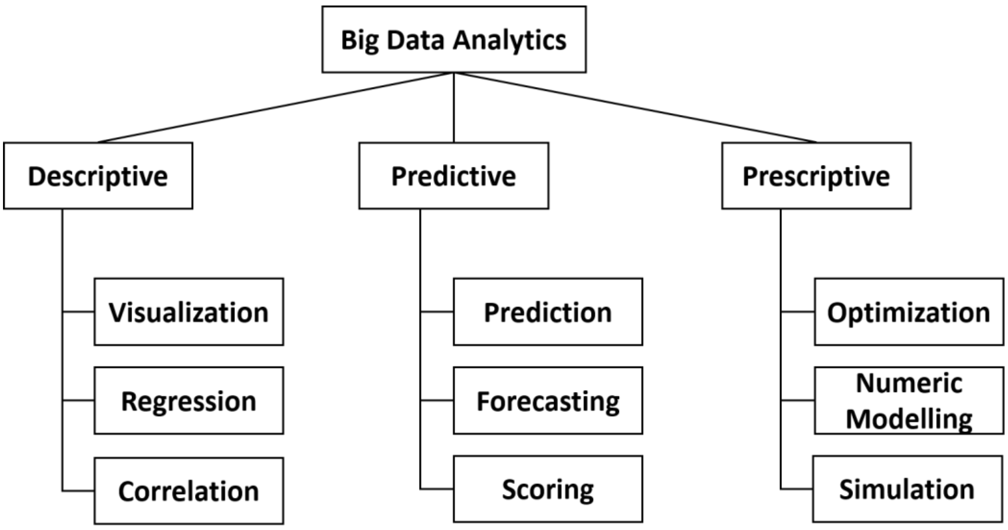

Figure 7. Big Data Analytics

Nowadays, there are many big data platforms. Apache Hadoop and Apache Spark can be considered as the beginners of open-source big data platforms. Apache Hadoop is the most popular MapReduce platform. It is introduced as open source distribution by Yahoo. Hadoop Distributed File System (HDFS) and MapReduce are two main building blocks of Hadoop. HDFS use master-slave architecture. The master server that stores metadata is known as Name node (NN). The separate machines are introduced as data nodes (DN). The clients communicate with Name node to store files. It splits the files into chunks and stores them into data nodes. The big data processes and MapReduce programming model are investigated in [24, 25] and are given intuitionistic fuzzy interpretation in [8]. *Apache Spark* is developed in UC Berkeley. Spark works using master/slave architecture. It works with batch and streaming data. The default programming language is Scala, but Python, R or Java can be additionally applied. Spark introduces RDDs (resilient distributed datasets) that implements the notation for read-only collections of objects partitioned across a set of machines [31]. The development of distributed big data platforms continue by introducing of *Apache Storm*, *Apache Flume*, *Apache Kafka*, *Apache Flink*, *Cloudera*, *Hortonworks Data Platform*, etc., depending of the purpose of the investigated solutions. Applications of big data technologies and data science processes are described in [26, 29, 30].

5 Conclusion

A brief Big Data systems overview is presented. The characteristics of Big Data are discussed. Big Data Storage engines are investigated. The main big data architectures and big data analytics are presented.

Acknowledgements

The authors are thankful for the support provided by the European Regional Development Fund and the Operational Program "Science and Education for Smart Growth" under contract UNITe No. BG05M2OP001-1.001-0004-C01 (2018-2023).

References

[1] Abu-Salih B., Wongthongtham P., Zhu D., Chan K.Y., Rudra A., Introduction to Big Data Technology. In: *Social Big Data Analytics*. Springer, Singapore, 2021.

[2] Ajah I., Nweke H., *Big Data and Business Analytics: Trends, Platforms, Success Factors and Applications, Big Data and Cognitive Computing*, 2019.

[3] Aslett, M., "What we talk about when we talk about NewSQL", 451 Group, Retrieved August 1, 2021.

[4] Bellatreche L., Chakravarthy S., *A special issue in extending data warehouses to big data analytics, Distributed and Parallel Databases*, 37, 2019, pp. 323–327 https://doi.org/10.1007/s10619-019-07262-1

[5] Brewer E., Towards robust distributed systems, *PODC '00: Proceedings of the nineteenth annual ACM Symposium on Principles of Distributed Computing*, 2000.

[6] Castrounis A., Data Science and Big Data, Explained, InnoArchiTech, Accessed: 2.08.2021, https://www.kdnuggets.com/2016/11/big-data-data-science-explained.html

[7] Chang F., Dean J., Ghemawat S., Hsieh W., Wallach D., Burrows M., Chandra T., Fikes A., Gruber R., Bigtable: A Distributed Storage System for Structured Data, *OSDI'06: 6th Symposium on Operating Systems Design and Implementation*, USENIX Association, 2006, pp. 205–218.

[8] Chountas, P., Atanassov, K., Atanassova, V., Sotirova, E., Sotirov, S., Roeva, O., Big data, intuitionistic fuzzy sets and MapReduce operators. Notes on Intuitionistic Fuzzy Sets, Vol. 24, 2018, No. 2, 129–135.

[9] Dean J., Ghemawat S., MapReduce: Simplified Data Processing on Large Clusters, *OSDI'04: 6th Symposium on Operating Systems Design and Implementation*, USENIX Association, 2004, pp. 137–149.

[10] Erl T., Khattak W., Buhler P., *Big Data Fundamentals: Concepts, Drivers & Techniques*, Arcitura Education, 2016.

[11] Gessert, F., Wingerath, W., Friedrich, S. et al. NoSQL database systems: a survey and decision guidance. *Computer Science Research and Development*, vol. 32, 2017, pp. 353–365.

[12] Ghemawat S., Gobioff H., Leung S., The Google File System, Google, *ACM SIGOPS Operating Systems Review*, Volume 37, Issue 5, 2003, pp. 29–43.

[13] Gilbert S., Lynch N., Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. *SigAct News*, June 2002.

[14] Harrison G., *Next Generation Databases: NoSQL, NewSQL, and Big Data*, Apress, 2015.

[15] Hassanien A., Azar A., Snasel V., Kacprzyk J., Abawajy J., *Big Data in Complex Systems: Challenges and Opportunities*, Springer Switzerland, 2015.

[16] Kalipe G., Behera R., Big Data Architectures: A Detailed and Application Oriented Analysis, *International Journal of Innovative Technology and Exploring Engineering* (IJITEE), Vol. 8, Issue 9, 2019.

[17] Khan N., Shah H., Badsha G., Salehian S., Alsaqer M., Abbasi A., The 10 Vs, Issues and Challenges of Big Data, *ICBDE '18: Proceedings of the 2018 International Conference on Big Data and Education*, 2018, pp. 52–56.

[18] Lake P., Crowther P., *Concise Guide to Databases: A Practical Introduction*, Springer, 2013.

[19] Levandoski, J. J., Larson, P., Stoica, R. "Identifying hot and cold data in main-memory databases," *2013 IEEE 29th International Conference on Data Engineering (ICDE)*, 2013, pp. 26-37, doi: 10.1109/ICDE.2013.6544811.

[20] Madison M., Barnhill M., Napier C., Godin J., NoSQL Database Technologies, *Journal of International Technology and Information Management*, Vol. 24, Iss. 1, 2015, Article 1.

[21] Marz N., warren J., *Big Data: Principles and Best Practices of Scalable Real-Time Data Systems*, Manning Publications, 2015.

[22] Mohanty H., Bhuyan P., Chenthati D., *Big Data: A Primer*, Springer India, 2015.

[23] Nayak A., Poriya A., Poojary D., Type of NOSQL Databases and its Comparison with Relational Databases, International Journal of Applied Information Systems, *Foundation of Computer Science FCS,* New York, USA, Vol. 5, No. 4, 2013.

[24] Orozova, D., Atanassov, K. Model of Big Data MapReduce processing. *Comptes rendus de l'Academie bulgare des Sciences*, 72(11), 2019, pp. 1537–1545.

[25] Orozova, D., Atanassov, K. Generalized net model of processes related to Big Data. *Comptes rendus de l'Academie bulgare des Sciences*, 71(12), 2018, pp. 1679–1686.

[26] Orozova D., Popchev, I. Cyber-Physical-Social Systems for Big Data, *XXI International Symposium on Electrical Apparatus and Technologies SIELA 2020*, 3-6 June 2020, Bourgas, Bulgaria, pp. 334–337, ISBN 978-1-7281-4345-3.

[27] Özsu M., Valduriez P., *Principles of Distributed Database Systems*, Fourth Edition, Springer, 2020.

[28] Panimalar A., Shree V., Kathrine V., The 17 V's Of Big Data, *International Research Journal of Engineering and Technology*, Vol. 4, Issue 9, 2017, pp. 329–333.

[29] Popchev I., D. Orozova, Data Science: Experience and Trends, ICTTE 2020, International Conference on Technics, Technologies and Education, Journal IOP Conference Series: Materials Science and Engineering, Volume 1031 (2021) 012057, IOP Publishing, DOI: 10.1088/1757-899X/1031/1/012057.

[30] Popchev I., D. Orozova. Towards Big Data Analytics in the E-learning Space Cybernetics and Information Technologies, Vol. 19(3), 2019, pp. 16–25.

[31] Pyne S., Rao B., Rao S., *Big Data Analytics Methods and Applications*, Springer, 2016.

[32] Singh K., Behera R., Mantri J., Big Data Ecosystem: Review on Architectural Evolution, IN: A. Abraham et al. (eds.), *Emerging Technologies in Data Mining and Information Security, Advances in Intelligent Systems and Computing* 813, Springer, 2019, pp. 335–345.

[33] Sullivan D., *NoSQL for Mere Mortals*, Pearson Education, 2015.

[34] Valduriez P., Jimenez-Peris R., Özsu M., Distributed Database Systems: The Case for NewSQL, Transactions on Large-Scale Data- and Knowledge-Centered Systems, Springer Berlin /Heidelberg, 2021, *Lecture Notes in Computer Science*, 12670, pp. 1–15.

[35] Zomaya A., Sakr S., *Handbook of Big Data Technologies*, Springer, 2017.