# Data Lake, Spark and Hive

Subjects: Computer Science, Theory & Methods

Contributor: Marwa Salah Farhan , Amira Youssef , Laila Abdelhamid

Big data are a large number of datasets that are difficult to store and process using existing database management tools. Big data have some characteristics, denoted by 5Vs: volume, velocity, veracity, variety, and value. Volume refers to the size of the data, velocity refers to the speed of the data from the sources to the destination (data flow), variety refers to different format types of the data, veracity refers to the quality of the data, and value refers to the importance of the data collected without analysis and insight. Lastly, the characteristics have become more than ten, like volatility and visualization value.

big data unstructured data warehouse ETL

## 1. Introduction

Every online interaction, social media post, financial transaction, sensor reading, and digital communication generates data. The proliferation of digital technologies, the widespread use of the internet, and the advent of connected devices have contributed to this massive growth in data. Furthermore, organizations accumulate vast amounts of data in the form of customer records, sales transactions, operational logs, and so on. As a result, an unprecedented amount of data is available for processing and analysis. A data warehouse is a central repository that deals with highly structured, cleansed, processed, stored, and integrated data from a variety of sources to give business intelligence users and decision-makers a single view [1]. These data are processed by an Extract–Transform–Load (ETL) process. The are two types of processes for extracting data from various sources: full extraction and incremental extraction. The data are then transformed through actions such as joining, converting, filtering, cleaning, aggregation, and so on. Finally, these transformed data are loaded into a data warehouse [2][3]. Full extraction is employed when replicating data from a source for the first time or when some sources cannot identify changed data, necessitating a complete reload for the entire table. Incremental extraction is utilized when some data sources cannot provide notifications about updates but can identify modified records and extract them [4]. Cleaning is essential for data warehouses before data are stored; for example, erroneous or misleading information will result from duplicated, inaccurate, or missing data. Data cleaning is regarded as one of the most difficult tasks in data warehousing throughout the ETL process due to the vast variety of possible data discrepancies and the enormous amount of data [5]. Recently, there has been a growing interest in the ELT approach, which prioritizes loading data into a data warehouse before performing transformations. This approach gains speed by delaying the transformation until it is necessary. This ELT process is becoming popular where business requirements are rapidly changing. ‘EL’ essentially implies data replication in numerous real-world scenarios, and the problem is to accomplish it efficiently and with high accuracy. ELT has grown in popularity owing to a variety of causes. Data are being created in ever-increasing quantities, frequently without human intervention.

Storage is becoming more affordable, whether on-premises or in the cloud. With the proliferation of open-source technologies (e.g., Apache Spark, Apache Hadoop, and Apache Hive) and cloud solutions (e.g., Microsoft Azure, Google Cloud, and AWS), the cloud provides low-cost solutions for analyzing disparate and dispersed data sources in an integrated environment [6]. Based on the sources of the internet, the growth of data has increased incredibly with different types of structured, semi-structured, and unstructured data, and that gives an idea of how much the volume of data has increased [7][8]. Structured data are information that is well ordered and stored in a relational database or a spreadsheet. Semi-structured data are data that have not been recorded in standard ways. Nevertheless, the data are not entirely unstructured; examples include metadata and emails. Text, photos, and videos are examples of unstructured data. Text data have garnered special attention among various forms of unstructured data, as they stand out as the most suitable technique for describing and conveying information [9]. These data are distinguished by their complexity, variety, volume, and application specificity and are generally referred to as big data.

## 2. Data Lake

A data lake is a headquartered repository that keeps massive amounts of raw, unprocessed, and diversified data in its natural format [10][11]. It is intended to hold structured, semi-structured, and unstructured data, offering an expandable and affordable data storage and analysis solution. Data are gathered in a data lake through a variety of sources, such as databases, log files, social media feeds, and sensors. Data lakes, unlike typical data storage platforms, do not impose a fixed structure or schema on data at the moment of input. Instead, data are saved in their raw form, keeping their natural structure and inherent flexibility. Because the structure and purpose of the data may be specified later within the analysis phase, this strategy allows businesses to gather huge volumes of data without the requirement for prior data modeling. So, on the other hand, it is seen as the next stage in displacing data warehouses as an enhanced present approach to raw analytics information storage [12][13]. While a data lake has tremendous benefits, it also has certain drawbacks. Because data lakes hold raw and unprocessed data, they are exposed to data quality, security, and privacy challenges. Without effective governance and data management techniques, a data lake may quickly devolve into a data swamp. A data swamp is a data lake that has become bloated with inconsistent, incomplete, erroneous, and ungoverned data. It is frequently caused by a lack of processes and standards that are not effectively regulated. As a result, data in a data swamp are difficult to locate, process, and analyze. Users may need to invest substantial time and effort in data searches and understanding the data's context when there is no defined data model or schema [14]. Given the advantages and disadvantages of both data warehouses and data lakes, a recent approach has emerged, known as a data lakehouse.

A data lakehouse is a combination of both a data warehouse and a data lake. A data lakehouse is a single and integrated platform that combines a data lake's scalability and flexibility with a data warehouse's structured querying and performance improvements. It provides enterprises with a unified platform for organized, semi-structured, and unstructured data. It removes the need for separate storage systems and enables users to effortlessly access and analyze various kinds of data. A data lakehouse allows for schema evolution. It supports schema-on-read, allowing users to apply schemas and structures while querying data. In addition, cloud-based

storage and computation resources are used in a data lakehouse, allowing enterprises to expand resources as needed and employ sophisticated query engines, such as Apache Spark or Presto, to analyze enormous amounts of data quickly and efficiently [15][16].

## 3. Spark and Hive

Spark is a powerful distributed processing system that provides a simple tool for analyzing heterogeneous data from various sources. It supports batch processing, real-time processing, and near-real-time processing (DStream). Spark can be deployed as a stand-alone cluster (if associated with a capable storage layer) or as an alternative to the MapReduce system by connecting to Hadoop. Spark uses a model called Resilient Distributed Datasets (RDDs) to implement batch calculations in memory, which allows it to maintain fault tolerance without having to write to disk after each operation [17]. As a result, the buffer memory enables it to process a large volume of incoming data, increasing overall throughput, and thus, in-memory processing contributes significantly to speed. Batch processing in Spark offers incredible advantages in terms of speed and memory consumption. Spark, which stores intermediate results in memory, is only influenced by the HDFS configuration when reading the initial input and writing the final output [18]. In ELT, new data sources can be easily added to the model. Consequently, various transformations may be applied to the data as needs vary. When raw data are loaded, numerous transformations can be implemented based on changes in requirements [19]. There are big data processing technologies like Map-Reduce, Storm, Kafka, Sqoop, and Flink; the best technology for parallelism is Spark. Spark Core serves as the foundational execution engine for the Spark platform, serving as the base for all other functionalities. It offers capabilities for working with Resilient Distributed Datasets (RDDs) and performing in-memory computing tasks. PySpark serves as a Python interface for Apache Spark, enabling the development of Spark applications and the analysis of data within a distributed environment and allowing users to write data from Spark DataFrame or RDDs to Hive tables [20].

Hive is a data warehousing infrastructure tool based on the Hadoop Distributed File System (HDFS) [21][22] used for analyzing, managing, and querying large amounts of data distributed on the HDFS. Reading and writing data are supported by Hive. Hive is mainly used for structured data, but for this research, researchers can load text data using SerDe, which stands for “Serializer and Deserializer”. When an object is transformed into a binary format for writing to permanent storage, such as the HDFS, this process is referred to as serialization, while the process of converting binary data back into objects is known as deserialization. Tables are turned into row elements in Hive, and then row objects are put onto the HDFS using a built-in Hive serializer. These row objects are then transformed back into tables using a built-in Hive Deserializer. Hive is allowed to integrate with other data processing tools. For example, the HCatalog SerDe allows reading and writing Hive tables via Spark.

## References

1. Dhaouadi, A.; Bousselmi, K.; Mohsen, G.; Monnet, S.; Hammoudi, S. Data Warehousing Process Modeling from Classical Approaches to New Trends: Main Features and Comparisons. Data 2022, 7, 113.
2. Santoso, L. Yulia Data Warehouse with Big Data Technology for Higher Education. Procedia Comput. Sci. 2017, 124, 93–99.
3. Alqarni, A.; Pardede, E. Integration of Data Warehouse and Unstructured Business Documents. In Proceedings of the 15th International Conference on Network-Based Information Systems, Melbourne, VIC, Australia, 26–28 September 2012; ISBN 1467323314.
4. Diaz-Chito, K.; Ferri, F.J.; Hernández-Sabaté, A. An Overview of Incremental Feature Extraction Methods Based on Linear Subspaces. Knowl. Based Syst. 2018, 145, 219–235.
5. Rahm, E.; Do, H.H. Data Cleaning: Problems and Current Approaches. IEEE Data Eng. Bull. 2000, 23, 3–13.
6. Simitsis, A.; Skiadopoulos, S.; Vassiliadis, P. The History, Present, and Future of ETL Technology. Invited Talk. 2023. Available online: https://dblp.org/rec/conf/dolap/SimitsisSV23.html (accessed on 25 January 2024).
7. Bose, S.; Dey, S.K.; Bhattacharjee, S. Big Data, Data Analytics and Artificial Intelligence in Accounting: An Overview. In Handbook of Big Data Research Methods: 0; Edward Elgar: Northampton, MA, USA, 2023; p. 32.
8. Ernst & Young. Changing the Way Businesses Compete and Operate. Insights on Governance, Risk and Compliance, EY Building a Better Working World. 2014. Available online: https://dl.icdst.org/pdfs/files2/8e7f03e2a5c148145615328ec03b2e33.pdf (accessed on 25 January 2024).
9. Bochkay, K.; Brown, S.V.; Leone, A.J.; Tucker, J.W. Textual Analysis in Accounting: What's Next? Contemp. Account. Res. 2023, 40, 765–805.
10. El Aissi, M.E.M.; Benjelloun, S.; Loukili, Y.; Lakhrissi, Y.; Boushaki, A.E.; Chougrad, H.; Elhaj Ben Ali, S. Data Lake Versus Data Warehouse Architecture: A Comparative Study. In Proceedings of the 6th International Conference on Wireless Technologies, Embedded and Intelligent Systems, WITS 2020, Fez, Morocco, 14–16 October 2020; Volume 745, pp. 201–210.
11. Liu, R.; Isah, H.; Zulkernine, F. A Big Data Lake for Multilevel Streaming Analytics. arXiv 2020, arXiv:2009.12415.
12. Oreščanin, D.; Hlupić, T. Data Lakehouse—A Novel Step in Analytics Architecture. In Proceedings of the 2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 27 September–1 October 2021; pp. 1242–1246.

13. Nambiar, A.; Mundra, D. An Overview of Data Warehouse and Data Lake in Modern Enterprise Data Management. Big Data Cogn. Comput. 2022, 6, 132.
14. Azeroual, O.; Schöpfel, J.; Ivanovic, D.; Nikiforova, A. Combining Data Lake and Data Wrangling for Ensuring Data Quality in CRIS. Procedia Comput. Sci. 2022, 211, 3–16.
15. Begoli, E.; Goethert, I.; Knight, K. A Lakehouse Architecture for the Management and Analysis of Heterogeneous Data for Biomedical Research and Mega-Biobanks. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 4643–4651.
16. Armbrust, M.; Ghodsi, A.; Xin, R.; Zaharia, M. Lakehouse: A New Generation of Open Platforms That Unify Data Warehousing and Advanced Analytics. In Proceedings of the CIDR 2021, Virtual, 11–15 January 2021; Volume 8.
17. Al-Bana, M.R.; Farhan, M.S.; Othman, N.A. An Efficient Spark-Based Hybrid Frequent Itemset Mining Algorithm for Big Data. Data 2022, 7, 11.
18. Kandrouch, I.; Oughannou, Z.; Hmina, N.; Chaoui, H. Comparative and Analytical Study of Big Data Technologies: A Survey. In Advanced Intelligent Systems for Sustainable Development (AI2SD'2019); Advances in Intelligent Systems and Computing Book Series; Springer: Cham, Switzerland, 2020; Volume 1105, pp. 184–193.
19. Dias, H.; Henriques, R. Augmenting Data Warehousing Architectures with Hadoop. In Proceedings of the 19th Conference of the Portuguese Association for Information Systems, CAPSI 2019, Lisboa, Portugal, October 2019; Available online: https://aisel.aisnet.org/capsi2019/2 (accessed on 25 January 2024).
20. Drabas, T.; Lee, D. Learning PySpark; Packt Publishing Ltd.: Birmingham, UK, 2017; ISBN 1786466252.
21. Camacho-Rodríguez, J.; Chauhan, A.; Gates, A.; Koifman, E.; O'Malley, O.; Garg, V.; Haindrich, Z.; Shelukhin, S.; Jayachandran, P.; Seth, S.; et al. Apache Hive: From Mapreduce to Enterprise-Grade Big Data Warehousing. In Proceedings of the 2019 International Conference on Management of Data, SIGMOD 2019, Amsterdam, The Netherlands, 30 June–5 July 2019; pp. 1773–1786.
22. Costa, E.; Costa, C.; Santos, M. Evaluating Partitioning and Bucketing Strategies for Hive-Based Big Data Warehousing Systems. J. Big Data 2019, 6, 34.