University
for the
Creative Arts

BERLIN SCHOOL OF
BUSINESS & INNOVATION

Essay / Assignment Title: Big Data Analytics with a Special Focus on Distributed File Systems

Program title: MSc. Data Analytics

Name: Abijith Mullancherry Asokan

Year: 2023

CONTENTS

Contents

Statement of compliance with academic ethics and the avoidance of plagiarism

I honestly declare that this dissertation is entirely my own work and none of its part has been copied from printed or electronic sources, translated from foreign sources, and reproduced from essays of other researchers or students. Wherever I have been based on ideas or other people's texts I clearly declare it through the good use of references following academic ethics.

(In the case that is proved that part of the essay does not constitute an original work, but a copy of an already published essay or from another source, the student will be expelled permanently from the postgraduate program).

Name and Surname (Capital letters): Abijith Mullancherry Asokan

Date: 15/06/2023

INTRODUCTION

We are producing an incredible amount of data every second in the modern digital world. The quantity, velocity, and variety of data being produced are all increasing exponentially, and this includes everything from social media posts and online transactions to sensor readings and machine-generated logs. Big Data, or the enormous amount of data it generates, offers opportunities and difficulties for people, corporations, and society at large.

The term "Big Data" refers to extraordinarily massive and complicated datasets that cannot be efficiently processed or evaluated using conventional data processing methods. The three main dimensions it incorporates are volume, velocity, and diversity. Volume, velocity, and variety all refer to the different sorts and forms of data that are available, while volume is the overall amount of data that is being generated.

Big Data's development has completely changed how we perceive and use information. Data-driven decision-making has the potential to be made possible across a variety of industries, including business, healthcare, finance, and government. To properly tackle the difficulties posed by its collection, storage, processing, and analysis, however, unique tools, techniques, and technologies are needed.

CHAPTER ONE: BIG DATA

What is Big Data:

Big data is a collection of organized, semi-structured, and unstructured data that businesses collect and can mine for information to use in advanced analytics applications like machine learning and predictive modeling. Systems that handle and store big data, in conjunction with technologies that serve big data analytics goals, have become a common component of data management infrastructures in enterprises.

Big data is typically described using the three V's:

- the high *volume* of data in many situations
- The great *variety* of data types often kept in big data systems.
- The *velocity* with which most of the data is created, gathered, and analyzed.

Even if those are its core characteristics, the concept also has the following additional significant characteristics that might be linked to it.

- Veracity: The significance of having trustworthy data is veracity.
- Value: The notion that data should be valuable commercially and offer tangible advantages.
- Variability: The notion that dataflow is not always predictable and can change depending on the circumstances at any given time.

Structural aspect:

The structural description of Big Data refers to understanding the organization, architecture, and frameworks used to handle and store large and complex datasets. The following key aspects are as follows:

Data Structures:

Data and structure point to the idea of data structure by itself. In a big data environment, data comes in a variety of formats. Each component of the data structure has a name, a specific processing method, and a connection pattern. Different procedures are modified according to the kind of data structure to extract the necessary information from each data structure.

Distributed File Systems:

A distributed file system (DFS) is a data management and storage system that allows users or programs to access data files from shared storage across several networked servers, such as PDFs, Word documents, photographs, video clips, and audio recordings. DFS enables several users to share storage resources and data files across numerous workstations thanks to the sharing and storing of data across a cluster of servers.

Data Warehousing:

Consolidating and arranging data from numerous sources into a main repository is the process of data warehousing. These frameworks offer a schema-on-read methodology, enabling flexible Big Data querying and analysis.

NoSQL Databases:

We require highly scalable databases to solve the issue of the high volume of big data. NoSQL databases that operate effectively over several nodes are extremely scalable and ideal for the Big Data environment. Big Data is available in a variety of forms; this is its variety attribute. Since this sort of data adheres to a certain structure, relational databases are not appropriate. NoSQL databases are appropriate for semi-structured data and are better suited to handle the large data's diversity due to their flexibility and lack of a predefined schema.

Data Lakes:

Large volumes of organized, semi structured, and unstructured data may be stored, processed, and secured using a data lake, a centralized repository. Different varieties and magnitudes of data can be processed and saved in its original format.

Data Pipelines:

A set of data processing stages is referred to as a data pipeline. At the start of the pipeline, data is ingested if it has not already been placed into the data platform. After that, there are several phases, each of which produces an output that serves as the input for the following step. Until the pipeline is finished, this continues. Occasionally, separate actions could be carried out in concurrently.

Three essential components make up a data pipeline: a source, a processing step (or stages), and a destination. The last point of a data pipeline is sometimes referred to as a sink.

Data pipelines designed to suit one or more of the three characteristics of big data are known as big data pipelines. Building streaming data pipelines for large data is intriguing due to the velocity of the latter. Data may then be gathered and processed in real-time, allowing for immediate response. Because of the volume of massive data that may alter over time, data pipelines must be scalable. The big data pipeline must be able to scale to process substantial amounts of data concurrently since in fact multiple big data events are likely to occur at once or very near together.

Data Virtualization:

Big Data Virtualization offers a more contemporary method of integrating data while reducing the need for persistent data repositories and related expenses. It functions as a logical data layer that aggregates all company data to provide business users with real-time information.

Processing and Analyzing Big Data:

It is important to have good knowledge about the mathematical aspects of big data to understand how to put these huge amounts of data into good use. These aspects include methods and tools which are essential for modifying and interpreting the data.

The key aspects and procedures included are as follows:

- A summary of the big data is obtained using basic statistical values such as mean, median, mode etc. These values are used for representing the big data using visual representation for better understandability.
- Sampling of the data is done based on the values obtained above. Normal distribution is used for most cases for graphical representation of the whole data. This process helps in getting a subset of the big data.
- Dimensionality reduction is another mathematical concept that helps us in visualizing the big data in lower dimensions for better understandability. Techniques such as Principal Component Analysis, and t-distributed stochastic neighbor embedding (t-SNE) are used to reduce the dimension of the big data.

The data is cleaned and processed for further analyzing. Mathematical concepts such as clustering, regression etc., play a huge role in same. Machine learning and Data mining are then applied to this data to extract meaningful outcomes from this data. Mathematical concepts such as Decision tree algorithm, Linear regression, Clustering, Support Vector Machines are used to achieve the same. Deep learning and Neural networks are used to find hidden patterns among the data.

Big Data frequently consists of connected objects and connections, which are shown as networks or graphs. The study of network structure, connectivity, and dynamics is made possible using mathematical tools and methods called graph theory. To understand complicated interactions in Big Data, network analysis techniques such as centrality metrics, community discovery, and graph clustering are used.

Based on sample data from Big Data, inferences are made about the population using statistical methods. To assess the importance of correlations or differences found in the data, hypothesis testing is used. The statistical significance of results may be ascertained using methods like t-tests, chi-square tests, and analysis of variance (ANOVA).

Difference from Traditional data:

Organizations have been storing and analyzing relational data for decades. Traditional data is structured. The bulk of data in the world is still traditional data.

Traditional data may be used by businesses to monitor sales, manage client relationships, or organize workflows. Traditional data processing software can be used to manage traditional data, which is frequently simpler to manipulate. But compared to big data, it typically offers fewer complex insights and less advantages.

- **Size:** Traditional data is usually measured in Gigabytes and Terabytes. Moreover, these can be stored in one resource compared to big data which is usually measured in petabytes, zettabytes, or exabytes, which require better and huge architectures for proper storage and management.

- **Organization:** Traditional data is organized and structured and relational which makes it easier to manipulate using traditional query languages whereas, big data is raw and unstructured. The dynamic schema is applied to the raw data when large data is accessible. Given that they store data in files, contemporary non-relational or NoSQL databases like Cassandra and MongoDB are perfect for unstructured data.

- **Architecture:** Traditional data can be managed using a centralized architecture. Since big data is scalable and complex, it is not possible to manage it using such simple means. Distributed systems are used to manage them.

- **Sources:** Traditional data is usually obtained from Enterprise resource planning, transactions, Customer Relationships Management etc. On the other hand, big data is obtained from sensors, devices, social media, which is growing at every instant and is unstructured. The data will be in complex form and different data analyzing tools and methods should be used to make use of them.

- **Analysis:** Traditional data analysis happens step-by-step: following an event, data are created, and then the data are analyzed. Traditional data analysis may assist firms in comprehending the effects of certain strategies or adjustments on a defined set of metrics over a predetermined time frame. Whereas, big data analysis may take place while data is being gathered since it creates data at a second-by-second rate. Businesses may gain a more dynamic and comprehensive picture of their requirements and plans via the use of big data analysis.

Future Considerations:

Both Traditional data and big data provide different but related purposes. Big data can provide much higher potential, but not necessary in all cases as it may vary depending on the kind of data.

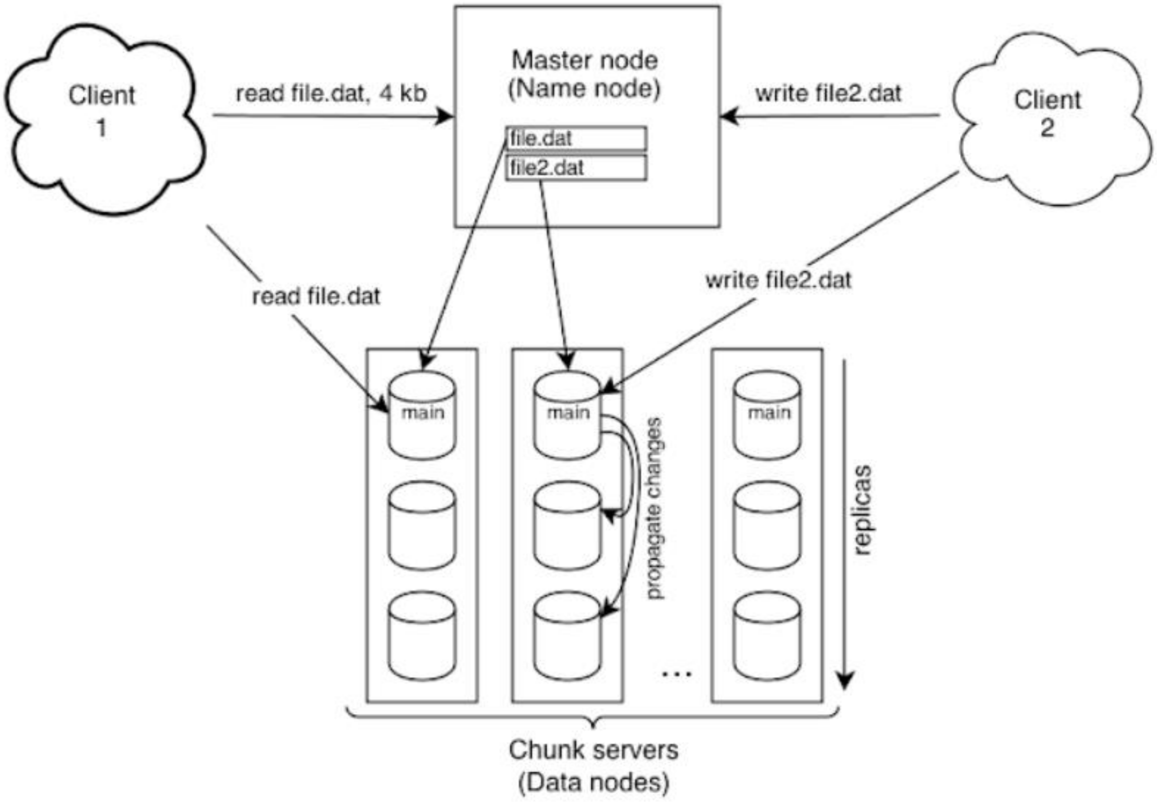
Advantages of big data:

- Deeper analysis can provide businesses with market trends and consumer behavior.
- Can provide competitive edge as data can be used for learning in real time.
- More efficient as this data can harnessed to achieve meaningful outcomes.
- Needs more processing power and better storage technologies to make use of this resource.

As more and more data are being created at every instance, it is required to know what tools to be used at different times for making use of the data generated.

CHAPTER TWO: Importance of Distributed File Systems in Big Data

A file system that spans over several physical locations in different servers is called a Distributed File System (DFS). Files may be accessed as if they were locally saved, from any device and from anywhere on the network. A DFS facilitates the regulated and permitted sharing of information and data among network users. They provide a Big Data analytics storage solution that is scalable, fault-tolerant, and high-performance.



The main reason for choosing DFS by enterprises is because files can be accessed from different locations for fast access.

Compared to other file systems, using DFS provides us with many advantages.

- Data Distribution and Replication: DFSs spread data across several cluster nodes, ensuring that data is redundantly stored for fault tolerance and high availability. This distributed data distribution and replication technique aids in the reduction of data loss while increasing data accessible. Traditional file systems or centralized storage solutions, on the other hand, may not provide the same level of fault tolerance or scalability.

- Scalability: DFSs are built to manage large amounts of data by allowing for horizontal scalability. As data expands, more nodes can be added to the cluster to meet the growing storage needs. DFSs are well-suited for managing the ever-increasing amount of Big Data because to their scalability. Traditional file systems, on the other hand, may have scalability constraints when dealing with enormous datasets.

- Parallel Processing: DFSs provide parallel data processing across several cluster nodes. This enables distributed and parallel data processing, which provides for quicker data access and analysis. Big Data analytics frequently involves complicated computing processes that may be partitioned and run in parallel over numerous nodes, yielding considerable performance gains. Traditional file systems or single-node storage solutions, on the other hand, may not provide the same level of parallel processing capabilities.

- Fault Tolerance: DFSs have systems in place to provide fault tolerance and data dependability. Data replication over many nodes mitigates data loss in the event of node failure. Furthermore, DFSs have systems for detecting and recovering from node failures, which ensures continuous data availability. Traditional file systems may not have the same level of fault tolerance and recovery capabilities as modern file systems.

- Data Locality: DFSs optimize data access by taking use of data locality. Data is kept on the same nodes that do calculations, reducing data transfer and enhancing overall speed. This type of locality-aware data access is critical in Big Data analytics, where vast amounts of data must be handled effectively. Traditional file systems, on the other hand, may not consider data locality and may incur higher data transport cost.

- Integration with Big Data Ecosystem: DFSs, such as Hadoop Distributed File System (HDFS), are essential components of the Big Data ecosystem. They function in tandem with other components such as data processing frameworks (for example, MapReduce and Spark), data integration tools (for example, Apache Kafka and Apache NiFi), and analytics platforms. This connection creates a unified platform for Big Data analytics, allowing for more efficient data storage, processing, and analysis. Such interactions with specialist Big Data tools and frameworks may be lacking in traditional file systems.

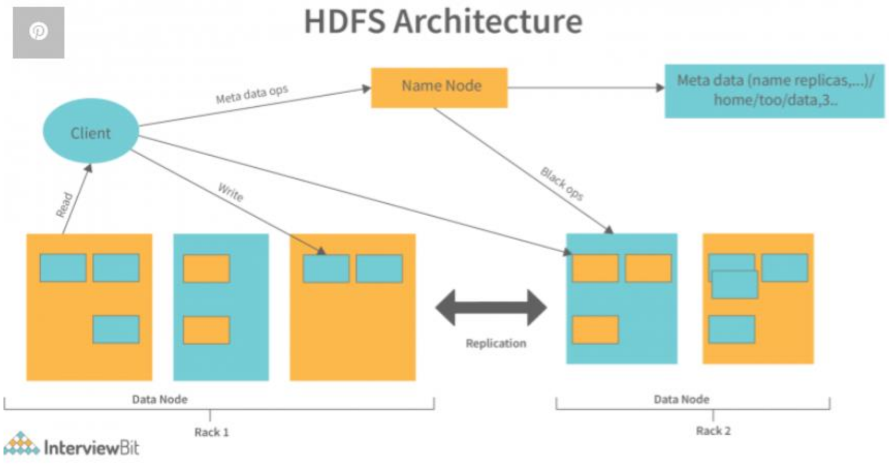
All these advantages of DFS are in accordance to why they are much more used and needed for Big Data Analytics.

CHAPTER 3: Hadoop Distributed File System

HDFS stands for Hadoop Distributed File System. It is a distributed file system that forms the core component of the Apache Hadoop framework. HDFS is designed to store and manage large volumes of data across multiple machines in a distributed computing environment.

HDFS is fault-tolerant and intended for use on low-cost commodity hardware. In Apache Hadoop, HDFS allows high throughput data access to application data and is appropriate for applications with huge data sets. It also enables streaming access to file system data.

So, what exactly is Hadoop? How does it differ from HDFS? The primary distinction between Hadoop and HDFS is that Hadoop is an open-source framework that can store, process, and analyze data, whereas HDFS is Hadoop's file system that allows data access.



As we can see, it is primarily concerned with NameNodes and DataNodes. The hardware that houses the GNU/Linux operating system and applications is known as the NameNode. The Hadoop distributed file system serves as the master server, managing files, controlling client access to files, and supervising file operations such as renaming, opening, and shutting files.

A piece of hardware that runs the GNU/Linux operating system and the DataNode software is called DataNode. A DataNode will be found for each node in an HDFS cluster. These nodes aid in the control of their system's data storage by performing actions on file systems as requested by the client, as well as creating, replicating, and blocking files as directed by the NameNode.

Learning from Big Data in Hadoop entails utilizing the Hadoop ecosystem's capabilities to extract insights and knowledge from massive amounts of data. The data is ingested into the HDFS. The transferred data is then prepared for further analysis. This process involves cleaning the data, filtering, and transforming data into structured form for better data quality.

Hadoop also incorporates **fault tolerance** mechanisms so that they are more reliable and the data is always available for computation. Since, data is replicated at different nodes making recovery easier in case of failure at any one node. Hadoop also provides tools which check the nodes health and transfers and stores data to healthy nodes on periodic intervals.

Moreover, in Hadoop, data is processed on the same node as which they reside, making the data transfer time shorter and minimizing latency, which all contributes towards faster data access and analysis. This property is highly beneficial when dealing with Big Data

The processed data is then used for further analysis using tools and frameworks provided by Hadoop. **MapReduce** is most used framework for this, where complex tasks are reduced into small simple tasks for analysis. This enables faster and more efficient analysis by use of parallel processing.

Flexibility and Compatibility: Machine learning libraries and frameworks such as Apache Mahout, Apache Spark MLlib, and TensorFlow on Hadoop are also part of the Hadoop ecosystem. These technologies make it possible to apply sophisticated analytics approaches to Big Data, such as predictive modeling, clustering, classification, and recommendation systems. Hadoop provides tools for visualization of these insights from the Big Data, which can help businesses take data driven decisions.

As Hadoop is **Scalable** and Big Data is a never-ending stretch of data which keeps on growing indefinitely. Hadoops architecture provides with the ease of storing more data as they are generated and provides all these tools by which these processes can be done in an iterative manner which helps in making use of the generated data in an efficient manner.

Cost-Effectiveness: Hadoop is built on commodity hardware and open-source software, making it a cost-effective solution for Big Data processing. It provides a scalable and distributed infrastructure without the need for expensive specialized hardware. This cost-effectiveness allows organizations to store and process large amounts of data at a fraction of the cost compared to traditional data processing systems.

CONCLUDING REMARKS

We have looked at three major features of Big Data: its basic, structural, and mathematical properties, the significance of Distributed File Systems (DFSs) in Big Data research, and the method of learning from Big Data using Hadoop.

Fundamentally, Big Data is differentiated by its vast volume, diverse data formats, rapid data production, and the need to address data quality concerns. It is distinct from regular data in terms of scale, complexity, and the need for specialist tools and methodologies.

DFSs, such as Hadoop's HDFS, provide a distributed and fault-tolerant file system infrastructure optimized for managing and processing Big Data. DFSs provide characteristics like as fault tolerance, scalability, and data localization optimization, making them well-suited to the Big Data issues. The distributed structure of DFSs enables parallel processing and optimal resource usage over numerous nodes in a cluster, allowing for the handling of large-scale data sets.

Hadoop, a prominent Big Data processing technology, offers a rich environment for Big Data learning. Its scalability lets it to process huge amounts of data, while fault tolerance measures assure data dependability and availability. Hadoop's data localization optimization reduces data transport while increasing processing performance. Furthermore, Hadoop's interoperability with a variety of tools and frameworks, including as MapReduce, Apache Spark, Hive, and Pig, makes sophisticated analytics and machine learning on Big Data possible.

Organizations may use Hadoop's traits and features to preprocess, analyze, and draw insights from Big Data, resulting in informed decision-making. The integration of Hadoop with visualization and reporting tools allows for better sharing of Big Data insights.

Big Data provides new difficulties and possibilities that necessitate the use of specialized infrastructure and tools for successful administration and analysis. DFSs, notably Hadoop's HDFS, provide critical functionality for Big Data storage, processing, and administration. Hadoop provides a robust platform for learning from Big Data and making data-driven choices because to its scalability, fault tolerance, data locality optimization, and interoperability with modern analytics tools. Organizations may obtain important insights, promote innovation, and gain a competitive edge in the domain of data-driven decision-making by embracing the potential of Big Data and exploiting Hadoop's capabilities.

BIBLIOGRAPHY

- Botelho, B. and Bigelow, S.J. (2022) What is Big Data and why is it important?, Data Management. Available at: https://www.techtarget.com/searchdatamanagement/definition/big-data#:~:text=Big%20data%20is%20a%20combination,and%20other%20advanced%20analytics%20applications. (Accessed: 12 June 2023).
- Sydle (2023) Big data: Definition, importance, and types, Blog SYDLE. Available at: https://www.sydle.com/blog/big-data-definition-importance-and-types-614b791388e600016afa7fc3#:~:text=with%20IoT%20technology.-,What’s%20the%20importance%20of%20Big%20Data%3F,to%20more%20efficient%20decision%2Dmaking. (Accessed: 12 June 2023).
- Bose, P. (2022) Structured data in Big Data: What it is and why is it important, Blogs &amp; Updates on Data Science, Business Analytics, AI Machine Learning. Available at: https://www.analytixlabs.co.in/blog/structured-data-in-big-data/ (Accessed: 12 June 2023).
- Distributed file systems - what is DFS? (no date) Nutanix. Available at: https://www.nutanix.com/info/distributed-file-systems (Accessed: 13 June 2023).
- Data pipeline (2023) Hazelcast. Available at: https://hazelcast.com/glossary/data-pipeline/ (Accessed: 13 June 2023).
- Reddy, A. (2018) NoSQL databases and Big Data, Medium. Available at: https://medium.com/@arunbollam/nosql-databases-and-big-data-57562e93f302 (Accessed: 13 June 2023).
- Simplilearn (2023) Exploring descriptive statistics: Everything you need to know!: Simplilearn, Simplilearn.com. Available at: https://www.simplilearn.com/what-is-descriptive-statistics-article (Accessed: 13 June 2023).
- Big Data vs. Traditional Data (no date) Pure Storage. Available at: https://www.purestorage.com/knowledge/big-data/big-data-vs-traditional-data.html (Accessed: 13 June 2023).
- Distributed file system (2022) Cohesity. Available at: https://www.cohesity.com/glossary/distributed-file-system/ (Accessed: 13 June 2023).

- Distributed File Systems (DFS) (2023) WEKA. Available at: https://www.weka.io/learn/distributed-file-systems/distributed-file-system/#:~:text=Features%20of%20Distributed%20File%20System,by%20the%20host%20accessing%20it. (Accessed: 14 June 2023).

- The advantages and disadvantages of the Distributed File System (DFS) (2020) FROMDEV. Available at: https://www.fromdev.com/2020/10/the-advantages-and-disadvantages-of-the-distributed-file-system-dfs.html (Accessed: 14 June 2023).

- What is Hadoop Distributed File System (HDFS) (no date) Databricks. Available at: https://www.databricks.com/glossary/hadoop-distributed-file-system-hdfs (Accessed: 14 June 2023).

- Key design of HDFS architecture (no date) Section. Available at: https://www.section.io/engineering-education/key-design-of-hdfs-architecture/#general-design-of-hdfs-architecture (Accessed: 14 June 2023).

- HDFS architecture - detailed explanation (2022) InterviewBit. Available at: https://www.interviewbit.com/blog/hdfs-architecture/#:~:text=HDFS%20is%20an%20Open%20source,up%20the%20architecture%20of%20HDFS. (Accessed: 14 June 2023).