SIMRAN.H. LAHANE ,PF-58

ASSIGNMENT NO: 04

Problem Statement:
Install Hadoop and perform basic Hadoop commands on it.

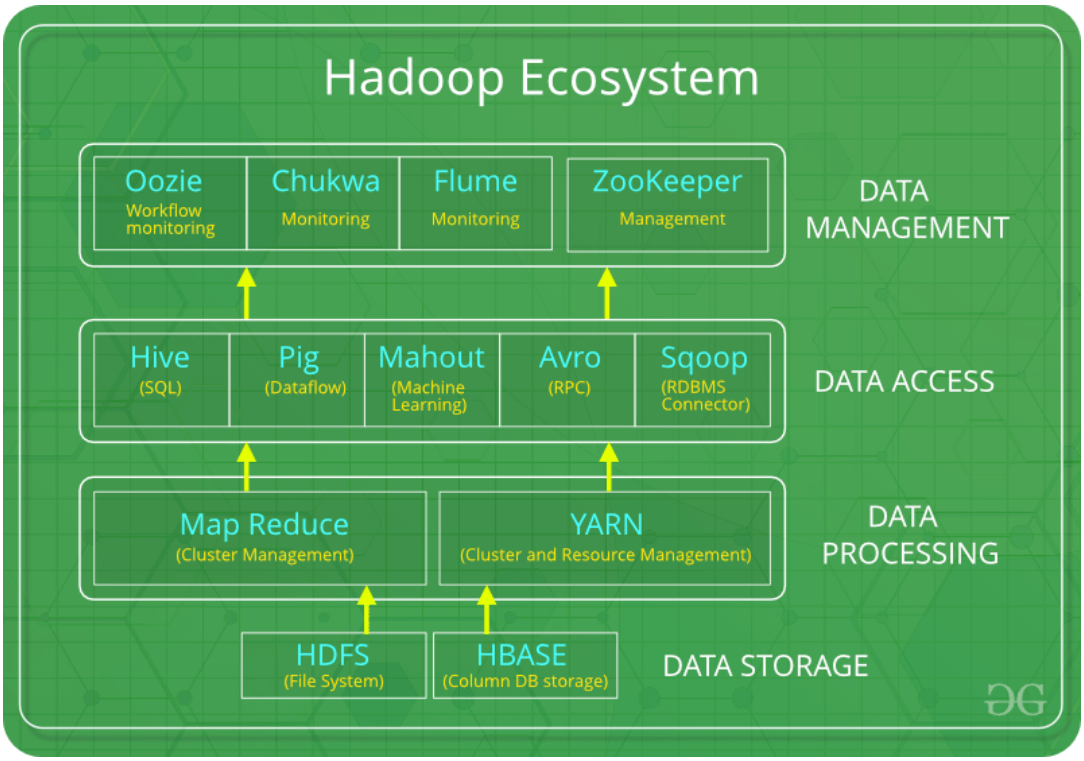
Objectives:

1. To learn concepts of Hadoop ecosystem
2. To learn how to install Hadoop and Perform basic Hadoop commands

Theory:

Explain following concepts

- **Draw the Hadoop Ecosystem diagram**



- **Explain the History of Hadoop in short**

Over the years, Hadoop has evolved into a comprehensive ecosystem of tools and frameworks for big data processing. The Hadoop ecosystem includes HDFS, YARN, MapReduce, Hive, Pig, Spark, HBase, ZooKeeper, and many other tools.

Hadoop is used by a wide range of companies and organizations, including Yahoo!, Facebook, Twitter, Netflix, Spotify, and many others. Hadoop is used for a variety of big data tasks, such as web search, social media analysis, machine learning, and scientific computing.

Here is a brief timeline of the history of Hadoop:

- 2002: Doug Cutting and Mike Cafarella start working on the Apache Nutch project.
- 2003: Google publishes a paper on its distributed file system, called GFS.
- 2006: Doug Cutting joins Yahoo! and develops the NDFS distributed file system. He renames the project Hadoop after his son's toy elephant.
- 2008: Hadoop is released as an open-source project under the Apache Software Foundation.
- 2009: Hadoop 0.20 is released, which introduces the YARN resource management framework.
- 2010: Hadoop 1.0 is released, which introduces the Hive data warehouse infrastructure and the Pig high-level language.
- 2012: Hadoop 2.0 is released, which includes a number of improvements to HDFS, YARN, and MapReduce.
- 2014: Apache Spark becomes a top-level Apache project.
- 2016: Hadoop 3.0 is released, which includes a number of improvements to HDFS, YARN, and MapReduce.
- 2023: Hadoop continues to be a widely used framework for big data processing.

Hadoop has revolutionized the way that big data is processed and analyzed. It has made it possible for companies and organizations to store and process massive datasets that would have been impossible to process just a few years ago.

- **Explain HDFS architecture in detail with suitable diagram**

The Hadoop Distributed File System (HDFS) is a distributed file system that stores data across multiple nodes in a cluster. It is highly scalable and fault-tolerant, making it ideal for storing large datasets.

HDFS has a master/slave architecture. The master node, called the NameNode, manages the file system namespace and regulates access to files by clients. The slave nodes, called DataNodes, store the data blocks that make up the files.

HDFS Architecture Diagram

Components of HDFS

- NameNode: The NameNode is the master node in the HDFS cluster. It is responsible for managing the file system namespace and regulating access to files by clients. The NameNode maintains a metadata table that maps file names to their corresponding data blocks.
- DataNode: DataNodes are the slave nodes in the HDFS cluster. They are responsible for storing the data blocks that make up the files. DataNodes periodically report their block status to the NameNode.
- Client: Clients are applications that use the HDFS file system to store and retrieve data. Clients communicate with the NameNode to get information about files and to get the locations of data blocks.

How HDFS Works

When a client wants to create a file in HDFS, it sends a request to the NameNode. The NameNode creates an entry in the metadata table for the file and assigns it a unique identifier. The NameNode then replicates the file across multiple DataNodes in the cluster.

When a client wants to read a file from HDFS, it sends a request to the NameNode. The NameNode returns the locations of the data blocks for the file to the client. The client then reads the data blocks from the DataNodes.

HDFS is highly scalable and fault-tolerant. It can scale to support petabytes of data and thousands of nodes. HDFS is also fault-tolerant because it replicates data across multiple DataNodes. If a DataNode fails, the other DataNodes in the cluster can still serve the data.

Benefits of Using HDFS

- Scalability: HDFS can scale to support petabytes of data and thousands of nodes.
- Fault tolerance: HDFS is fault-tolerant because it replicates data across multiple DataNodes.
- High performance: HDFS is highly performant for large datasets and workloads.
- Cost-effective: HDFS is a cost-effective solution for storing and processing large datasets.

Use Cases for HDFS

HDFS is used by a wide range of companies and organizations to store and process large datasets. Some common use cases for HDFS include:

- Web search
- Social media analysis
- Machine learning
- Scientific computing
- Log processing
- Data warehousing
- Data archiving

HDFS is a powerful tool for storing and processing large datasets. It is scalable, fault-tolerant, high performance, and cost-effective. HDFS is used by a wide range of companies and organizations for a variety of big data tasks.

Platform: 64 –bit Open source Linux/Windows

Conclusion: Hence, I learned to install Hadoop and perform basic Hadoop commands on it.

FAQs: write answers

1) Explain with syntax and example of any 10 basic Hadoop commands

Ans

Here is a list of 10 basic Hadoop commands with their syntax and examples:

Command	Syntax	Example
ls	hadoop fs -ls <path>	List the contents of a directory in HDFS. For example, hadoop fs -ls /user/my_user/data will list the contents of the /user/my_user/data directory.

mkdir	hadoop fs -mkdir <path>	Create a directory in HDFS. For example, hadoop fs -mkdir /user/my_user/data will create a directory called data in the /user/my_user directory.
rm	hadoop fs -rm <path>	Delete a file or directory from HDFS. For example, hadoop fs -rm /user/my_user/data/my_file.txt will delete the file my_file.txt from the /user/my_user/data directory.
cp	hadoop fs -cp <path1> <path2>	Copy a file or directory from one location in HDFS to another. For example, hadoop fs -cp /user/my_user/data/my_file.txt /user/my_user/data_backup will copy the file my_file.txt from the /user/my_user/data directory to the /user/my_user/data_backup directory.
mv	hadoop fs -mv <path1> <path2>	Move a file or directory from one location in HDFS to another. For example, hadoop fs -mv /user/my_user/data/my_file.txt /user/my_user/data_backup will move the file my_file.txt from the /user/my_user/data directory to the /user/my_user/data_backup directory.
put	hadoop fs -put <local_file> <hdfs_path>	Copy a file from a local file system to HDFS. For example, hadoop fs -put my_file.txt /user/my_user/data will copy the file my_file.txt from the current working directory to the /user/my_user/data directory in HDFS.
get	hadoop fs -get <hdfs_path> <local_file>	Copy a file from HDFS to a local file system. For example, hadoop fs -get /user/my_user/data/

		my_file.txt ./ will copy the file my_file.txt from the /user/my_user/data directory in HDFS to the current working directory.
copyFromLocal	hadoop fs -copyFromLocal <source_dir> <hdfs_dir>	Copy a directory and all of its contents from a local file system to HDFS. For example, hadoop fs -copyFromLocal /data/my_data /user/my_user/data will copy the directory /data/my_data and all of its contents to the /user/my_user/data directory in HDFS.
copyToLocal	hadoop fs -copyToLocal <hdfs_dir> <destination_dir>	Copy a directory and all of its contents from HDFS to a local file system. For example, hadoop fs -copyToLocal /user/my_user/data /data/my_data will copy the directory /user/my_user/data and all of its contents to the /data/my_data directory on the local file system.
du	hadoop fs -du <path>	Calculate the disk usage of a file or directory in HDFS. For example, hadoop fs -du /user/my_user/data will calculate the disk usage of the /user/my_user/data directory.
text	hadoop fs -text <path>	Display the contents of a text file in HDFS. For example, hadoop fs -text /user/my_user/data/my_file.txt will display the contents of the file my_file.txt in the /user/my_user/data directory.

These are just a few of the many Hadoop commands that are available. For more information, please consult the Hadoop documentation.

2) State the use of Name node and Data node
Ans

The NameNode and DataNode are the two main components of the Hadoop Distributed File System (HDFS).

NameNode

The NameNode is the master node in the HDFS cluster. It is responsible for managing the file system namespace and regulating access to files by clients. The NameNode maintains a metadata table that maps file names to their corresponding data blocks.

DataNode

DataNodes are the slave nodes in the HDFS cluster. They are responsible for storing the data blocks that make up the files. DataNodes periodically report their block status to the NameNode.

Use of NameNode and DataNode

The NameNode and DataNode work together to provide a reliable and scalable file system for storing large datasets.

When a client wants to create a file in HDFS, it sends a request to the NameNode. The NameNode creates an entry in the metadata table for the file and assigns it a unique identifier. The NameNode then replicates the file across multiple DataNodes in the cluster.

When a client wants to read a file from HDFS, it sends a request to the NameNode. The NameNode returns the locations of the data blocks for the file to the client. The client then reads the data blocks from the DataNodes.

The NameNode is a single point of failure in the HDFS cluster. However, HDFS provides a number of features to mitigate this risk, such as checkpointing and failover.

Benefits of Using HDFS

- Scalability: HDFS can scale to support petabytes of data and thousands of nodes.

- Fault tolerance: HDFS is fault-tolerant because it replicates data across multiple DataNodes.
- High performance: HDFS is highly performant for large datasets and workloads.
- Cost-effective: HDFS is a cost-effective solution for storing and processing large datasets.

Use Cases for HDFS

HDFS is used by a wide range of companies and organizations to store and process large datasets. Some common use cases for HDFS include:

- Web search
- Social media analysis
- Machine learning
- Scientific computing
- Log processing
- Data warehousing
- Data archiving

HDFS is a powerful tool for storing and processing large datasets. It is scalable, fault-tolerant, high performance, and cost-effective. HDFS is used by a wide range of companies and organizations for a variety of big data tasks.

3) State the different applications of Hadoop

Ans

Hadoop has a wide range of applications in various industries, including:

- Web search: Hadoop is used by search engines to index and search the web. For example, Google uses Hadoop to index and search the trillions of web pages that it has crawled.

- Social media analysis: Hadoop is used by social media companies to analyze user data, such as posts, likes, and shares. This data can be used to understand user behavior and to improve the user experience.
- Machine learning: Hadoop is used to train and deploy machine learning models. For example, Netflix uses Hadoop to train and deploy its recommendation engine.
- Scientific computing: Hadoop is used by scientists to process large datasets from experiments and simulations. For example, the Large Hadron Collider uses Hadoop to process the petabytes of data that it generates each year.
- Log processing: Hadoop is used to process and analyze log data from servers and other applications. This data can be used to identify problems and to improve performance.
- Data warehousing: Hadoop can be used to build data warehouses for storing and analyzing large datasets.
- Data archiving: Hadoop can be used to archive large datasets that are no longer needed for daily operations.

In addition to these specific applications, Hadoop can also be used for a variety of other tasks, such as:

- Data mining: Hadoop can be used to mine large datasets for patterns and insights.
- Natural language processing: Hadoop can be used to process and analyze natural language text, such as tweets and emails.
- Fraud detection: Hadoop can be used to detect fraudulent transactions and other suspicious activity.
- Risk management: Hadoop can be used to assess and manage risk.
- Financial analysis: Hadoop can be used to analyze financial data, such as stock prices and market trends.

Hadoop is a versatile tool that can be used for a wide range of tasks. It is used by companies and organizations of all sizes to store, process, and analyze large datasets.