MapReduce –Introduction

- **MapReduce** is a programming model and an associated implementation for processing and generating big data sets with a parallel, distributed algorithm on a cluster.
- (https://en.wikipedia.org/wiki/MapReduce)

1

MapReduce –Introduction

- MapReduce Architecture is a programming model and a software framework utilized for preparing enormous measures of data.
- MapReduce program works in two stages, to be specific, Map and Reduce.
- Map requests that arrange with mapping and splitting of data while Reduce tasks reduce and shuffle the data.

2

MapReduce – History

- MapReduce was first popularized as a programming model in 2004 by Jeffery Dean and Sanjay Ghemawat of Google.
- In their paper, "MAPREDUCE: SIMPLIFIED DATA PROCESSING ON LARGE CLUSTERS," they discussed Google's approach to collecting and analyzing website data for search optimizations.

3

MapReduce – History

- Google's proprietary MapReduce system ran on the Google File System (GFS).
- Apache began using MapReduce in the "Nutch" project, and moved it to the new Hadoop subproject in January 2006.
- Hadoop began as a subproject in the Apache Lucern project, which provides text search capabilities across large databases.
  - (In 2006, Doug Cutting, an employee of Yahoo!, designed Hadoop, naming it after his son's toy elephant.)

4

MapReduce – History

- Hadoop was released as an open source Apache project in 2007.
- In 2008, Hadoop became a top level project at Apache.
- On July 2008, an experimental 4000 node cluster was created using Hadoop, and in 2009 during a performance test, Hadoop was able to sort a terabyte of data in 17 hours.

5

MapReduce – History

Why MapReduce?

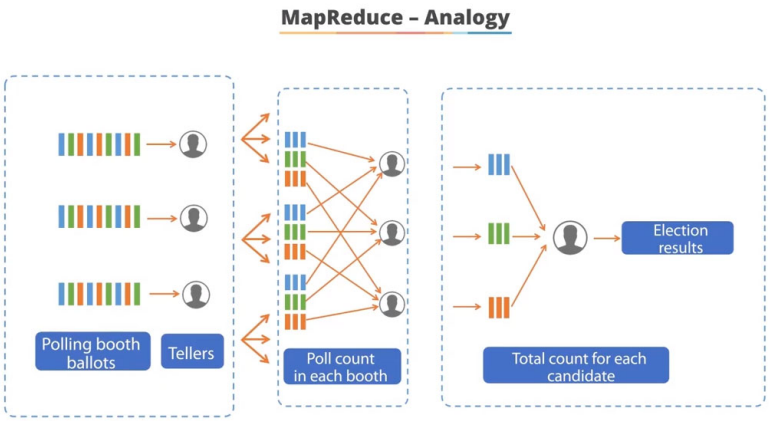
Huge amounts of data were stored in single servers prior to 2004.

The threat of data loss, challenge of data backup, and reduced scalability resulted in the issue snowballing into a crisis of sorts.

Queries could run simultaneously on multiple servers, search results could belogically integrated, and data could be analyzed in real time.

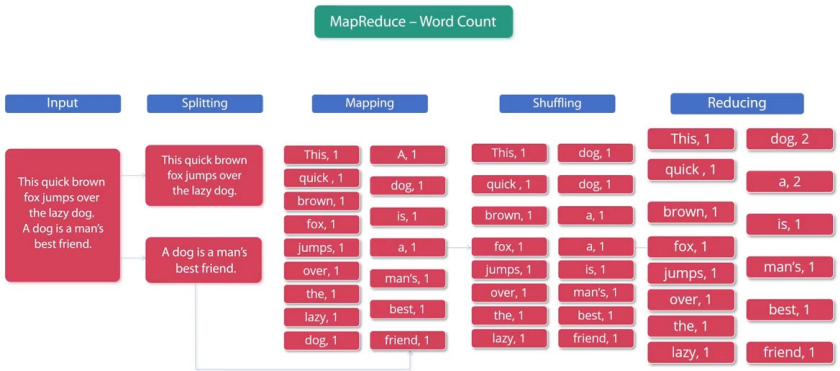
6

MapReduce – Analogy



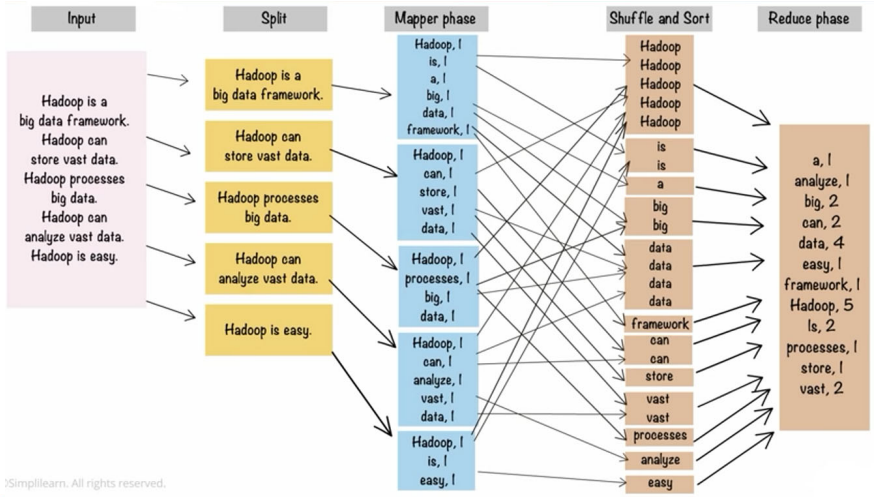
7

MapReduce – Example 1



8

MapReduce – Example 2



9

MapReduce – Example

MapReduce Jobs

A job is a MapReduce program that causes multiple map and reduce functions to run parallelly over the life of the program.

ApplicationMaster and NodeManager functions:

Application Master

- Responsible for the execution of a single application or MapReduce job
- Divides job requests into tasks and assigns them to NodeManagers running on the slave node

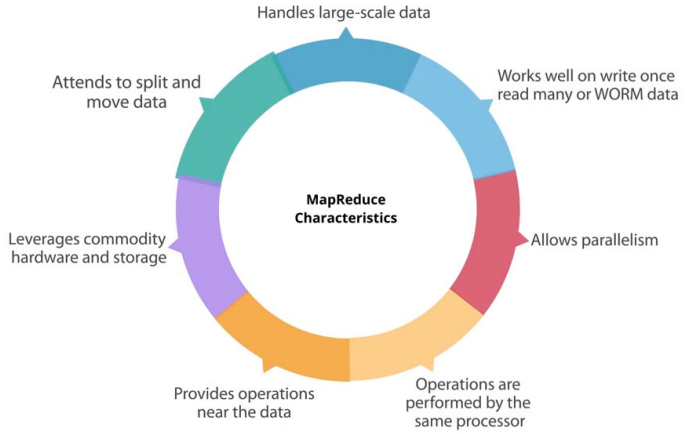
NodeManager

- Has many dynamic resource containers
- Executes each active map or reduce task
- Communicates regularly with the Application Master

10

MapReduce – Example

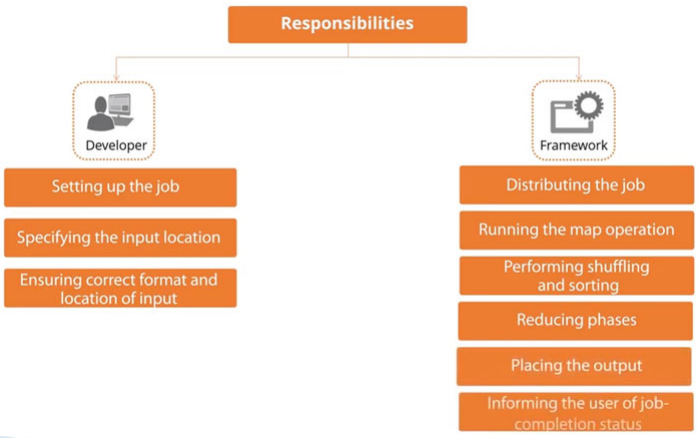
Characteristics of MapReduce



11

MapReduce – Example

MapReduce - Responsibilities



12

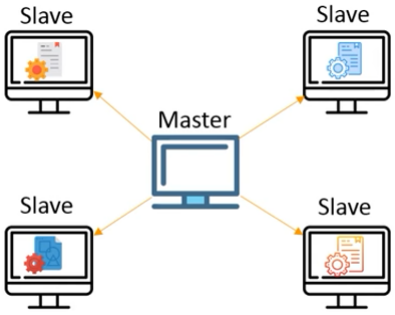
Hadoop

- Hadoop is for parallel processing with distributed storage.
- Hadoop can be installed on any commodity hardware.
  - Commodity hardware is affordable and easy to obtain. It can be a low-performance system such as IBM PC-compatible or Linux.
- Hadoop distributed file system (HDFS) handles large data sets running on commodity hardware and it is highly fault tolerant.

13

Hadoop

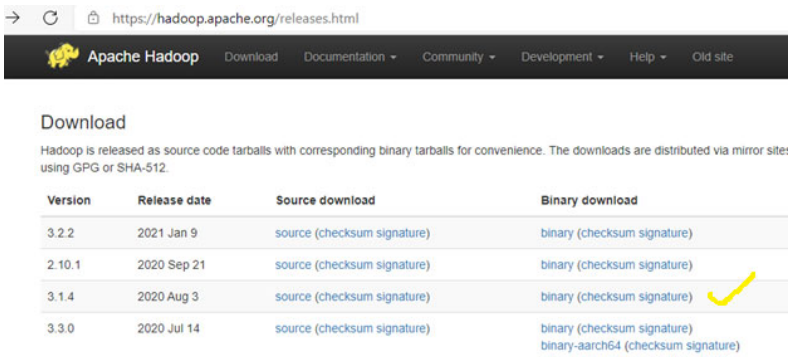
Data is processed at the Slave nodes in MapReduce



14

Hadoop

- How to install Hadoop:



15

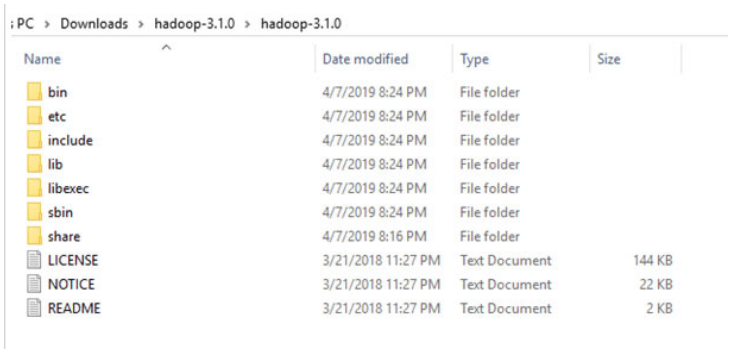
Hadoop

- How to install Hadoop:

16

Hadoop

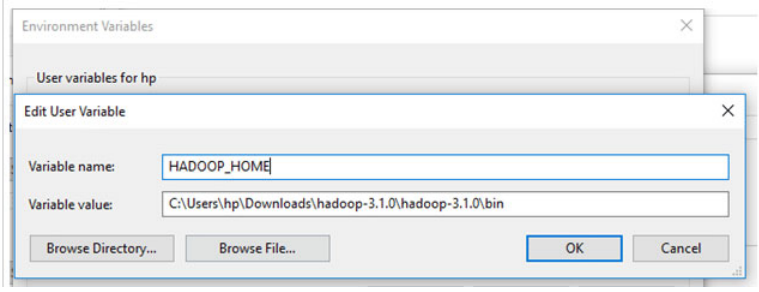
Extract it to a folder:



17

Hadoop

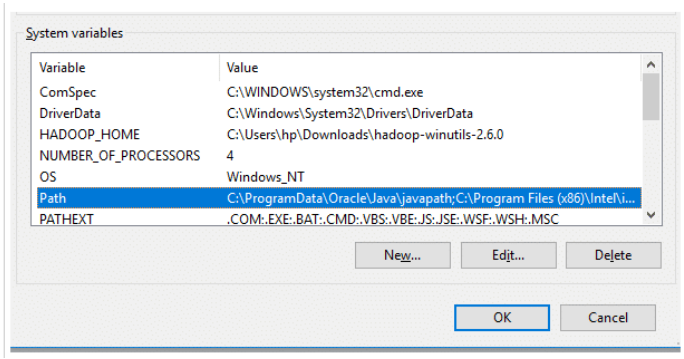
Create a new user variable. Put the Variable_name as HADOOP_HOME and Variable_value as the path of the bin folder where you extracted hadoop.



18

Hadoop

Set Hadoop bin directory path in system variable path.

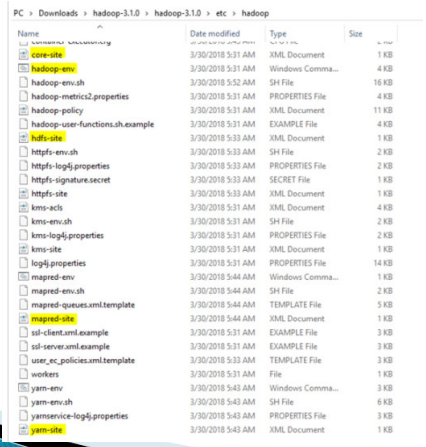


19

Hadoop

Configurations

Need to edit some files located in the hadoop directory of the etc folder where we installed hadoop. The files that need to be edited have been highlighted.



20

Hadoop

Edit the file core-site.xml in the hadoop directory.

```
1/span>configuration>
2   /span>property>
3       /span>name>fs.defaultFS/span>/name>
4       /span>value>hdfs://localhost:9000</value>
5   /span>/property>
6/span>/configuration>
```

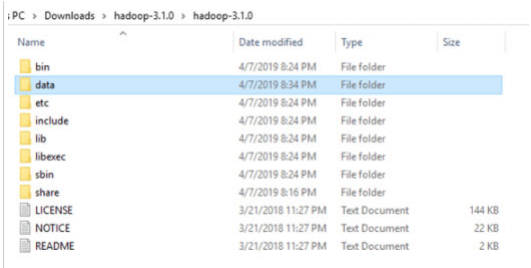
Edit mapred-site.xml

```
1/span>configuration>
2   /span>property>
3       /span>name>mapreduce.framework.name/span>/name>
4       /span>value>yarn/span>/value>
5   /span>/property>
6/span>/configuration>
```

21

Hadoop

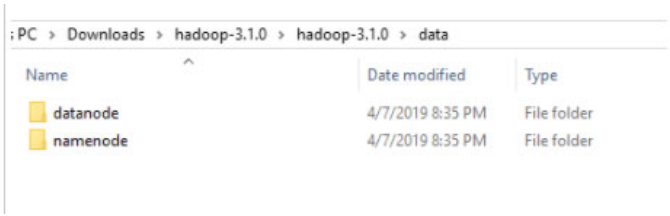
Create a folder ‘data’ in the hadoop directory



22

Hadoop

Create a folder with the name ‘datanode’ and a folder ‘namenode’ in this data directory



23

Hadoop

Edit the file hdfs-site.xml and add below property in the configuration

Note: The path of namenode and datanode across value would be the path of the datanode and namenode folders you just created.

```
1  /span>configuration>
2     /span>property>
3        /span>name>dfs.replication/span>/name>
4        /span>value>1/span>/value>
5     /span>/property>
6     /span>property>
7        /span>name>dfs.namenode.name.dir/span>/name>
8        /span>value>C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\data\namenode/span>/value>
9     /span>/property>
10    /span>property>
11       /span>name>dfs.datanode.data.dir/span>/name>
12       /span>value> C:\Users\hp\Downloads\hadoop-3.1.0\hadoop-3.1.0\data\datanode/span>/value>
13    /span>/property>
14/span>/configuration>
```

24

Hadoop

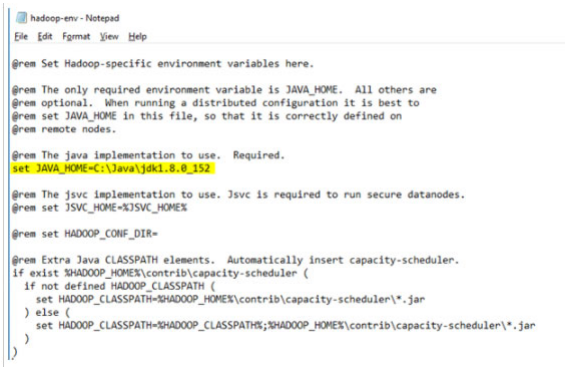
Edit the file yarn-site.xml and add below property in the configuration

```
1  /span>configuration>
2     /span>property>
3       /span>name>yarn.nodemanager.aux-services/span>/name>
4       /span>value>mapreduce_shuffle/span>/value>
5     /span>/property>
6     /span>property>
7        /span>name>yarn.nodemanager.auxservices.mapreduce.shuffle.class/span>/name>
8  /span>value>org.apache.hadoop.mapred.ShuffleHandler/span>/value>
9     /span>/property>
10/span>/configuration>
```

25

Hadoop

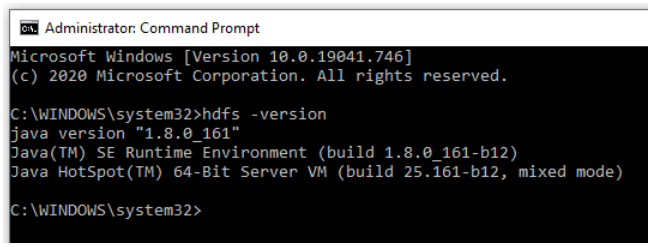
Edit hadoop-env.cmd and replace %JAVA_HOME% with the path of the java folder where your jdk 1.8 is installed



26

Hadoop

6. Check the Hadoop version.



cd C:\hadoop\sbin
Start namenode and datanode with this command –
> start-dfs.cmd
Start yarn through this command-

- start-yarn.cmd
- Or
- start-all.cmd

27

Hadoop

More windows will open, one for yarn resource manager and one for yarn node manager.

28

Hadoop

1. Create an input directory in HDFS.
   cd to the hadoop directory
   >hadoop fs -mkdir /input_dir
2. Copy the input text file named input_file.txt in the input directory (input_dir) of HDFS.
   >hadoop fs -put C:/input_file.txt /input_dir

```
C:\>hadoop dfs -cat /input_dir/input_file.txt
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
23   23   27  43   24   25   26   26   26   26   25   26  25
26   27   28  28   28   30   31   31   31   30   30   30  29
31   32   32  32   33   34   35   36   36   34   34   34  34
39   38   39  39   39   41   42   43   40   39   38   38  40
38   39   39  39   39   41   41   41   28   40   39   39  45
```

29

Hadoop

3. Run MapReduceClient.jar and also provide input and out directories.
   >hadoop jar C:/MapReduceClient.jar wordcount /input_dir /output_dir
4. Verify content for generated output file.
   >hadoop dfs -cat /output_dir/*

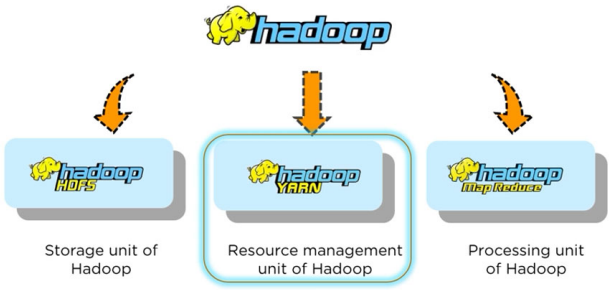
```
C:\>hadoop dfs -cat /output_dir/*
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
23      12
24      6
25      18
26      36
27      12
28      24
29      6
30      24
31      24
32      18
33      6
34      30
35      6
36      12
38      24
39      66
40      18
41      24
42      6
43      12
45      6

C:\>
```

30

Hadoop

Components of Hadoop version 2.0

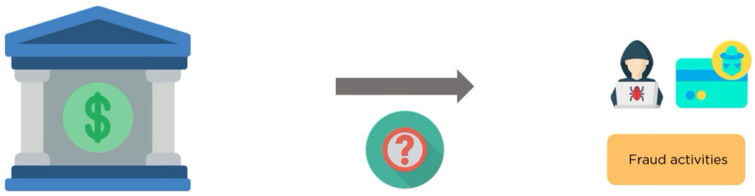


31

Hadoop

Hadoop use case – Combating fraudulent activities

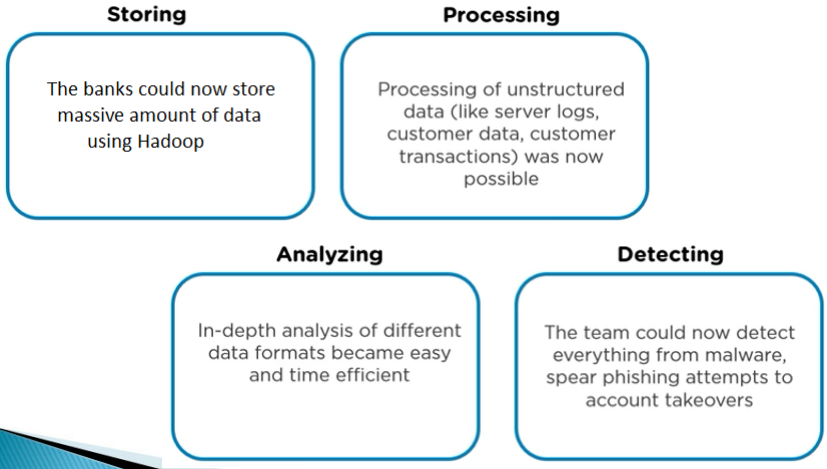
Detecting fraudulent transactions is one among the various problems any bank faces



32

Hadoop

How Hadoop solved the problem.



33

Hadoop and Data Science

Hadoop for Data Science – **An important tool for Data Scientists**. Hadoop is an important tool for data science when the volume of data exceeds the system memory or when the business case requires data to be distributed across multiple servers.

Example: Analyzing Customer Data for a Retail Company

34

Hadoop and Data Science

Example:

The retail company wants to analyze its customer data to gain insights into customer behavior, preferences, and purchasing patterns. The dataset is large, containing millions of records, and is too big to be processed on a single machine.

Hadoop Components Used:

- Hadoop Distributed File System (HDFS): For storing large volumes of customer data in a distributed and fault-tolerant manner.
- MapReduce : For processing and analyzing the data in parallel across multiple nodes.

35

Hadoop and Data Science

Steps:

Data Ingestion:
The customer data, which includes information such as purchase history, demographics, and browsing behavior, is stored in HDFS.

Data Cleaning and Preprocessing:
Use MapReduce to clean and preprocess the data. This may involve handling missing values, normalizing data, or transforming it into a suitable format for analysis.

Customer Segmentation:
Apply machine learning algorithms for customer segmentation. Use algorithms like k-means clustering to group customers based on their behavior, preferences, or demographics. This step can be done in parallel across the distributed data using Hadoop's processing capabilities.

Product Recommendation:
Utilize collaborative filtering or other recommendation algorithms to suggest products to customers based on their purchase history or the behavior of similar customers. This can be achieved through MapReduce running on the Hadoop cluster.

36

Hadoop and Artificial Intelligence

- While Hadoop itself is not an AI framework, it provides a distributed storage and processing framework that is well-suited for managing and analyzing massive datasets—something that is often required in AI applications.
- Hadoop can be used as a part of an AI infrastructure, particularly for handling large-scale data processing and storage.

37

Hadoop and Artificial Intelligence

- Here are some ways in which Hadoop can be integrated into AI workflows:
- Data Storage and Processing:
  - Hadoop Distributed File System (HDFS) is designed to handle large volumes of data across multiple nodes. This makes it suitable for storing and managing the extensive datasets commonly used in AI, such as training data for machine learning models.
- Integration with Machine Learning Frameworks:
  - Hadoop can be integrated with various machine learning frameworks and libraries. For example, Apache Mahout, which runs on top of Hadoop, provides scalable machine learning algorithms. Alternatively, Hadoop can be used in conjunction with other machine learning frameworks like Apache Spark MLlib.

38

Hadoop and Artificial Intelligence

- Distributed Computing for Deep Learning:
  - While Hadoop is not commonly used for deep learning, which often requires specialized hardware like GPUs, it can be part of a larger infrastructure for distributed computing. Technologies like TensorFlow or PyTorch can be integrated with Hadoop clusters for distributed deep learning tasks.

39