BDI Exp-1: Case Study on organization using Big Data

Name: Anay Shah

SAP ID: 60004210073

Class: C1

Batch: C12

Date of Submission: 23/02/24

Aim: To study how big data is implemented in an organization.

Theory: Big Data refers to the massive volume of structured and unstructured data generated by organizations, individuals, and machines on a daily basis. Analysing and extracting meaningful insights from this vast amount of data can lead to valuable information that can be used for decision-making, forecasting, and improving various processes. Several theories and concepts surround Big Data:

The concept of Big Data is characterized by the Three Vs—Volume, Velocity, and Variety. Volume pertains to the vast size of data, posing challenges for traditional processing systems. Velocity emphasizes the speed at which data is generated, collected, and processed, crucial in the era of real-time analytics. Variety encompasses different data types, including structured, semi-structured, and unstructured, such as text, images, and videos. Some discussions introduce a fourth V, Veracity, which focuses on the reliability and accuracy of the data, particularly important given the increasing variety and sources of data.
Additionally, the Five W's of Big Data—What, Why, Where, When, and Who— outline essential considerations in data management. It delves into the kind of data collected, the goals behind data collection, the sources and storage locations of data, the timing of data collection, and the responsibilities for data analysis. The Data Lifecycle involves stages from data generation to archival or deletion, including data processing, analysis, and visualization.

The Hadoop ecosystem, featuring a distributed file system and the MapReduce programming model, plays a fundamental role in Big Data processing, enabling parallel processing across distributed clusters. The Lambda Architecture proposes a hybrid approach, combining batch and real-time stream processing for historical and real-time data analysis. Big Data often integrates with machine learning and predictive analytics to identify patterns and automate decision-making processes.

Ethical considerations, including privacy, security, and responsible data use, have gained prominence due to the vast amounts of personal and sensitive data being collected. Data Governance, involving policies, procedures, and standards, ensures data quality, integrity, and compliance with regulations.

Furthermore, the distinction between Data Lakes and Data Warehouses highlights different approaches to storing and managing large volumes of data. Data lakes store raw, unstructured data, offering flexibility for diverse analyses, while data warehouses organize and structure data for specific analytical purposes.

Collectively, these theories and concepts form the foundation for understanding and leveraging Big Data's power in organizations, facilitating informed decision- making and fostering innovation.

How Amazon uses Big Data:

In the ever-evolving landscape of e-commerce, Amazon stands as a beacon of innovation and customer-centricity. Central to its triumph is the strategic and prolific use of big data. Amazon's ability to harness and analyze vast amounts of data has profoundly shaped its operations, enhancing customer experiences, optimizing supply chains, and fostering continuous growth.

Customer Experience Optimization: Amazon's relentless commitment to understanding its customers is powered by big data analytics. The company meticulously analyzes customer behaviors, preferences, and purchase histories. This wealth of information fuels Amazon's recommendation engine, providing users with personalized and relevant product suggestions. This personalized touch not only improves customer satisfaction but also significantly contributes to increased sales and customer loyalty.

Supply Chain Efficiency: Big data analytics plays a pivotal role in optimizing Amazon's complex supply chain. The company deals with an extensive and dynamic inventory, and the ability to predict demand accurately is paramount. Through predictive analytics, Amazon forecasts future demand patterns, strategically manages inventory levels, and ensures that products are available when and where customers need them. This precision in supply chain operations not only reduces costs but also minimizes stockouts and enhances overall operational efficiency.

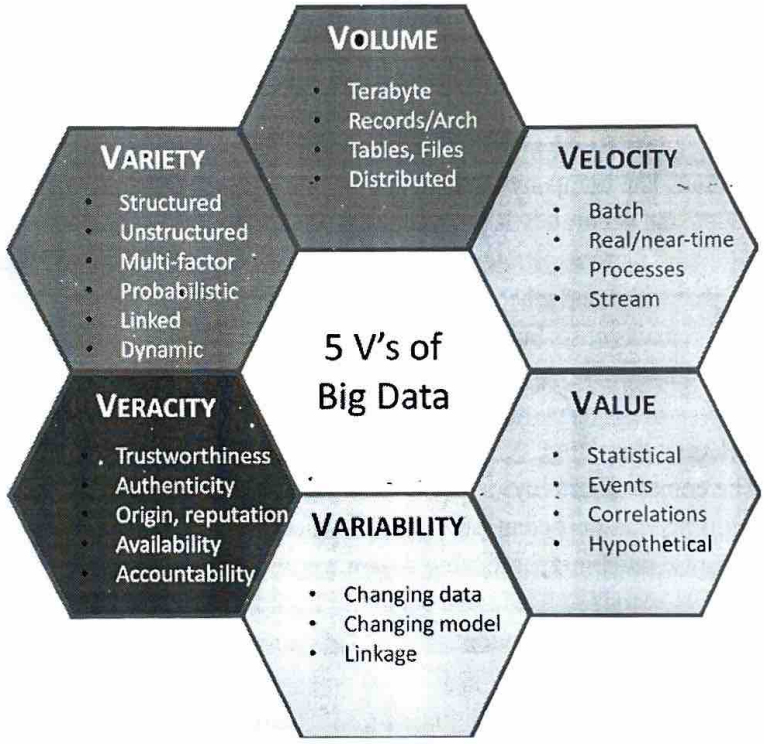
Logistics and Delivery Precision: Amazon's vast logistics network relies heavily on real-time data analytics. The company employs sophisticated algorithms to optimize the routes of delivery vehicles, ensuring timely and efficient deliveries. Predictive analytics also aids in maintenance planning for warehouse machinery, reducing downtime and streamlining the fulfillment process. By leveraging big data, Amazon has transformed the last-mile delivery process, providing customers with rapid and reliable service.

Fraud Detection and Security: In the digital era, safeguarding customer data is paramount. Amazon utilizes big data analytics for robust fraud detection and security measures. Advanced algorithms analyze patterns of user transactions, swiftly identifying anomalies that may indicate fraudulent activities. This proactive approach not only protects customers but also maintains the integrity of the platform, fostering trust among users.

Continuous Innovation and Adaptability: Amazon's success is not solely attributed to its past achievements but to its commitment to continuous improvement. Big data provides Amazon with valuable insights into emerging trends and changing customer behaviors. This adaptability allows the company to stay ahead of the curve, introducing new services and features to meet evolving customer expectations.

Challenges and Future Outlook: While big data has been instrumental in Amazon's success, it is not without challenges. Balancing data utilization with privacy concerns, ensuring data security, and managing the continuous investments in infrastructure are ongoing considerations. However, Amazon's proactive approach and commitment to innovation suggest that these challenges will be met with strategic solutions.

Relevance and Characteristics:



Volume: Amazon, as one of the world's largest e-commerce platforms, manages an extraordinary volume of data daily. Millions of transactions, customer interactions, and an extensive product catalog contribute to an unparalleled scale of data processing. Amazon's ability to handle this immense volume of data is realized through sophisticated storage infrastructure, such as Amazon S3, allowing the company to maintain a comprehensive record of customer behaviors and purchase histories. This vast dataset serves as the bedrock for analytical insights, personalized recommendations, and operational optimizations that drive the company's success.

Velocity: The speed at which transactions occur on Amazon's platform is a defining characteristic requiring real-time processing capabilities. Instantaneous updates to inventory levels as products are purchased, swift processing of customer orders, and responsive services are essential for meeting customer expectations. Amazon's ability to process data in real-time ensures the seamless functioning of its operations, enabling dynamic adjustments to inventory, logistics, and the delivery process. This velocity is foundational to Amazon's commitment to providing a quick, efficient, and customer-centric online shopping experience.

Variety: Amazon's data landscape is characterized by its diversity, encompassing structured, semi-structured, and unstructured data. From transaction records and customer information (structured) to customer reviews (semi-structured) and product images (unstructured), the variety of data types reflects the complexity of Amazon's operations. The ability to manage and analyze this diverse dataset is crucial for understanding customer preferences, conducting sentiment analysis from reviews, and leveraging image recognition for improved product discovery. Amazon's big data infrastructure is adept at handling this variety, facilitating a holistic approach to data analysis.

Veracity: Veracity, representing the reliability and accuracy of data, is paramount for Amazon's operations. Inaccurate data could lead to errors in order fulfillment, disruptions in inventory management, and compromised product recommendations. Amazon invests significantly in maintaining the veracity of its datasets, implementing stringent data quality assurance measures. Ensuring that the data accurately reflects customer interactions, transaction details, and inventory levels is foundational to Amazon's ability to provide reliable and trustworthy services.

Value: Extracting value from data is a central objective for Amazon's big data strategy. The company utilizes advanced analytics to derive actionable insights that enhance customer experiences and optimize operational processes. By understanding customer preferences, Amazon can offer personalized recommendations, improving user satisfaction and driving increased sales. The value derived from data also extends to supply chain optimization, where predictive analytics ensures efficient inventory management. Amazon's focus on extracting meaningful insights from its vast datasets directly contributes to its ability to innovate, adapt to market trends, and maintain a competitive edge in the e-commerce landscape.

Technologies Used:

Distributed Storage (Amazon S3): Amazon S3, or Simple Storage Service, is a fundamental component of Amazon's big data infrastructure. It provides a scalable and secure object storage solution in the cloud. This distributed storage system enables Amazon to efficiently store and

retrieve vast amounts of data, accommodating the immense volume of information generated by its operations.

Computing Power (Amazon EC2): Amazon EC2, Elastic Compute Cloud, offers scalable computing capacity. It allows Amazon to quickly scale its computational resources based on demand, providing the flexibility required for processing large datasets and conducting complex analytics. This scalable computing power is crucial for handling the velocity and variety of data generated on the platform.

Processing Tools (Apache Hadoop and Apache Spark): Amazon utilizes powerful open-source tools like Apache Hadoop and Apache Spark for distributed processing of large datasets. These tools enable efficient batch processing and real-time analytics, supporting Amazon in gaining valuable insights from its diverse and dynamic data sources.

Proprietary Services (Amazon Redshift and AWS Glue): Amazon Redshift, a fully managed data warehouse service, facilitates fast SQL queries for efficient data retrieval and analysis. AWS Glue, a serverless extract, transform, and load (ETL) service, is employed for preparing and loading data. These proprietary services contribute to the seamless integration and management of Amazon's extensive datasets.

How it helped Amazon:

Customer Experience: Big data analytics empowers Amazon to comprehend customer behavior, preferences, and purchase history. This knowledge is leveraged to provide personalized recommendations, enhancing the overall customer experience. The ability to tailor recommendations based on individual preferences contributes to customer satisfaction and loyalty.

Supply Chain Optimization: Big data plays a pivotal role in optimizing Amazon's supply chain. Predictive analytics aids in forecasting demand, managing inventory efficiently, and reducing order fulfillment times. These optimizations contribute to a streamlined supply chain, ensuring that products are available when and where customers need them.

Logistics Efficiency: Real-time data analytics supports Amazon in optimizing logistics and delivery processes. Route optimization for delivery vehicles and predictive maintenance for warehouse machinery contribute to efficient and reliable logistics operations. This results in timely deliveries, a key aspect of Amazon's commitment to customer satisfaction.

Fraud Detection: Big data analytics is instrumental in fraud detection. By analyzing patterns of transactions, Amazon can identify and prevent fraudulent activities, ensuring the security of customer transactions. This proactive approach protects both customers and the integrity of the platform.

Limitations:

Privacy Concerns: Amazon faces challenges related to privacy concerns, particularly due to the vast amount of customer data it holds. Striking a delicate balance between utilizing data for personalized services and respecting user privacy is an ongoing challenge that requires continuous attention and robust privacy policies.

Data Security: Handling sensitive customer information necessitates robust data security measures. Ensuring the confidentiality and integrity of customer data is crucial to maintain trust. Amazon must continually invest in security protocols to safeguard against potential breaches and protect user data.

Continuous Investment: The implementation and maintenance of big data infrastructure require substantial investments. Continuous updates, expansions, and the need for skilled personnel contribute to ongoing costs. Despite the undeniable benefits, managing the financial aspects of a large-scale big data operation demands strategic financial planning and resource allocation.

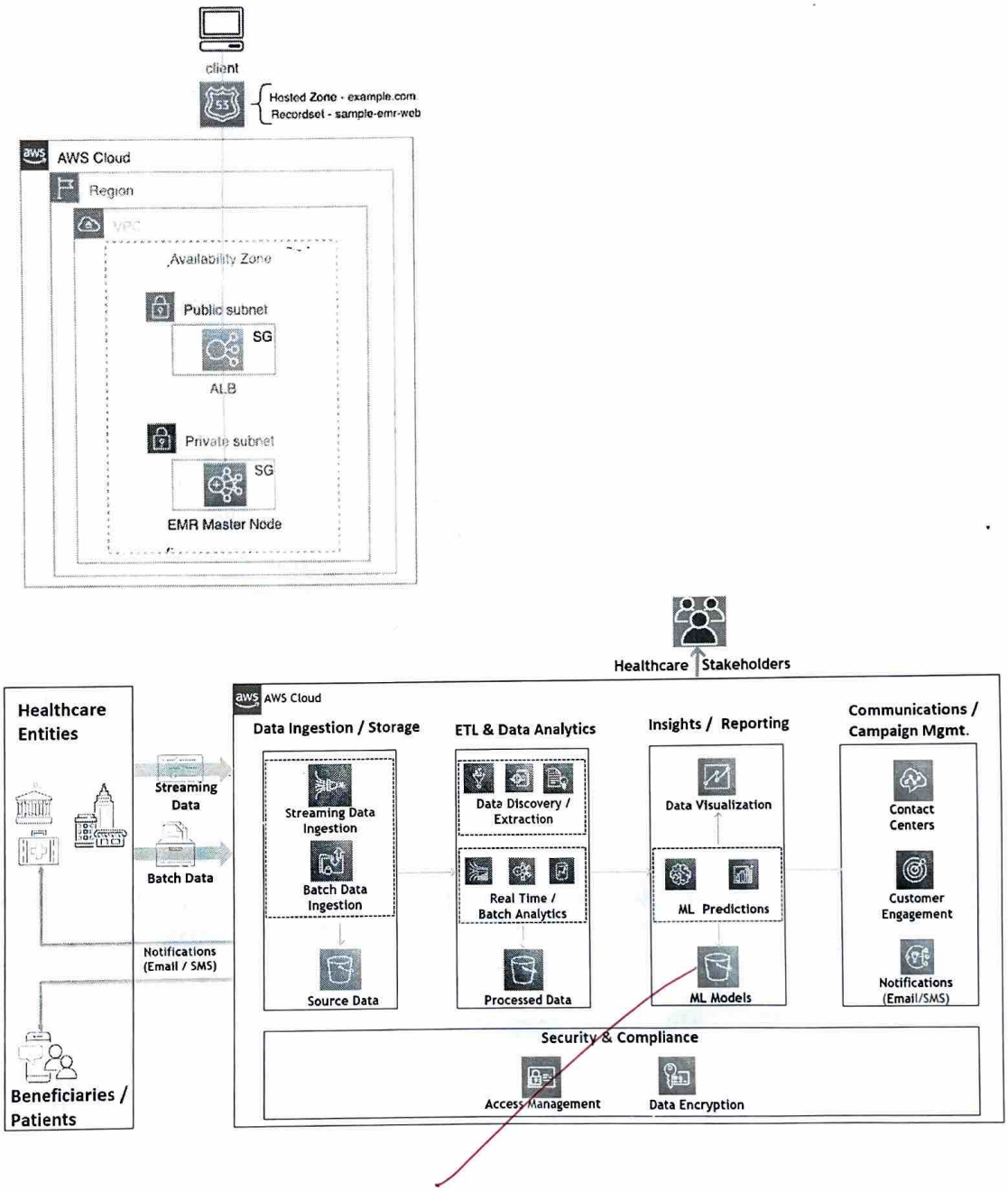
Type of Architecture:

Distributed Computing Architecture: Amazon's big data architecture is intricately designed based on distributed computing principles. At its core, this architecture revolves around the concept of distributing the processing workload across multiple nodes, offering a scalable and flexible solution for handling the immense volume, velocity, and variety of data generated by Amazon's operations. The cornerstone components of this architecture include Amazon S3 for distributed storage and Amazon EC2 for computing power. By leveraging distributed storage, Amazon ensures that large datasets can be efficiently stored and accessed. Simultaneously, the computing power provided by Amazon EC2 allows for parallelized data processing, enabling the execution of complex analytics tasks across the distributed environment. This distributed computing architecture is key to Amazon's ability to manage and analyze vast datasets in a manner that is both scalable and adaptive to the dynamic nature of its operations.

Cloud-Based Services: Amazon's reliance on cloud-based services, notably within the Amazon Web Services (AWS) ecosystem, represents a pivotal aspect of its big data architecture. The

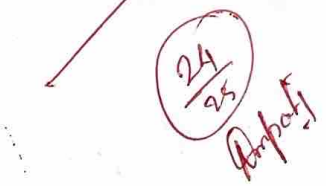
utilization of cloud-based services provides Amazon with on-demand access to computing resources. This cloud-centric approach enhances scalability and cost-effectiveness, allowing Amazon to adapt swiftly to changing computational requirements without the need for significant upfront investments in physical infrastructure. AWS offers a diverse set of services beyond just computing, including storage, databases, analytics, and machine learning, contributing to the overall flexibility and efficiency of Amazon's big data architecture. The cloud-based paradigm aligns with the principles of scalability and accessibility, ensuring that Amazon can harness the computing power needed for its data-intensive operations seamlessly.

Parallel Processing: To tackle the challenge of efficiently analyzing large datasets, Amazon employs parallel processing as a core component of its big data architecture. This methodology involves breaking down analytical tasks into smaller, manageable parts that can be processed simultaneously across multiple computing nodes. By doing so, Amazon achieves two critical objectives: speed and efficiency. Parallel processing enables faster data processing, significantly reducing the time required for complex analytics tasks. Moreover, it enhances efficiency by distributing the computational workload, preventing bottlenecks and optimizing resource utilization. This parallelized approach aligns with the distributed computing architecture, allowing Amazon to derive valuable insights from its vast datasets in a timely and resource-efficient manner. In essence, parallel processing forms a crucial element in the toolkit that empowers Amazon's big data analytics capabilities.


client
Hosted Zone - example.com
Recordset - sample-emr-web
AWS Cloud
Region
VPC
Availability Zone
Public subnet
SG
ALB
Private subnet
SG
EMR Master Node
Healthcare Stakeholders
Healthcare Entities
Streaming Data
Batch Data
Notifications (Email / SMS)
Beneficiaries / Patients
AWS Cloud
Data Ingestion / Storage
Streaming Data Ingestion
Batch Data Ingestion
Source Data
ETL & Data Analytics
Data Discovery / Extraction
Real Time / Batch Analytics
Processed Data
Insights / Reporting
Data Visualization
ML Predictions
ML Models
Communications / Campaign Mgmt.
Contact Centers
Customer Engagement
Notifications (Email/SMS)
Security & Compliance
Access Management
Data Encryption

Conclusion:

In conclusion, Amazon's big data architecture, anchored in distributed computing principles, cloud-based services, and parallel processing, epitomizes a robust and adaptive framework capable of efficiently managing the extensive volume, velocity, and variety of data inherent in the company's operations. The synergy between distributed storage (Amazon S3) and computing power (Amazon EC2) ensures scalability and flexibility, while the reliance on cloud-based services exemplifies a strategic move towards cost-effectiveness and dynamic resource allocation. The pivotal role of parallel processing emerges as a linchpin, facilitating faster data analysis and optimization of computational resources. This holistic architecture underscores Amazon's commitment to harnessing the full potential of its data for personalized customer experiences, supply chain optimization, logistics efficiency, and fraud detection. As a technological cornerstone, Amazon's big data architecture remains integral to the company's ability to navigate and excel in the ever-evolving landscape of e-commerce.



EXPERIMENT NO. 2
Installation of Hadoop on a single node cluster

AIM: Install Hadoop on a Single Node Cluster

THEORY:
Apache Hadoop is an open source framework that is used to efficiently store and process large datasets ranging in size from gigabytes to petabytes of data. Instead of using one large computer to store and process the data, Hadoop allows clustering multiple computers to analyze massive datasets in parallel more quickly.

Hadoop consists of four main modules:

- Hadoop Distributed File System (HDFS) – A distributed file system that runs on standard or low-end hardware. HDFS provides better data throughput than traditional file systems, in addition to high fault tolerance and native support of large datasets.
- Yet Another Resource Negotiator (YARN) – Manages and monitors cluster nodes and resource usage. It schedules jobs and tasks.
- MapReduce – A framework that helps programs do the parallel computation on data. The map task takes input data and converts it into a dataset that can be computed in key value pairs. The output of the map task is consumed by reduce tasks to aggregate output and provide the desired result.
- Hadoop Common – Provides common Java libraries that can be used across all modules.

How Hadoop Works

Hadoop makes it easier to use all the storage and processing capacity in cluster servers, and to execute distributed processes against huge amounts of data. Hadoop provides the building blocks on which other services and applications can be built.

Applications that collect data in various formats can place data into the Hadoop cluster by using an API operation to connect to the NameNode. The NameNode tracks the file directory structure and placement of "chunks" for each file, replicated across DataNodes. To run a job to query the data, provide a MapReduce job made up of many map and reduce tasks that run against the data in HDFS spread across the DataNodes. Map tasks run on each node against the input files supplied, and reducers run to aggregate and organize the final output.

The Hadoop ecosystem has grown significantly over the years due to its extensibility. Today, the Hadoop ecosystem includes many tools and applications to help collect, store, process, analyze, and manage big data. Some of the most popular applications are:

- Spark – An open source, distributed processing system commonly used for big data workloads. Apache Spark uses in-memory caching and optimized execution for fast performance, and it supports general batch processing, streaming analytics, machine learning, graph databases, and ad hoc queries.
- Presto – An open source, distributed SQL query engine optimized for low-latency, ad-hoc analysis of data. It supports the ANSI SQL standard, including complex queries, aggregations, joins, and window functions. Presto can process data from multiple data sources including the Hadoop Distributed File System (HDFS) and Amazon S3.

- Hive – Allows users to leverage Hadoop MapReduce using a SQL interface, enabling analytics at a massive scale, in addition to distributed and fault-tolerant data warehousing.
- HBase – An open source, non-relational, versioned database that runs on top of Amazon S3 (using EMRFS) or the Hadoop Distributed File System (HDFS). HBase is a massively scalable, distributed big data store built for random, strictly consistent, real-time access for tables with billions of rows and millions of columns.
- Zeppelin – An interactive notebook that enables interactive data exploration.

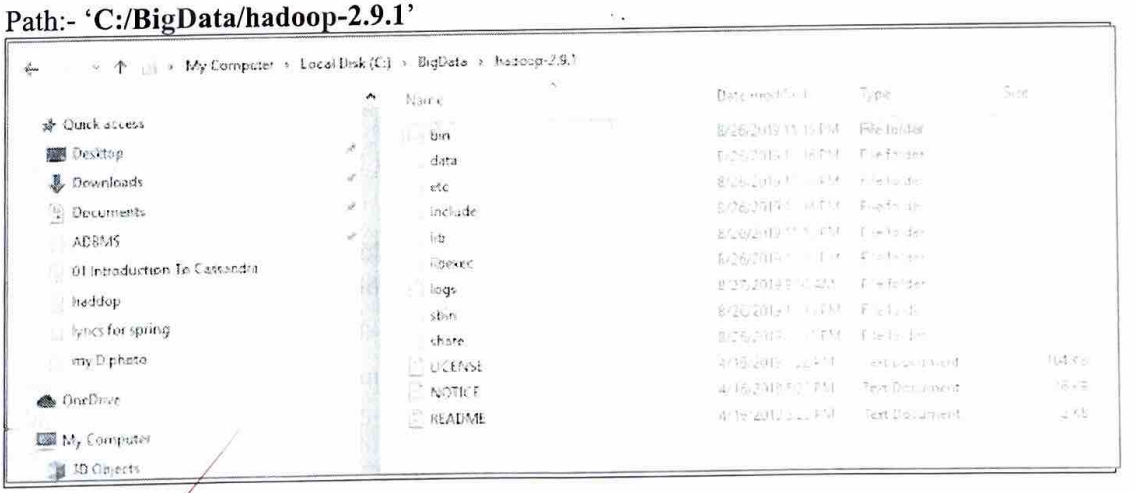
Install Hadoop 2.9.1 on Windows 10

First download the **Hadoop 2.9.1** from the below link.

https://www.apache.org/dyn/closer.cgi/hadoop/common/hadoop-2.9.1/hadoop-2.9.1.tar.gz

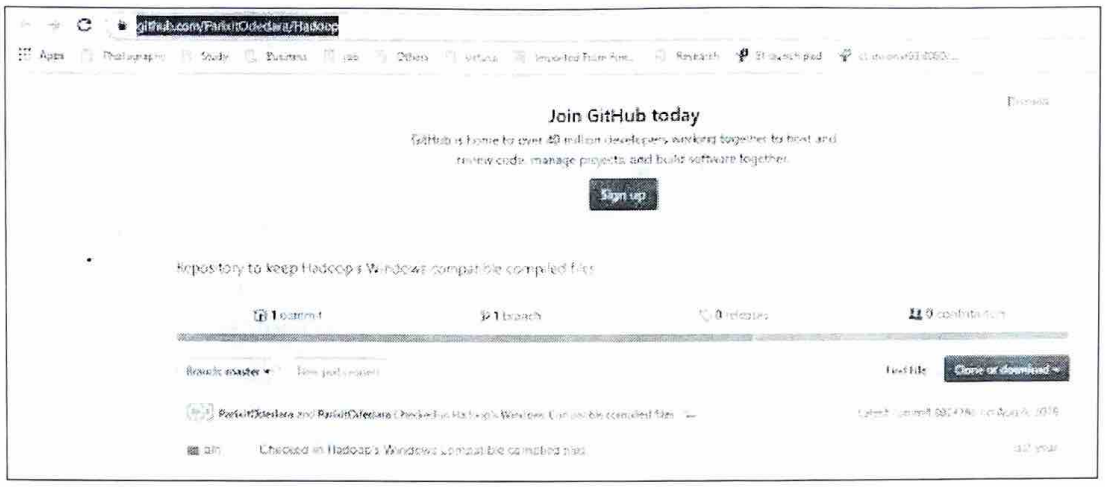
Create a folder path as below and copy the downloaded msi into this folder.

Path:- **'C:/BigData/hadoop-2.9.1'**



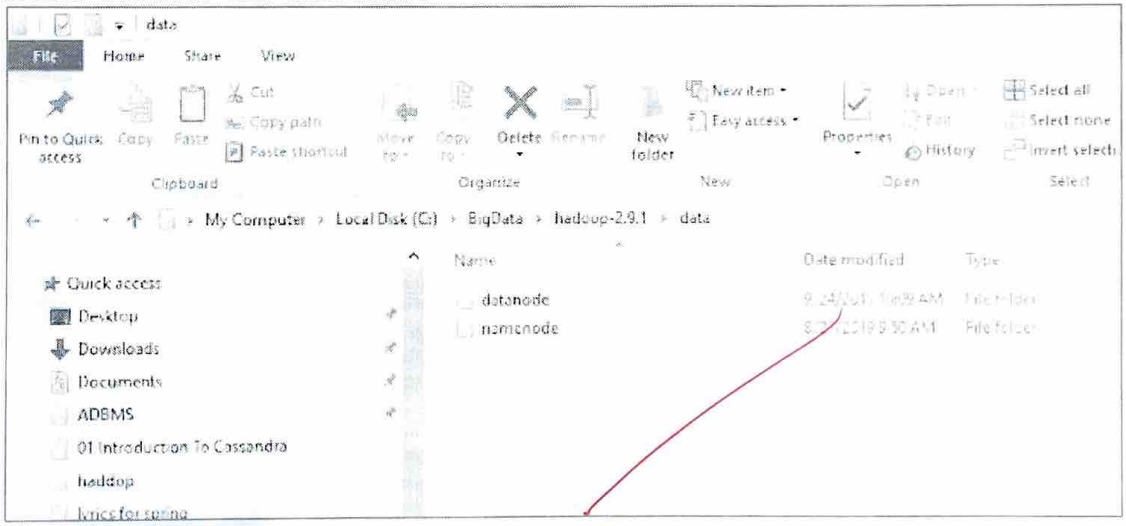
Then download the windows compatible binaries from the git hub repo.

Link:- https://github.com/ParixitOdedara/Hadoop



Extract the zip and copy all the files present under bin folder to C:\BigData\hadoop-2.9.1\bin. Replace the existing files as well.

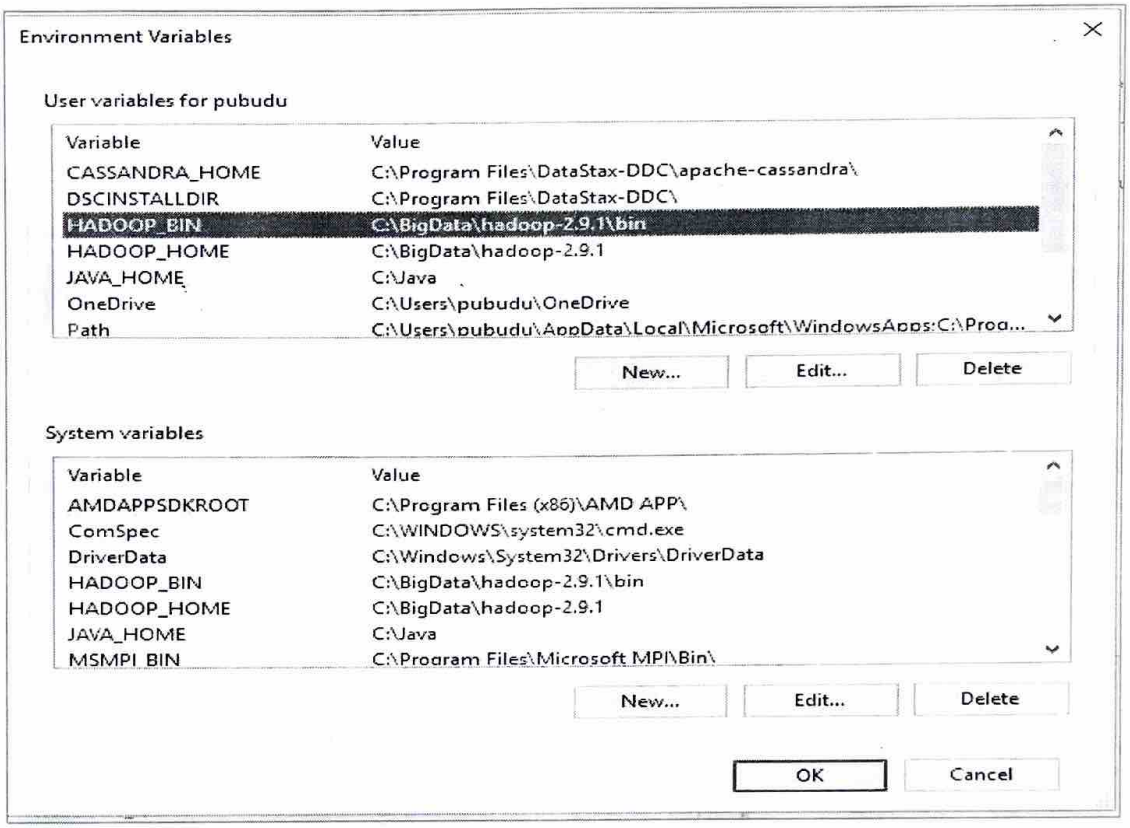
Go to **C:/BigData/3adoop-2.9.1** and create a folder '**data**'. Inside the 'data' folder create two folders '**datanode**' and '**namenode**'.



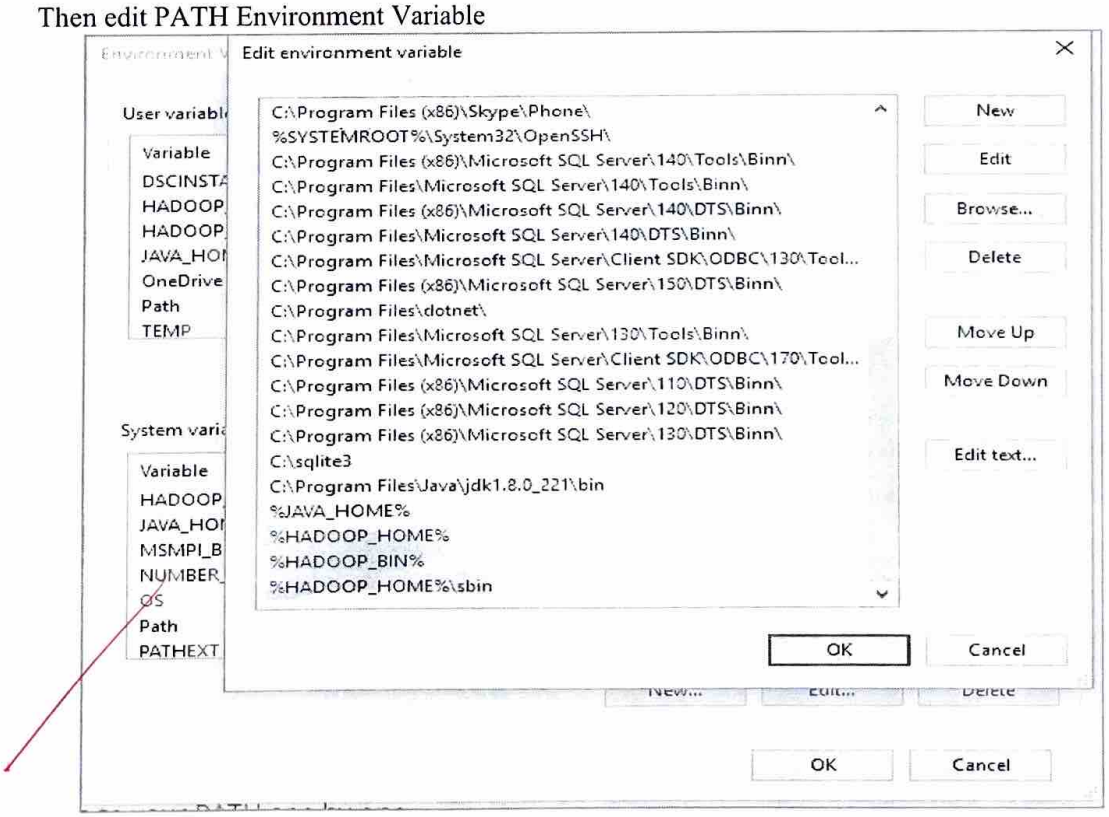
Then Set Hadoop Environment Variables

HADOOP_HOME="C:\BigData\hadoop-2.9.1"
HADOOP_BIN="C:\BigData\hadoop-2.9.1\bin"
JAVA_HOME=<JDK installation location>"

To set these variables, go to My Computer or This PC. Right click --> Properties --> Advanced System settings --> Environment variables. Click New to create a new environment variables.



Then edit PATH Environment Variable

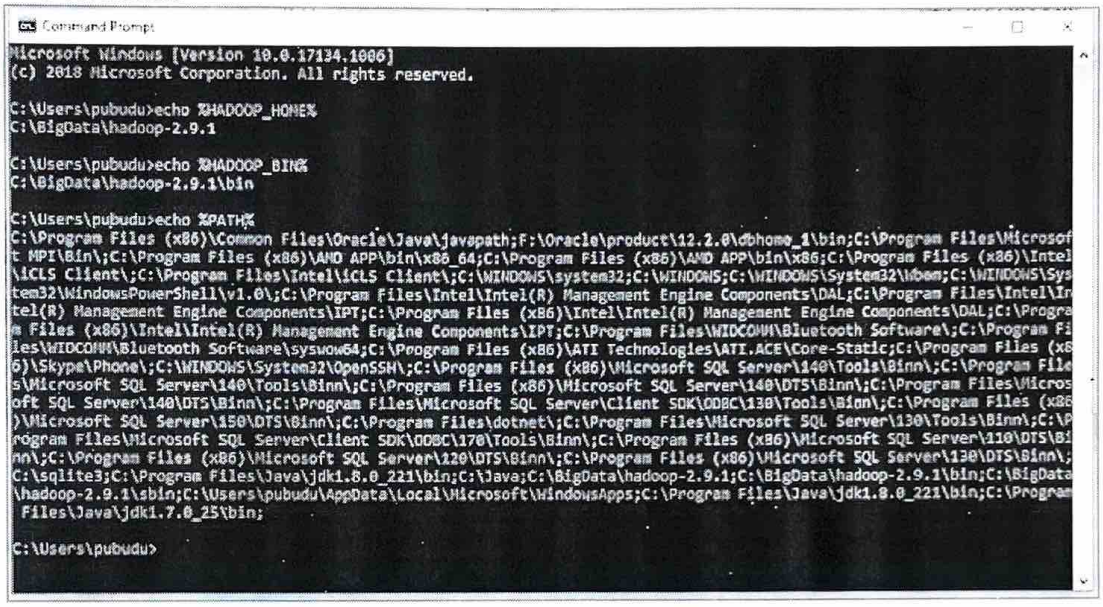


To validate the above setting, **open new cmd** and check the output.

echo %HADOOP_HOME%

echo %HADOOP_BIN%

echo %PATH%



To configure the Hadoop on windows we have to edit below mention files in the extracted location.

1. hadoop-env.cmd
2. core-site.xml
3. hdfs-site.xml
4. mapred-site.xml
5. yarn-site.xml

Edit hadoop-env.cmd

File location:-C:\BigData\hadoop-2.9.1\etc\hadoop\hadoop-env.cmd

Need to add:-

```
set HADOOP_PREFIX=%HADOOP_HOME%
set HADOOP_CONF_DIR=%HADOOP_PREFIX%\etc\hadoop
set YARN_CONF_DIR=%HADOOP_CONF_DIR%
set PATH=%PATH%;%HADOOP_PREFIX%\bin
```

```
@rem        potential for a symlink attack.
set HADOOP_PID_DIR=%HADOOP_PID_DIR%
set HADOOP_SECURE_DN_PID_DIR=%HADOOP_PID_DIR%

@rem A string representing this instance of hadoop. %USERNAME% by default.
set HADOOP_IDENT_STRING=%USERNAME%
set HADOOP_PREFIX=%HADOOP_HOME%
set HADOOP_CONF_DIR=%HADOOP_PREFIX%\etc\hadoop
set YARN_CONF_DIR=%HADOOP_CONF_DIR%
set PATH=%PATH%;%HADOOP_PREFIX%\bin
```

Edit core-site.xml

File Location:- C:\BigData\hadoop-2.9.1\etc\hadoop\core-site.xml

Need to add:-content within <configuration> </configuration> tags.

```
<configuration>
 <property>
  <name>fs.default.name</name>
  <value>hdfs://0.0.0.0:19000</value>
 </property>
</configuration>
```

```
    limitations under the License. See accompanying LICENSE file.
  -->

  <!-- Put site-specific property overrides in this file.  >

<configuration>
<property>
      <name>fs.default.name</name>
      <value>hdfs://0.0.0.0:19000</value>
    </property>
</configuration>
```

Edit hdfs-site.xml

File Location:- C:\BigData\hadoop-2.9.1\etc\hadoop\hdfs-site.xml.

Need to add;- below content within <configuration> </configuration> tags.

```
<configuration>
 <property>
  <name>dfs.replication</name>
  <value>1</value>
 </property>
 <property>
  <name>dfs.namenode.name.dir</name>
  <value>C:\BigData\hadoop-2.9.1\data\namenode</value>
 </property>
 <property>
```

```
    <name>dfs.datanode.data.dir</name>
    <value>C:\BigData\hadoop-2.9.1\data\datanode</value>
  </property>
</configuration>
```

Edit mapred-site.xml

File location:- Open C:\BigData\hadoop-2.9.1\etc\hadoop\mapred-site.xml

Need to add:- below content within <configuration> </configuration> tags. If you don't see mapred-site.xml then open mapred-site.xml.template file and rename it to mapred-site.xml

```
 <configuration>
  <property>
    <name>mapreduce.job.user.name</name>
    <value>%USERNAME%</value>
  </property>
  <property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
  <property>
    <name>yarn.apps.stagingDir</name>
    <value>/user/%USERNAME%/staging</value>
  </property>
  <property>
    <name>mapreduce.jobtracker.address</name>
    <value>local</value>
  </property>
</configuration>
```

Editing yarn-site.xml

Right click on the file, select edit and paste the following content within <configuration> </configuration> tags.

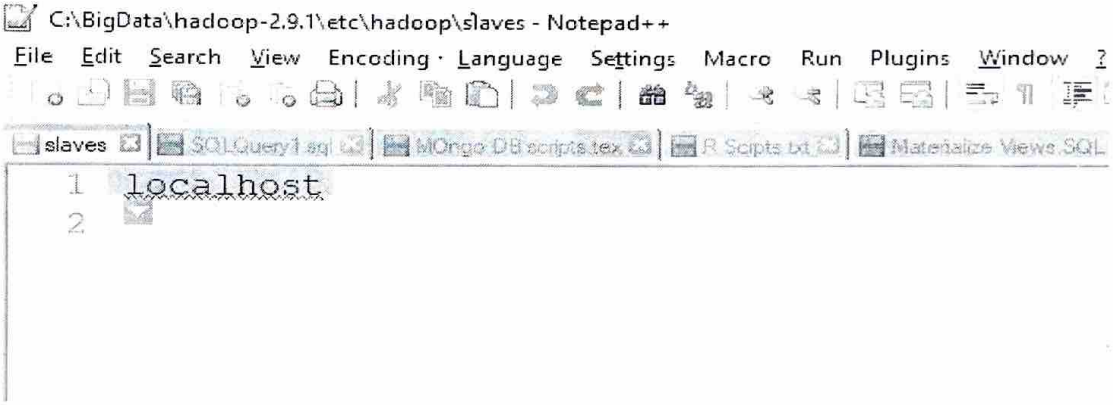
Note:- Below part already has the configuration tag, we need to copy only the part inside it.

```
<configuration>
 <property>
  <name>yarn.nodemanager.aux-services</name>
  <value>mapreduce_shuffle</value>
 </property>
 <property>
```

```
<name>yarn.nodemanager.auxservices.mapreduce.shuffle.class</name>
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
<!-- Site specific YARN configuration properties --></configuration>
```

Additional Configuration:-

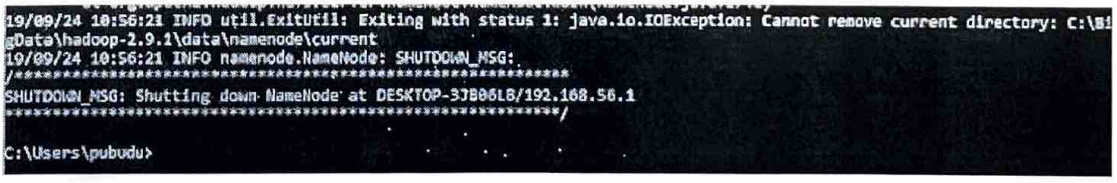
Check if C:\BigData\hadoop-2.9.1\etc\hadoop\slaves file is present, if that file not available create the file called slave and insert locahost as below.



Node formatting

To format the node, **open the cmd** and execute the below command.

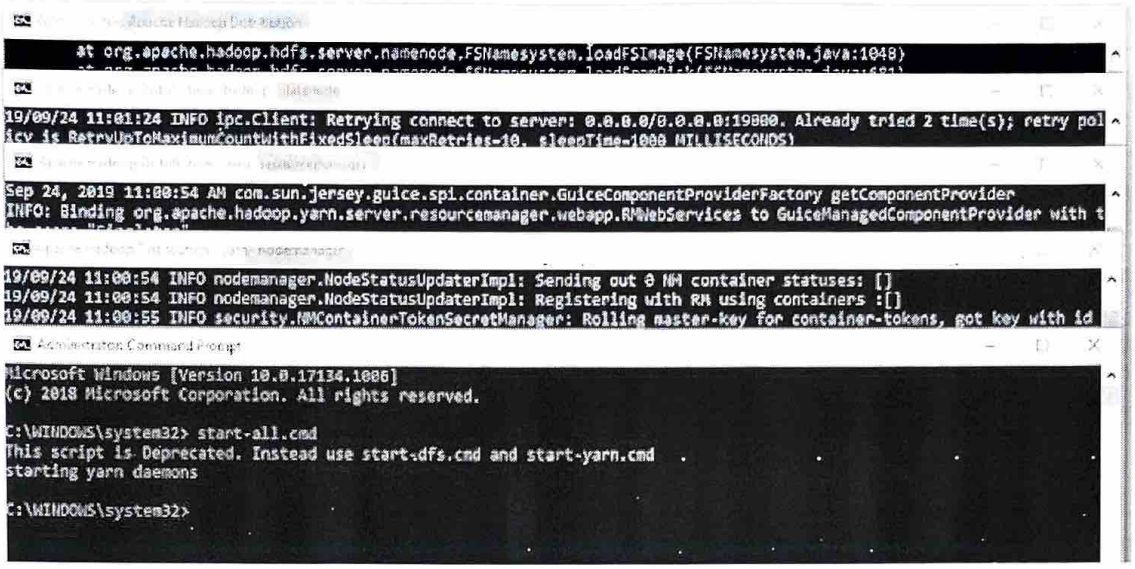
hadoop namenode -format



To enable the Hadoop open the **CMD as Administrator** and type below command

start-all.cmd

It will open 4 new windows cmd terminals for 4 daemon processes, namely namenode, datanode, nodemanager, and resourcemanager.



Then you have succesfully install the hadoop 2.9.1 on windows platform.

Now you can access all the Hadoop components via web urls.

To access Resource Manager go to http://localhost:8088 from your web browser.

To access Node Manager go to http://localhost:8042 from your web browser.

To access Name Node go to http://localhost:50070 from your web browser.

Overview '0.0.0.0:19000' (active)

Started:	Tue Sep 24 12:50:12 +0530 2019
Version:	2.9.1, re30710aea4e6e55e69372929106cf119af06fd0e
Compiled:	Mon Apr 16 15:03:00 +0530 2018 by root from branch-2.9.1
Cluster ID:	CID-dff52b8b-b137-4888-bdb8-982a44236747
Block Pool ID:	BP-1603389017-192.168.56.1-1569309564864

Summary

To access Data Node go to http://localhost:50075 from your web browser.

DataNode on 192.168.56.1:50010

Cluster ID:	CID-dff52b8b-b137-4888-bdb8-982a44236747
Version:	2.9.1

Block Pools

Namenode Address	Block Pool ID	Actor State	Last Heartbeat	Last Block Repo
0.0.0.0:19000	BP-1603389017-192.168.56.1-1569309564864	RUNNING	2s	a few seconds

CONCLUSION: We have successfully installed Hadoop 2.9.1 on Windows 10.

24/25 Gunjal