Father Saturnino Urios University
Computer Studies Program
Butuan City

IT 374 – Integrative Programming and Technologies 2

FINAL EXAM: HADOOP SOFTWARE

(NameNode and DataNode)

Submitted by:

Rena Jane B. Opong

Submitted to:

Mr. Dominic B. Guiritan

I. Introduction

Hadoop is an open source, Java based framework used for storing and processing big data. The data is stored on inexpensive commodity servers that run as clusters. Its distributed file system enables concurrent processing and fault tolerance. Developed by Doug Cutting and Michael J. Cafarella, Hadoop uses the MapReduce programming model for faster storage and retrieval of data from its nodes. The framework is managed by Apache Software Foundation and is licensed under the Apache License 2.0.

For years, while the processing power of application servers has been increasing manifold, databases have lagged behind due to their limited capacity and speed. However, today, as many applications are generating big data to be processed, Hadoop plays a significant role in providing a much-needed makeover to the database world.

From a business point of view, too, there are direct and indirect benefits. By using open-source technology on inexpensive servers that are mostly in the cloud (and sometimes on-premises), organizations achieve significant cost savings.Additionally, the ability to collect massive data, and the insights derived from crunching this data, results in better business decisions in the real-world—such as the ability to focus on the right consumer segment, weed out or fix erroneous processes, optimize floor operations, provide relevant search results, perform predictive analytics, and so on.

How Hadoop Improves on Traditional Databases

Hadoop solves two key challenges with traditional databases:

1. Capacity: Hadoop stores large volumes of data.

By using a distributed file system called an HDFS (Hadoop Distributed File System), the data is split into chunks and saved across clusters of commodity servers. As these commodity servers are built with simple hardware configurations, these are economical and easily scalable as the data grows.

2. Speed: Hadoop stores and retrieves data faster.

Hadoop uses the MapReduce functional programming model to perform parallel processing across data sets. So, when a query is sent to the database, instead of handling data sequentially, tasks are split and concurrently run across distributed servers. Finally, the output of all tasks is collated and sent back to the application, drastically improving the processing speed.

II. Installing a Hadoop software

Step 1: https://hadoop.apache.org/releases.html

Step 2: choose 3.3.0 and click the binary

Step 3: After clicking The binary click the

https://downloads.apache.org/hadoop/common/hadoop-3.3.0/hadoop-3.3.0.tar.gz

Step 4: Before you install the hadoop you must install the JDK. Link:
https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html

Step 5: Install the JDK First Then Move The Jdk Folder to local C.

Step 6: Now go to Environment variable for the setup. Click the Environment Variable then
click the New button then Input the Variable Name:(it depends on you) Variable Value:(the
location of file/ Path of the folder).

Step 7: In System Variable Click the Path Column then paste the (Path of the file) also.

Step 8: Do this in CMD: cd Program Files\Java\jdk-13.0.2\bin then type javac.

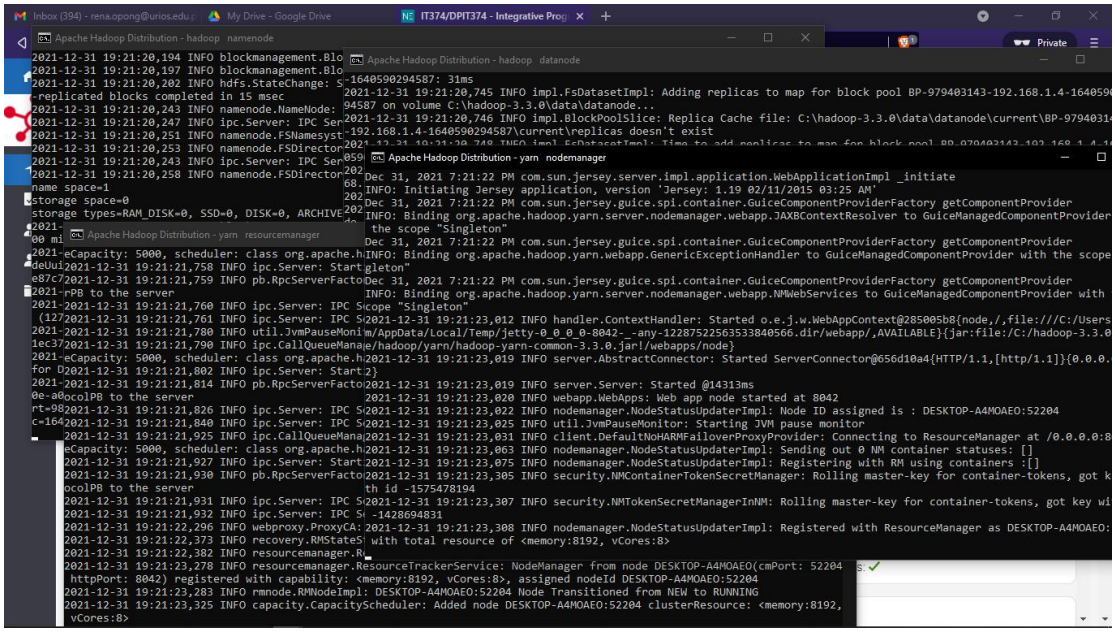
Step 9: After You installed the JDK u must install now the hadoop file. First you need to
extract the file then after you extracted the Zip file. now move the file to the local C.

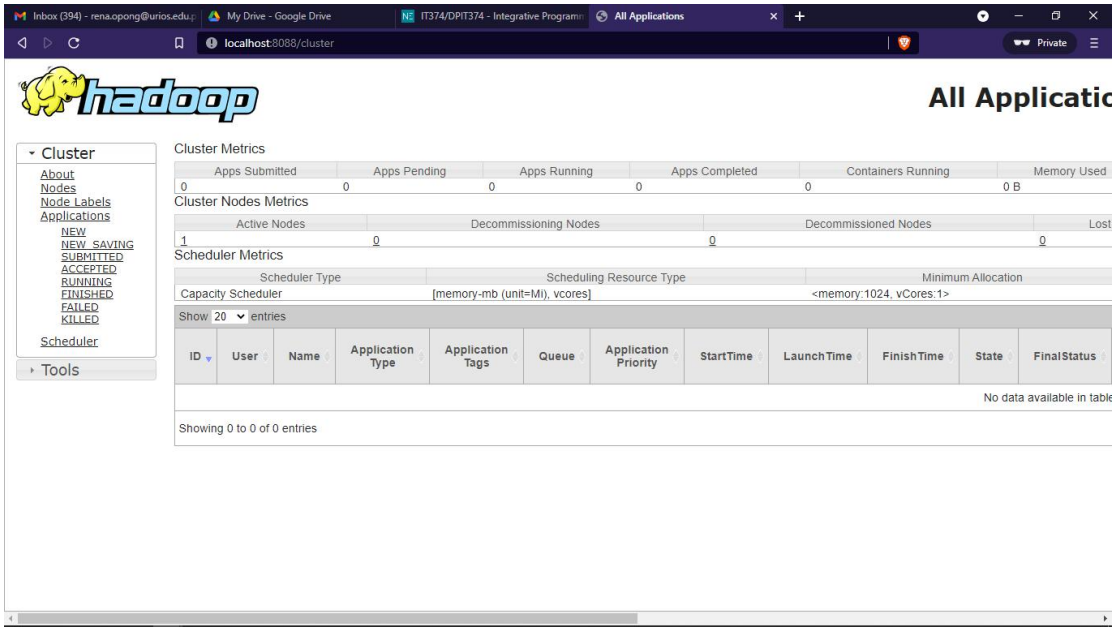
Step 10: After Extracted the Zip file of hadoop. Now go to ETC folder then find the.
-- core-site-- xml mode
-- hadoop-env-- cmd mode
-- mapred-site -- xml mode
-- yarn site -- xml mode
-- hdfs site - xml mode

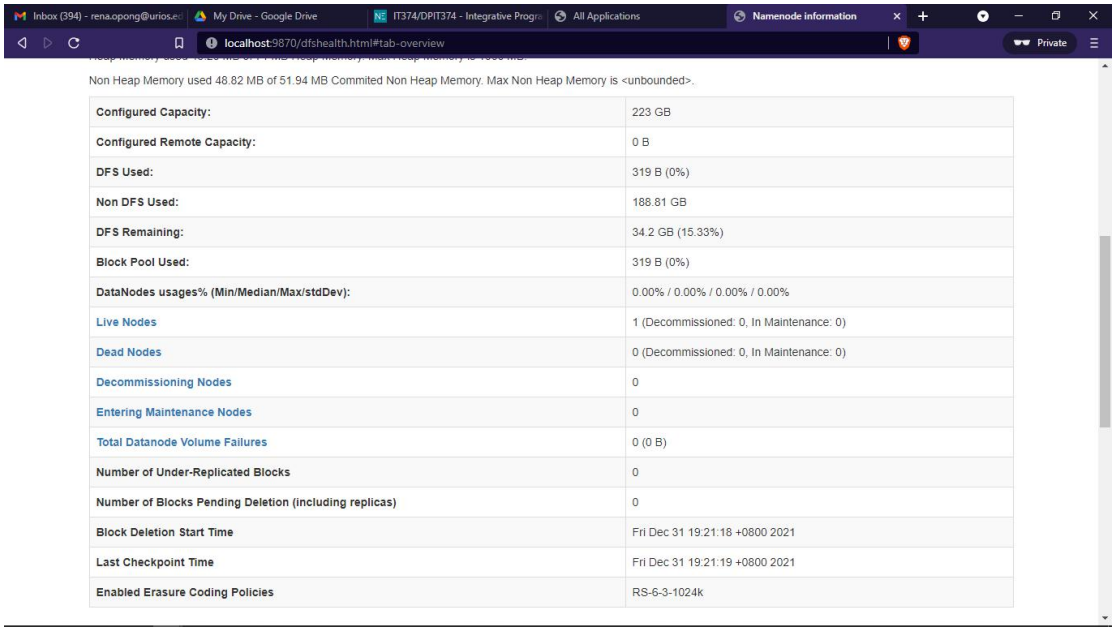
Step 11: Create folder inside the hadoop folder (data).

Step 12: Open the folder that you created then create folder again for (datanode and
namenode).

III. Documentation







Configured Capacity:	223 GB
Configured Remote Capacity:	0 B
DFS Used:	319 B (0%)
Non DFS Used:	188.81 GB
DFS Remaining:	34.2 GB (15.33%)
Block Pool Used:	319 B (0%)
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0, In Maintenance: 0)
Dead Nodes	0 (Decommissioned: 0, In Maintenance: 0)
Decommissioning Nodes	0
Entering Maintenance Nodes	0
Total Datanode Volume Failures	0 (0 B)
Number of Under-Replicated Blocks	0
Number of Blocks Pending Deletion (including replicas)	0
Block Deletion Start Time	Fri Dec 31 19:21:18 +0800 2021
Last Checkpoint Time	Fri Dec 31 19:21:19 +0800 2021
Enabled Erasure Coding Policies	RS-6-3-1024k

IV. Importance of Hadoop

Big Data masters are facing serious challenges in storing, cleaning, and analyzing colossal data sets economically in real time. Increasingly, enterprises are looking for data solutions to turn analysis into insights for making solid decisions. For that, they need data professionals who know how to convert BIG DATA into BIG OPPORTUNITIES. Excerpts from a speech delivered by Hadoop founder Doug cutting at Cloud factory in Banff, Canada are listed below. Clearly the future of big data is Hadoop. In the future we'll be able to store and process more data than we can now. The enterprises that will do best are those that will best leverage Hadoop. Not only can you afford to store more data in the future, but in many ways, you can't afford not to. Hadoop will get better. More and more data will move out of silo systems and into central systems that provide a variety of tools running on a variety of data sets … essentially an 'enterprise data hub.

Stored Data to Data Node

Instruction: If you are successful connected the server also if your Data node and Name node isn't shutdown it means you are successful connected to Hadoop. By inserting data inside the Data node you must do this command.

Step 1: C:\Users\artam>hadoop fs –mkdir /sample_dir
Step 2: C:\Users\artam>hadoop fs –put [File_location_that_you_want_to_stored] /sample_dir
Step 3: C:\Users\artam>hadoop fs –cat /sample_dir/[File_location_that_you_want_to_stored]

Explanation:
Step 1: You are making a file directory in DATANODE.
Step 2: Inserting File to the sample_dir where sample_dir is a folder in DATANODE.
Step 3: Display Data from DATANODE

Other Command:

* C:\Users\artam> hadoop dfsadmin -safemode leave

* C:\Users\artam> hadoop dfsadmin –safemode enter

* C:\Users\artam> hadoop fs -rm -r /input_dir/sample.txt

* C:\Users\artam> hadoop fs -rm -r /input_dir

Explanation:
* You are leaving in a Safe Mode.
* You are Entering in a Safe mode

* Deleting file/data inside the DATANODE where you pointing to inside the input_dir.

* Deleting input_dir where input_dir is a folder of DATANODE.