
Lab 01: A Gentle Introduction to Hadoop

CSC14118 Introduction to Big Data 20KHMT1

All in

2023-02-17

Contents

Task progression	2
Self-reflection	3
1 Setting up Single-node Hadoop Cluster	4
1.1 Member 1: 20120090 - Nguyen The Hoang	4
1.1.1 Running in Local (Standalone Mode)	6
1.1.2 Running in Pseudo-Distributed Mode	7
1.1.3 Running in Pseudo-Distributed Mode with YARN	9
1.1.4 Accessing the Web interfaces of running Hadoop process	9
1.2 Member 2: 20120011 - Nguyen Hoang Huy	12
1.3 Member 3: 20120030 - Nguyen Thien An	14
1.4 Member 4: 20120165 - Hong Nhat Phuong	22
2 Introduction to MapReduce	24
3 Running a warm-up problem: Word Count	26
3.1 Member 1: 20120011 - Nguyen Hoang Huy	26
3.2 Member 2: 20120030 - Nguyen Thien An	27
3.3 Member 3: 20120090 - Nguyen The Hoang	30
3.4 Member 4: 20120165 - Hong Nhat Phuong	34
4 Bonus	37
4.1 Bad Relationship	37
5 References	38

- Author: [All in] group
- Date: 21/03/2023
- Subtitle: CSC14118 Introduction to Big Data 20KHMT1
- Language: English

Task progression

No. of task	% completed
1	100%
2	100%
3	100%
4.1	100%
4.2	15%

Self-reflection

- Advantages
 - All members in the group are supportive, are eager to learn, try to solve the problems/bugs as best as possible.
 - Groups have set up the plan, communication devices, document, working directory, etc. to work efficiently, as soon as this project started.
 - Everyone helped each other to solve the problems.
 - The group has learned and understood more about the workflow, components of Hadoop; knowed some tricky point in Hadoop; be ready for next projects.
- Disadvantage
 - Due to large amount of works in this project and other subjects, the quality of conducting plan has fallen off gradually.
 - We have not set up the real distributed Hadoop system. We will study this topics as soon as possible. As a result, we have not achieved the bonus scores fully.

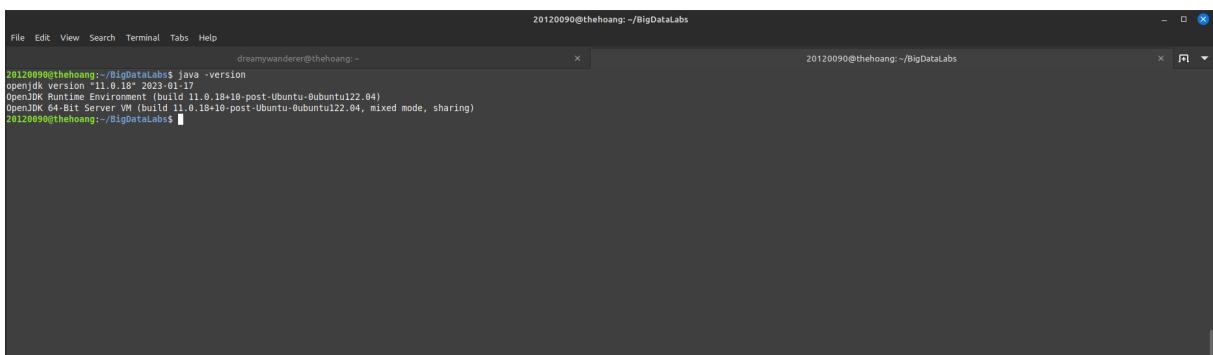
1 Setting up Single-node Hadoop Cluster

Each member has completed setting up a Single node cluster on their local machine, as described in supplied tutorial. Also, this process is captured in screenshots.

1.1 Member 1: 20120090 - Nguyen The Hoang

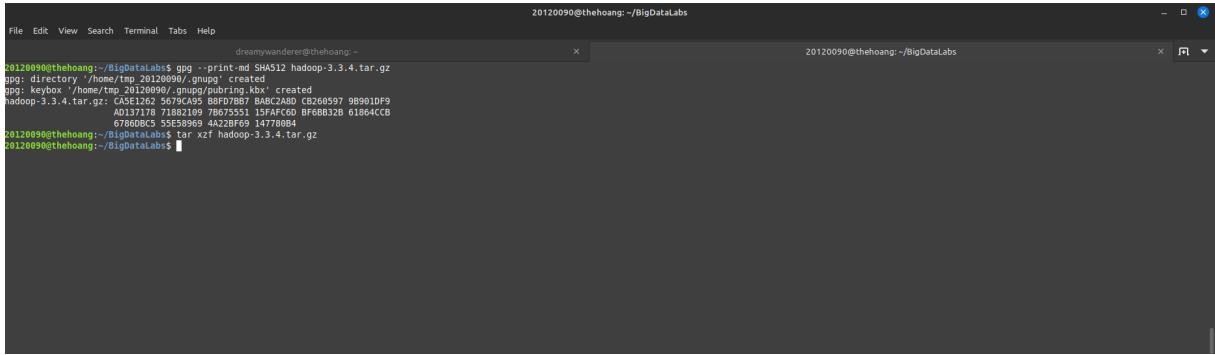
This member will present detail of the steps setting up Single-node Hadoop cluster. Other members will only show result in their machines. Some steps or result may be different between machines of each member, due to some difference of software/hardware environment and tutorials that they followed. But without loss of generality, the main steps are same.

This Single-node Hadoop Cluster is installed in the Linux OS, Mint distro.



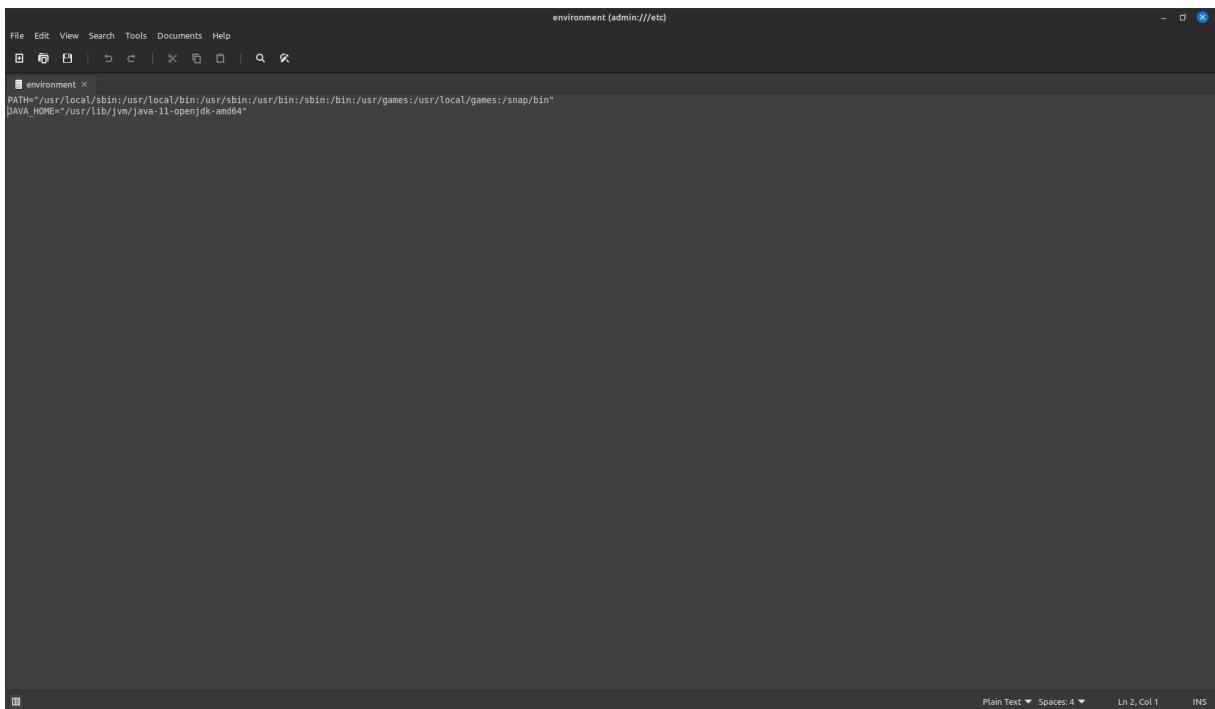
```
File Edit View Search Terminal Tabs Help
dreamywanderer@thehoang:~
20120090@thehoang:~/BigDataLabs$ java -version
openjdk version "11.0.18" 2023-01-17
OpenJDK Runtime Environment (build 11.0.18+10-post-Ubuntu-0ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.18+10-post-Ubuntu-0ubuntu122.04, mixed mode, sharing)
20120090@thehoang:~/BigDataLabs$
```

Figure 1.1: Step 1: Make sure that a suitable version of Java installed



The screenshot shows two terminal windows side-by-side. The left terminal window has the title 'File Edit View Search Terminal Tabs Help' and the prompt 'dromwanderer@thehoang: ~'. It contains the command: 'gpg --print-md SHA512 hadoop-3.3.4.tar.gz'. The right terminal window has the title '20120090@thehoang: ~/BigDataLabs' and also contains the same command. Both terminals show the output of the gpg command, which includes the creation of a keybox and the verification of the file's integrity.

Figure 1.2: Step 2: Download and extracted the Hadoop packaged file from the Apache Hadoop releases page



The screenshot shows a single terminal window with the title 'environment (admin:///etc)'. The prompt is 'environment'. The terminal displays the contents of the '/etc/environment' file, which contains the following lines: 'PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"' and 'JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"'. At the bottom right of the terminal window, there are status indicators: 'Plain Text ▾ Spaces: 4 ▾ Ln 2, Col 1 INS'.

Figure 1.3: Step 3: Set up the environment variable JAVA_HOME in the file etc/environment which points to the binary folder of Java, so that Hadoop can use Java when compiling MapReduce programs and execute daemons/processes.

```

20120090@thehoang:~$ echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk-amd64
20120090@thehoang:~$ export HADOOP_HOME=
20120090@thehoang:~$ bin/dreamywanderer ~gnats -list -news -root/ -sssd -systemd-network/ -usbmux
-apt ~colorl ~flatpak ~hplip ~ip ~nm-openvpn/ -rtkit/ -sync/ -systemd-resolve/ -uucp
~avahi ~cups-pk-helper ~refresh ~mail/ ~nobody ~saned/ -sys/ -systemd-timesync/ -uuid/
~avahi-autopid/ ~daemon/ ~games/ ~kernooops/ ~man/ ~proxy/ -speech-dispatcher/ -syslog -tcpdump
~backup/ ~dnsmasq/ ~geoclue/ ~lightdm ~messagebus ~pulse -sshd -systemd-coredump/ -tss/
~dnsutils/ ~gnome/ ~ibus ~lightdm ~pulse -sshd -systemd-coredump/ -tss/
20120090@thehoang:~$ export HADOOP_HOME=~/BigDataLabs/hadoop-3.3.4/
20120090@thehoang:~$ SHADOOP_HOME=/BigDataLabs/hadoop-3.3.4
20120090@thehoang:~$ export PATH=$PATH:$HADOOP_HOME/bin:$SHADOOP_HOME/sbin
20120090@thehoang:~$ echo PATH
PATH=...
20120090@thehoang:~$ echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/bin:/usr/games:/snap/bin:/home/tmp_20120090/BigDataLabs/hadoop-3.3.4/bin:/home/tmp_20120090/BigDataLabs/hadoop-3.3.4/sbin:/home/tmp_20120090/BigDataLabs/hadoop-3.3.4
20120090@thehoang:~$ 

```

Figure 1.4: Step 4: Set up the environment variable HADOOP_HOME in the file `~/.bashrc` (or by using `export` temporarily) which points to the extracted folder of Hadoop, so that it is more convenient to call Hadoop command after that.

```

20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ hadoop version
Hadoop 3.3.4
Source code repository https://github.com/apache/hadoop.git -r a585a73c3e02ac62350c136643a5e7f6995a3db
Compiled by stevel on 2022-07-29T12:32Z
Compiled with protoc 3.7.1
From source with checksum fb99d8918a7ba5ab430d1af858f6ec
This command was run using /home/tmp_20120090/BigDataLabs/hadoop-3.3.4/share/hadoop/common/hadoop-common-3.3.4.jar
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ Usage: hadoop [OPTIONS] SUBCOMMAND [SUBCOMMAND OPTIONS]
or: hadoop [OPTIONS] CLASSNAME [CLASSNAME OPTIONS]
where CLASSNAME is a user-provided Java class
OPTIONS is none or any of:
buildpaths      attempt to add class files from build tree
--config dir    Hadoop config directory
--debug         turn on shell script debug mode
--help          usage information
hostnames list[,of,host,names] hosts to use in slave mode
hosts filename   list of hosts to use in slave mode
loglevel level   set the log4j level for this command
workers         turn on worker mode
SUBCOMMAND is one of:

```

Figure 1.5: Step 5: Check that Hadoop runs by typing `hadoop version`

1.1.1 Running in Local (Standalone Mode)

```

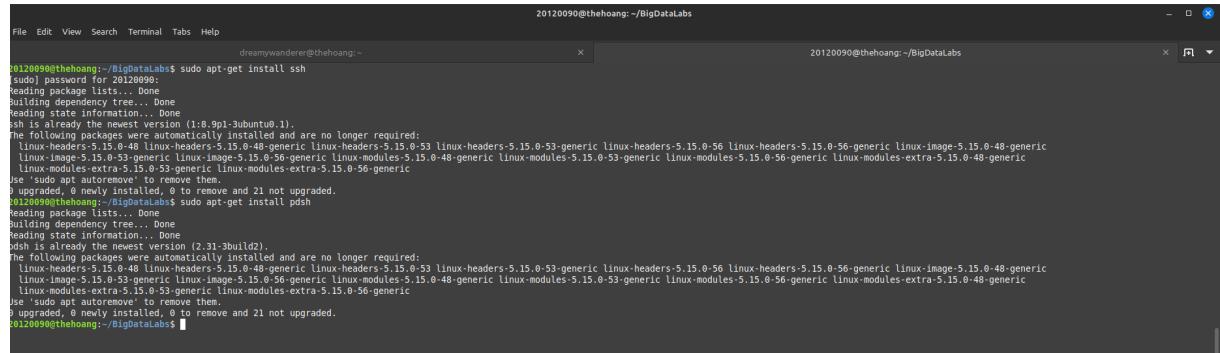
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ hadoop fs -put /tmp/testfile /output
Combine output records=0
Reduce input groups=1
Reduce shuffle bytes=25
Reduce input records=1
Reduce output records=1
Spilled Records=2
Shuffled Maps =1
Failed Shuffles=0
Merged Map Outputs=1
GC Time elapsed (ms)=7
Total committed heap usage (bytes)=719323136
Shuffle Errors
Bad Src=0
CONNECTION=0
IO ERROR=0
WRONG LENGTH=0
WRONG MAP=0
WRONG REDUCE=0
File Input Format Counters
Bytes Read=123
File Output Format Counters
Bytes Written=23
DFS Admin Counters
1 dfsadmin
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ dir
bin include input lib libexec LICENSE-binary licenses-binary LICENSE.txt NOTICE-binary NOTICE.txt output README.txt sbin share
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ cat output/*
1

```

Figure 1.6: Step 6: Keep the default configure file of Hadoop. Conduct the Standalone Operation as instructed in ¹, the above picture shows the result in the output folder

1.1.2 Running in Pseudo-Distributed Mode

Step 7: Change the configure files: core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml as instructed in² and [^b] to set up for Pseudo-Distributed Mode

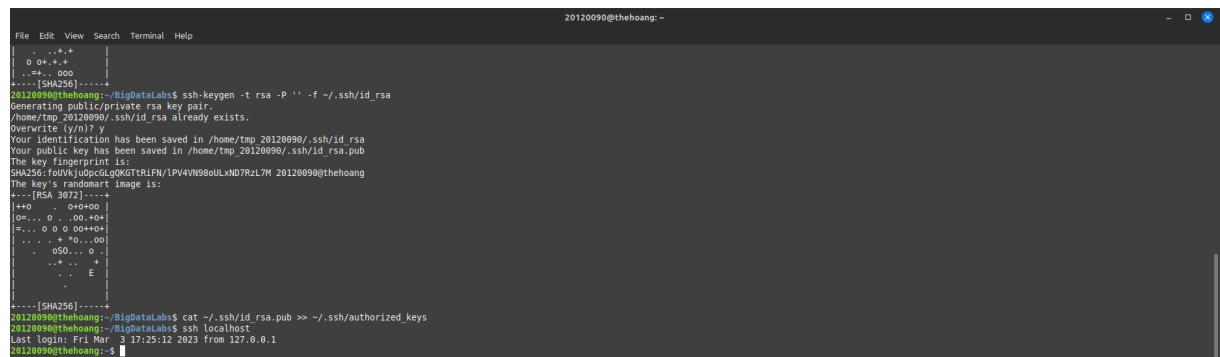


```
File Edit View Search Terminal Tabs Help
dreamwanderer@thehoang:~
```

```
20120090@thehoang:~/BigDataLabs
```

```
20120090@thehoang:~/BigDataLabs$ sudo apt-get install ssh
[sudo] password for 20120090:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ssh is already the newest version (1:8.9p1-3ubuntu0.1).
The following packages were automatically installed and are no longer required:
  linux-headers-5.15.0-48 generic linux-headers-5.15.0-53 linux-headers-5.15.0-56 linux-headers-5.15.0-56-generic linux-image-5.15.0-48-generic
  linux-image-5.15.0-53-generic linux-image-5.15.0-56-generic linux-modules-5.15.0-48-generic linux-modules-5.15.0-53-generic linux-modules-5.15.0-56-generic
  linux-modules-extra-5.15.0-53-generic linux-modules-extra-5.15.0-56-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 21 not upgraded.
20120090@thehoang:~/BigDataLabs$ sudo apt-get install pdsh
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
pdsh is already the newest version (2.31-3build2).
The following packages were automatically installed and are no longer required:
  linux-headers-5.15.0-48 generic linux-headers-5.15.0-53 linux-headers-5.15.0-56 linux-headers-5.15.0-56-generic linux-image-5.15.0-48-generic
  linux-image-5.15.0-53-generic linux-image-5.15.0-56-generic linux-modules-5.15.0-48-generic linux-modules-5.15.0-53-generic linux-modules-5.15.0-56-generic
  linux-modules-extra-5.15.0-53-generic linux-modules-extra-5.15.0-56-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 21 not upgraded.
20120090@thehoang:~/BigDataLabs$
```

Figure 1.7: Step 8: Install ssh



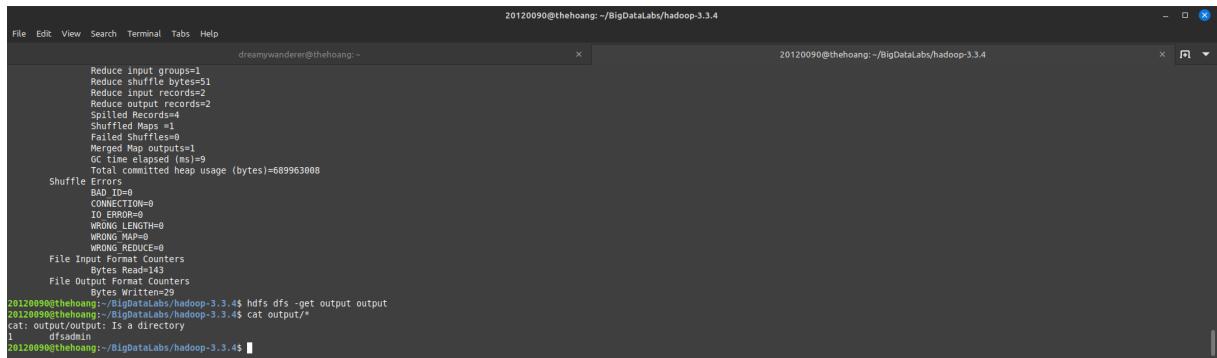
```
File Edit View Search Terminal Help
dreamwanderer@thehoang:~
```

```
20120090@thehoang:~/BigDataLabs$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
/home/tmp/20120090/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/tmp/20120090/.ssh/id_rsa
The key's randomart image is:
+---[RSA 3072]---+
|          .+ |
|         .o+++. |
|        .+=.. 000 |
+---[SHA256]---+
20120090@thehoang:~/BigDataLabs$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
20120090@thehoang:~/BigDataLabs$ ssh localhost
Last login: Fri Mar  3 17:25:12 2023 from 127.0.0.1
20120090@thehoang:~$
```

Figure 1.8: Step 9: Enable passwordless login by generating new SSH key with an empty passphrase

¹Apache Hadoop, “Hadoop: Setting up a Single Node Cluster,” Jul. 29, 2022. <https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html>.

²Apache Hadoop, “Hadoop: Setting up a Single Node Cluster,” Jul. 29, 2022. <https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html>.



The screenshot shows two terminal windows side-by-side. The left window title is "dreamywanderer@thehoang:~" and the right window title is "20120090@thehoang:~/BigDataLabs/hadoop-3.3.4". Both windows show the results of a Hadoop job execution. The left window displays various metrics like Reduce input groups=1, Reduce shuffle bytes=51, etc. The right window shows the command "hdfs dfs -get output output" being run, followed by the message "cat: output/output: Is a directory".

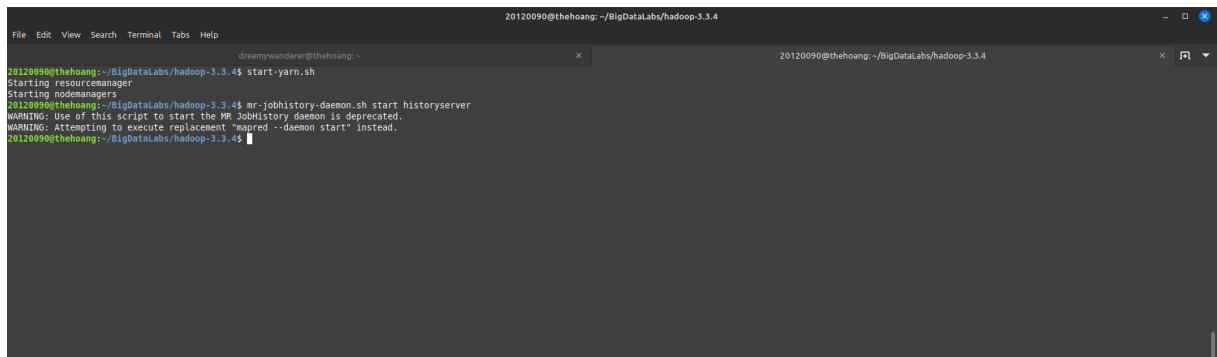
```

20120090@thehoang:~/BigDataLabs/hadoop-3.3.4
File Edit View Search Terminal Tabs Help
dreamywanderer@thehoang:~ x 20120090@thehoang:~/BigDataLabs/hadoop-3.3.4 x 🔍
Reduce input groups=1
Reduce shuffle bytes=51
Reduce input records=2
Reduce output records=2
Spilled Records=4
Shuffled Maps=1
Failed Shuffles=0
Merged Map outputs=1
GC time elapsed (ms)=9
Total committed heap usage (bytes)=689963008
Shuffle Errors=0
BAD ID=0
CONNECTION=0
ID ERROR=0
WRONG MAP=0
WRONG REDUCE=0
File Input Format Counters
Bytes Read=143
File Output Format Counters
Bytes Written=29
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ hdfs dfs -get output output
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ cat output
cat: output/output: Is a directory
1
dfadmin
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ 

```

Figure 1.12: Step 13: Copy the output files from the distributed filesystem to the local filesystem to examine them

1.1.3 Running in Pseudo-Distributed Mode with YARN



The screenshot shows a single terminal window titled "dreamywanderer@thehoang:~". It displays the command "start-yarn.sh" being run. The output shows the ResourceManager and NodeManager daemons starting, along with a warning about the deprecated "mr-jobhistory-daemon.sh start historyserver" command.

```

20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ start-yarn.sh
File Edit View Search Terminal Tabs Help
dreamywanderer@thehoang:~ x 20120090@thehoang:~/BigDataLabs/hadoop-3.3.4 x 🔍
Starting resourcemanager
Starting nodemanagers
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ mr-jobhistory-daemon.sh start historyserver
WARNING: Use of this script to start the MR JobHistory daemon is deprecated.
WARNING: Attempting to execute replacement "mapred --daemon start" instead.
20120090@thehoang:~/BigDataLabs/hadoop-3.3.4$ 

```

Figure 1.13: Step 14: Start the ResourceManager daemon and NodeManager daemon by running start-yarn.sh. Optionally, start the history server by running mr-jobhistory-daemon.sh start historyserver

1.1.4 Accessing the Web interfaces of running Hadoop process

These are the results when start all main components of a Single-node Hadoop Cluster

Overview 'localhost:9000' (✓active)

Started:	Fri Mar 03 23:54:27 +0700 2023
Version:	3.3.4, ra585a73c3e02ac62350c136643a5e7f6095a3dbb
Compiled:	Fri Jul 29 19:32:00 +0700 2022 by stevel from branch-3.3.4
Cluster ID:	CID-94bb0d6a-2942-4a14-8064-1cbfd948f79f
Block Pool ID:	BP-438193187-127.0.1.1-1677859565881

Summary

Security is off.
Safemode is off.
25 files and directories, 11 blocks (11 replicated blocks. 0 erasure coded block groups) = 36 total filesystem object(s).
Heap Memory used 154.57 MB of 246 MB Heap Memory. Max Heap Memory is 2.89 GB.
Non Heap Memory used 58.51 MB of 61.44 MB Committed Non Heap Memory. Max Non Heap Memory is <unbounded>.

Configured Capacity:	27.53 GB
Configured Remote Capacity:	0 B
DFS Used:	144 KB (0%)
Non DFS Used:	25.25 GB
DFS Remaining:	879.32 MB (3.12%)
Block Pool Used:	144 KB (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)

Figure 1.14: Web UI of NameNode through `http://localhost:9870/`

DataNode on thehoang:9866

Cluster ID:	CID-94bb0d6a-2942-4a14-8064-1cbfd948f79f
Started:	Fri Mar 03 23:54:30 +0700 2023
Version:	3.3.4, ra585a73c3e02ac62350c136643a5e7f6095a3dbb

Block Pools

Namenode Address	Block Pool ID	Actor State	Last Heartbeat	Last Block Report	Last Block Report Size (Max Size)
localhost:9000	BP-438193187-127.0.1.1-1677859565881	RUNNING	2s	4 minutes	100 B (128 MB)

Volume Information

Directory	StorageType	Capacity Used	Capacity Left	Capacity Reserved	Reserved Space for Replicas	Blocks
/tmp/hadoop-20120090/dfs/data	DISK	136.04 KB	887.46 MB	0 B	0 B	11

Hadoop, 2022.

Figure 1.15: Web UI of a DataNode

The screenshot shows the 'All Applications' page of the ResourceManager web UI. The left sidebar includes links for Cluster Metrics, About Nodes, Node Labels, Applications (with states: NEW, SAVING, SUBMITTED, ACCEPTED, RUNNING, FINISHED, FAILED, KILLED), Scheduler, and Tools. The main content area displays 'Cluster Metrics' with values: Apps Submitted: 0, Apps Pending: 0, Apps Running: 0, Apps Completed: 0, Containers Running: 0, Used Resources: <memory:0 B, vCores:0>, Total Resources: <memory:0 B, vCores:0>, and Reserved Resources: <memory:0 B, vCores:0>. Below this are sections for 'Cluster Nodes Metrics' (Active Nodes: 0, Decommissioning Nodes: 0, Decommissioned Nodes: 0, Lost Nodes: 1, Unhealthy Nodes: 0) and 'Scheduler Metrics' (Scheduler Type: Capacity Scheduler, Scheduling Resource Type: [memory-mb (unit=M), vcores], Minimum Allocation: <memory:1024, vCores:1>, Maximum Allocation: <memory:8192, vCores:4>). A table at the bottom shows 0 entries.

Figure 1.16: Web UI of a ResourceManager through `http://localhost:8088`

The screenshot shows the 'JobHistory' page of the MapReduce history server web UI. The left sidebar includes links for Application (About Jobs) and Tools. The main content area displays 'Retired Jobs' with a table header: Submit Time, Start Time, Finish Time, Job ID, Name, User, Queue, State, Maps Total, Maps Completed, Reduces Total, Reduces Completed, and Elapsed Time. Below the header is a search bar labeled 'Search:' and a message 'No data available in table'. A table at the bottom shows 0 entries.

Figure 1.17: Web UI of MapReduce history server `http://localhost:19888`

1.2 Member 2: 20120011 - Nguyen Hoang Huy

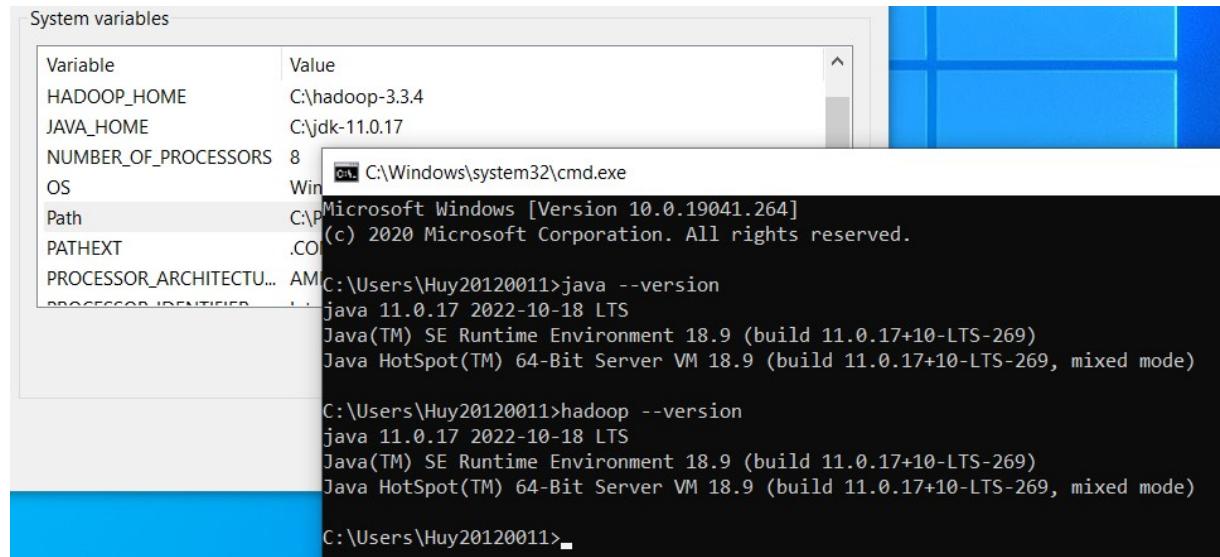


Figure 1.18: Install Java and Hadoop then setup the environment path

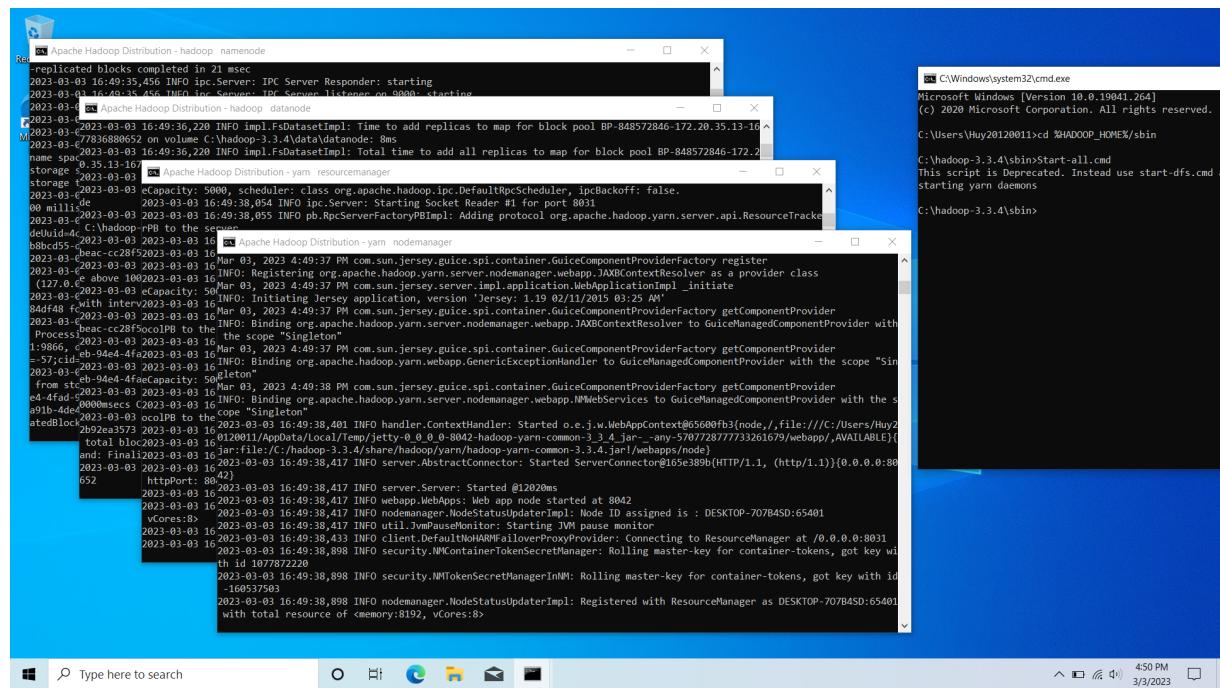


Figure 1.19: Run start-all.cmd to start namenode, datanode, resourcemanager, nodemanager program

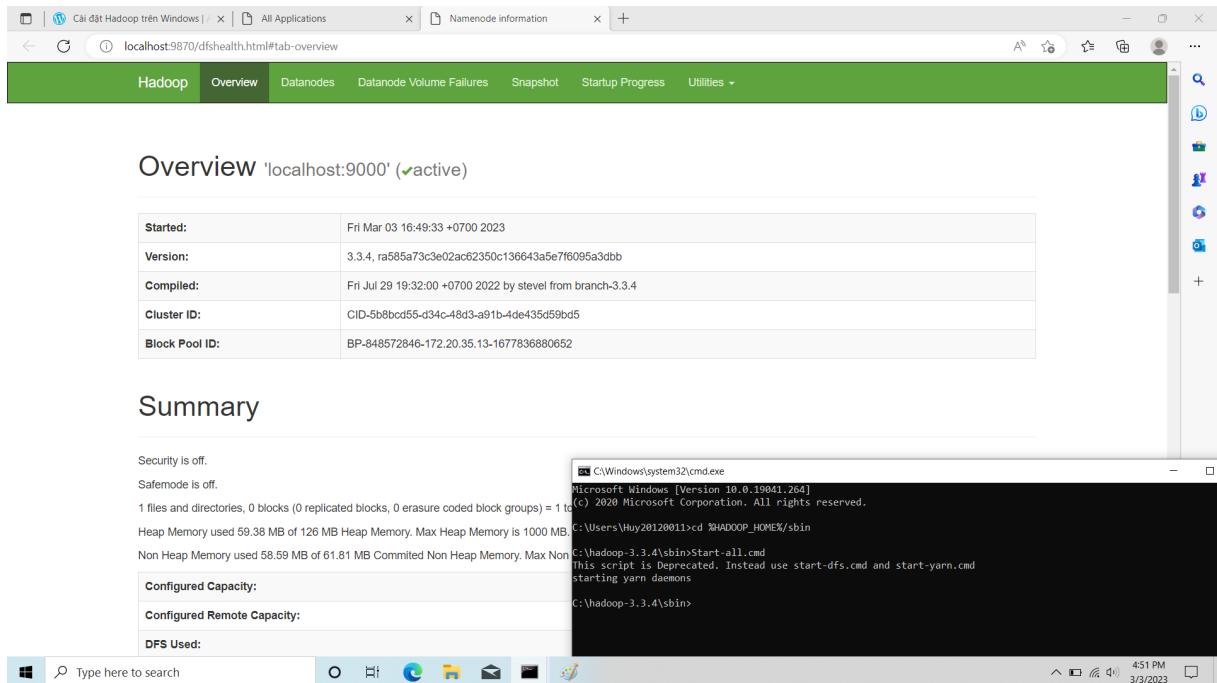


Figure 1.20: Access localhost:9000

1.3 Member 3: 20120030 - Nguyen Thien An

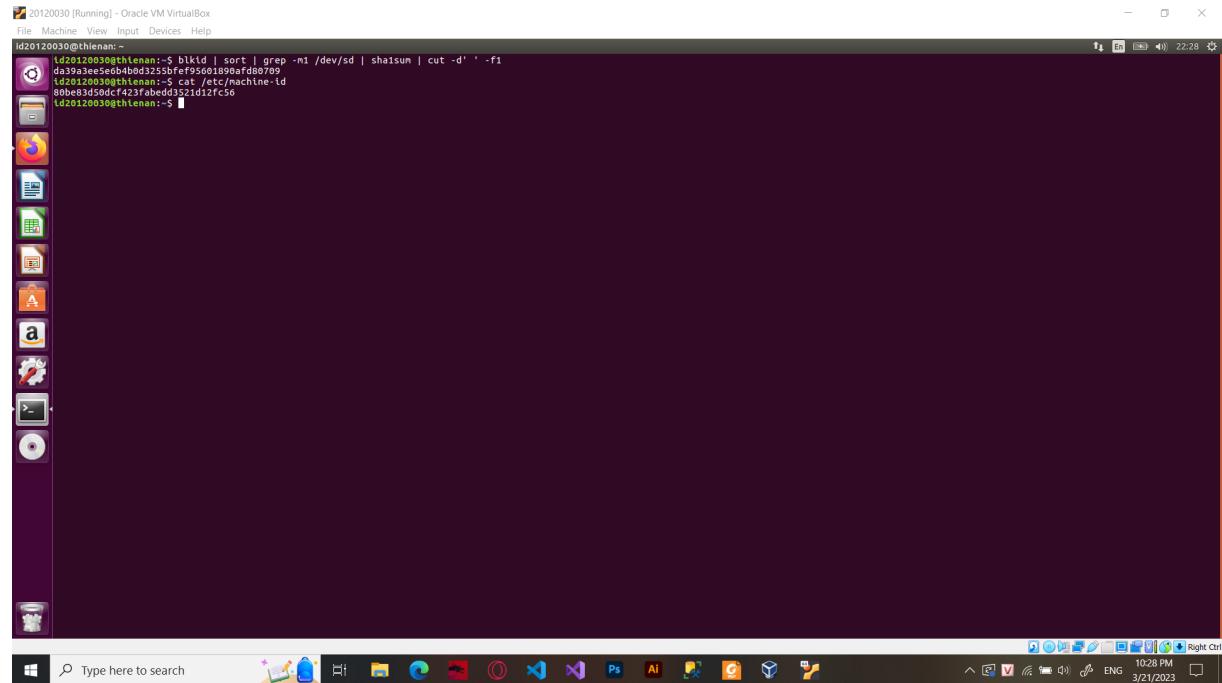


Figure 1.21: Id of the first block to proof this is a personal computer.

```
id20120030@thienan:~$ hadoop version
Hadoop 3.2.4
Source code repository Unknown -r 7e5d9983b388e372fe640f21f048f2f2ae6e9eba
Compiled by ubuntu on 2022-07-12T11:58Z
Compiled with protoc 2.5.0
From source with checksum ee031c16fe785bbb35252c749418712
This command was run using /home/id20120030/hadoop-3.2.4/share/hadoop/common/hadoop-common-3.2.4.jar
id20120030@thienan:~$ javac -version
javac 1.8.0_292
id20120030@thienan:~$ java -version
openjdk version "1.8.0_292"
OpenJDK Runtime Environment (build 1.8.0_292-8u292-b10-0ubuntu1~16.04.1-b10)
OpenJDK 64-Bit Server VM (build 25.292-b10, mixed mode)
id20120030@thienan:~$
```

Figure 1.22: Install Java and Hadoop then setup the environment path

```

id20120030@thienan:~ [id20120030@thienan:~]$ sudo apt-get install ssh
[sudo] password for id20120030:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  ssh
0 upgraded, 1 newly installed, 0 to remove and 285 not upgraded.
Need to get 169 kB of additional disk space will be used.
Selecting previously unselected package ssh.
(Reading database... 210418 files and directories currently installed.)
Preparing to unpack .../ssh_1%3a7_z02~4ubuntu2.10_all.deb ...
Setting Up ssh (1:7.2p2~4ubuntu2.10) ...
[id20120030@thienan:~]$ sudo apt-get install pdsh
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  genders libgenders0
Suggested packages:
  rdps
The following NEW packages will be installed:
  genders libgenders0 pdsh
0 upgraded, 2 newly installed, 0 to remove and 285 not upgraded.
Need to get 169 kB of additional disk space will be used.
After this operation, 501 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get: http://kh.archive.ubuntu.com/ubuntu xenial/universe amd64 libgenders0 amd64 1.21-1build1 [30.7 kB]
Get: http://kh.archive.ubuntu.com/ubuntu xenial/universe amd64 genders amd64 1.21-1build1 [30.6 kB]
Get:3 http://kh.archive.ubuntu.com/ubuntu xenial/universe amd64 pdsh amd64 2.31-3build1 [108 kB]
Fetched 169 kB in 0s (122 kB/s)
Preconfiguring packages...
Unpacking selected package libgenders0:amd64...
(Reading database... 210423 files and directories currently installed.)
Preparing to unpack .../libgenders0_1.21-1build1_amd64.deb ...
Unpacking libgenders0:amd64 (1.21-1build1) ...
Selecting previously unselected package genders.
Preparing to unpack .../genders_1.21-1build1_amd64.deb ...
Unpacking genders (1.21-1build1) ...
Selecting previously unselected package pdsh.
Preparing to unpack .../pdsh_2.31-3build1_amd64.deb ...
Unpacking pdsh (2.31-3build1) ...
Processing triggers for libc-bin (2.23-0ubuntu11.3) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up libgenders0:amd64 (1.21-1build1) ...
Setting up genders (1.21-1build1) ...
Setting up pdsh (2.31-3build1) ...
Processing triggers for libc-bin (2.23-0ubuntu11.3) ...

```

Figure 1.23: Install ssh and pdsh

```

id20120030@thienan:~ [id20120030@thienan:~]$ echo $JAVA_HOME
/usr/lib/jvm/java-8-openjdk-amd64
[id20120030@thienan:~]$ export HADOOP_HOME=-
[~/home/id20120030
[id20120030@thienan:~]$ echo $HADOOP_HOME
-/hadoop-3.2.4
[id20120030@thienan:~]$ export HADOOP_HOME=-/hadoop-3.2.4
[id20120030@thienan:~]$ echo $HADOOP_HOME
-/hadoop-3.2.4
[id20120030@thienan:~]$ sudo nano ~/.bashrc
[sudo] password for id20120030:
[id20120030@thienan:~]$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/bin:/sbin:/usr/games:/usr/local/games:/snap/bin:/home/id20120030/hadoop-3.2.4/sbin:/home/id20120030/hadoop-3.2.4/bin
[id20120030@thienan:~]

```

Figure 1.24: JAVA_HOME and HADOOP_HOME

```

id20120030@thienan:~/hadoop-3.2.4
[td20120030@thienan:~] cd /home/ld20120030/hadoop-3.2.4
[td20120030@thienan:~/hadoop-3.2.4$ hadoop version
Hadoop 3.2.4
Source code repository Unknown -r 7e5d9993b388e372fe40f21f048f2f2aede9e9ba
Compiled with compiler 2.5.0
From source with checksum eee31c16fe785bb35252c749418712
This command was run using /home/ld20120030/hadoop-3.2.4/share/hadoop/common/hadoop-common-3.2.4.jar
Usage: hadoop [OPTIONS] SUBCOMMAND [SUBCOMMAND OPTIONS]
      or   hadoop [OPTIONS] CLASSNAME [CLASSNAME OPTIONS]
      where CLASSNAME is a user-provided Java class
      OPTIONS is none or any of:
buildpaths          attempt to add class files from build tree
--config dir        Hadoop config directory
--debug             turn on internal debug mode
--help              usage information
hostnames list[,of,host,names] hosts to use in slave mode
hosts filename     list of hosts to use in slave mode
loglevel level    set the log4j level for this command
workers            turn on worker mode
SUBCOMMAND is one of:
  Admin Commands:
    daemonlog    get/set the log level for each daemon
  Client Commands:
    archive       create a Hadoop archive
    checksum      check native Hadoop and compression libraries availability
    claspath      prints the class path needed to get the Hadoop jar and the
                  required libraries
    conftest      validate configuration XML files
    credential    interact with credential providers
    distch       distribute metastore changer
    distcp      copy file or directories recursively
    dtutil       operations related to delegation tokens
    envvars      display current environment variables
    gridmix     submit a mix of synthetic job, modeling a profiled from
                production load
    jar <jar>      run a jar file. NOTE: please use "yarn jar" to launch YARN
    jnopath      prints the Java.library.path
    kdiag        Diagnose Kerberos Problems
    kerbnane    show auth_to_local principal conversion
    key         manage keys for the keyprovider
    rumenFolder  scale a rumen input trace
    rumenTrace   convert logs into a rumen trace

```

Figure 1.25: Show hadoop version and bin/hadoop

```

id20120030@thienan:-
[td20120030@thienan:~] cd /home/ld20120030/hadoop-3.2.4/etc/hadoop
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sudo nano core-site.xml
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sudo nano core-site.xml
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sudo nano hdfs-site.xml
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sudo nano mapred-site.xml
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sudo nano yarn-site.xml
[td20120030@thienan:~/hadoop-3.2.4/etc/hadoop$ sign_and_send_pubkey: signing failed: agent refused operation
id20120030@localhost's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-210-generic x86_64)
 * Documentation: https://help.ubuntu.com/
293 packages can be updated.
2 updates are security updates.

Last login: Tue Mar 21 04:38:43 2023 from 127.0.0.1
[td20120030@thienan:~] 

```

Figure 1.26: Make changes on core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml

```

id20120030@thienan:~ 
[ld20120030@thienan:~] cd /home/ld20120030/hadoop-3.2.4/etc/hadoop
[ld20120030@thienan:~] hadoop-3.2.4/etc/hadoop$ sudo nano core-site.xml
[sudo] password for id20120030:
[ld20120030@thienan:~] hadoop-3.2.4/etc/hadoop$ sudo nano hdfs-site.xml
[ld20120030@thienan:~] hadoop-3.2.4/etc/hadoop$ sudo nano mapred-site.xml
[ld20120030@thienan:~] hadoop-3.2.4/etc/hadoop$ sudo nano yarn-site.xml
[ld20120030@thienan:~] hadoop-3.2.4/etc/hadoop$ ssh localhost
ssh: connect to host localhost port 22: Connection refused
ssh: connect to host localhost port 22: Connection refused
ssh: connect to host localhost port 22: Connection refused
ssh: connect to host localhost port 22: Connection refused
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-210-generic x86_64)

 * Documentation: https://help.ubuntu.com/
 
293 packages can be updated.
2 updates are security updates.

Last login: Tue Mar 21 04:38:43 2023 from 127.0.0.1
[ld20120030@thienan:~] ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
/home/ld20120030/.ssh/id_rsa already exists.
Your identification has been saved in /home/ld20120030/.ssh/id_rsa.
Your public key has been saved in /home/ld20120030/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:Le8o5AGTB0n5v/NlSTMxmsrNDAhqCie0dC1NVnoY id20120030@thienan
The key's randomart image is:
+---[RSA 2048]---+
|Bo.o...|
|Oo. + |
|OoE = +|
|Oo. + B o|
|+.o . +S o|
|.... + |
|... . |
|.. o. |
+---[SHA256]---+
[ld20120030@thienan:~] cat ~/.ssh/id_rsa.pub > ~/.ssh/authorized_keys
[ld20120030@thienan:~] chmod 6000 ~/.ssh/authorized_keys
[ld20120030@thienan:~] hadoop-3.2.4/bin/hdfs namenode -format
namenode is running as process 4881. Stop it first and ensure /tmp/hadoop-id20120030-namenode.pid file is empty before retry.
[ld20120030@thienan:~] export PDSH_RCMD_TYPE=ssh
[ld20120030@thienan:~]

```

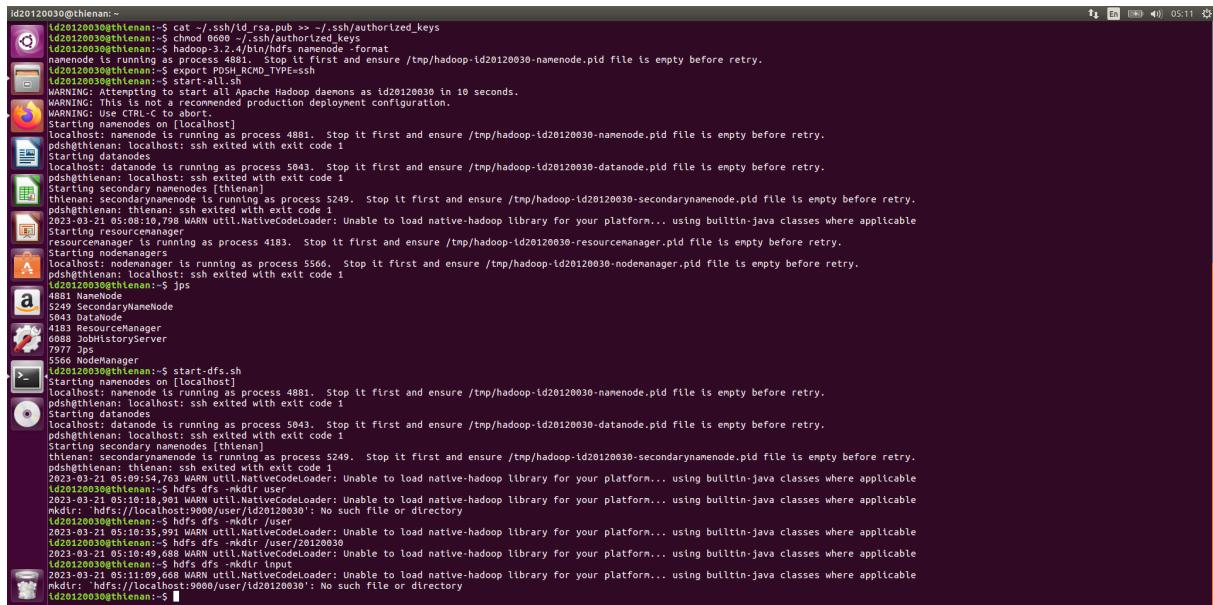
Figure 1.27: ssh localhost

```

id20120030@thienan:~ 
[ld20120030@thienan:~] ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
/home/ld20120030/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /home/ld20120030/.ssh/id_rsa.
Your public key has been saved in /home/ld20120030/.ssh/id_rsa.pub.
The key's randomart image is:
+---[RSA 2048]---+
|Bo.o...|
|Oo. + |
|OoE = +|
|Oo. + B o|
|+.o . +S o|
|.... + |
|... . |
|.. o. |
+---[SHA256]---+
[ld20120030@thienan:~] cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
[ld20120030@thienan:~] chmod 6000 ~/.ssh/authorized_keys
[ld20120030@thienan:~] ./bin/hdfs namenode -format
2023-03-20 05:08:10,798 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-Java classes where applicable
[ld20120030@thienan:~] start-all.sh
Starting namenodes on [localhost]
localhost: namenode is running as process 4881. Stop it first and ensure /tmp/hadoop-id20120030-namenode.pid file is empty before retry.
[ld20120030@thienan:~] start-datanodes
localhost: datanode is running as process 5043. Stop it first and ensure /tmp/hadoop-id20120030-datanode.pid file is empty before retry.
[ld20120030@thienan:~] start-secondarynamenode
thienan: secondarynamenode is running as process 5249. Stop it first and ensure /tmp/hadoop-id20120030-secondarynamenode.pid file is empty before retry.
[ld20120030@thienan:~] start-resourcemanager
thienan: ssh exited with exit code 1
2023-03-20 05:08:10,798 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-Java classes where applicable
[ld20120030@thienan:~] start-node-managers
localhost: nodemanager is running as process 5566. Stop it first and ensure /tmp/hadoop-id20120030-nodemanager.pid file is empty before retry.
[ld20120030@thienan:~] jps
4881 NameNode
5249 SecondaryNameNode
5043 DataNode
5566 NodeManager
6088 JobHistoryServer
7977 Jps
5566 NodeManager
[ld20120030@thienan:~]

```

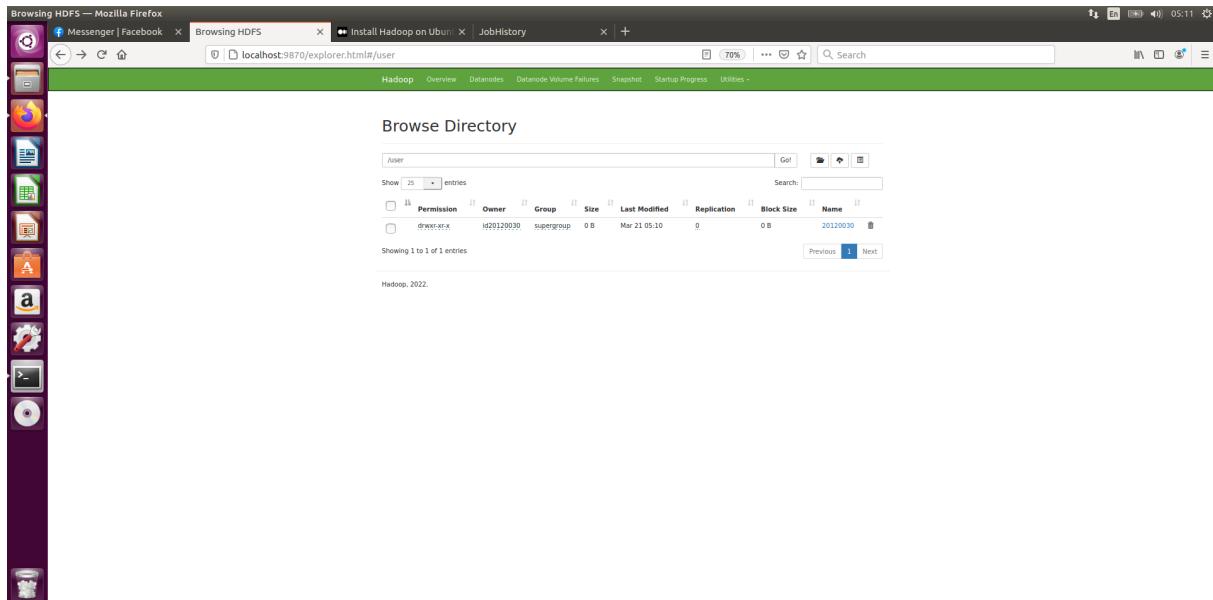
Figure 1.28: Start all daemons hadoop



```

id20120030@thienan:~ 
[120120030@thienan:~]$ cat ~/.ssh/id_rsa.pub > ~/.ssh/authorized_keys
[120120030@thienan:~]$ chmod 600 ~/.ssh/authorized_keys
[120120030@thienan:~]$ hadoop fs -rmr /user
[120120030@thienan:~]$ export HDFS_RCMD_TYPE=ssh
[120120030@thienan:~]$ hadoop namenode -format
WARNING: Attempting to start all Apache Hadoop daemons as id20120030 in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
[120120030@thienan:~]$ startnodename on [localhost]
localhost: namenode is running as process 4881. Stop it first and ensure /tmp/hadoop-id20120030-namenode.pid file is empty before retry.
pdsh@thienan: localhost: ssh exited with exit code 1
Starting datanodes
localhost: datanode is running as process 5043. Stop it first and ensure /tmp/hadoop-id20120030-datanode.pid file is empty before retry.
pdsh@thienan: localhost: ssh exited with exit code 1
Starting secondary namenodes [thienan]
thienan: secondarynamenode is running as process 5249. Stop it first and ensure /tmp/hadoop-id20120030-secondarynamenode.pid file is empty before retry.
pdsh@thienan: localhost: ssh exited with exit code 1
2023-03-21 05:08:10,798 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
resourcemanager is running as process 4183. Stop it first and ensure /tmp/hadoop-id20120030-resourcemanager.pid file is empty before retry.
pdsh@thienan: localhost: nodemanager is running as process 5566. Stop it first and ensure /tmp/hadoop-id20120030-nodemanager.pid file is empty before retry.
pdsh@thienan: localhost: ssh exited with exit code 1
[120120030@thienan:~]$ jps
4881 NameNode
5249 SecondaryNameNode
5043 DataNode
4183 ResourceManager
6088 JobHistoryServer
5566 NodeManager
[120120030@thienan:~]$ start-dfs.sh
Starting namenodes on [localhost]
pdsh@thienan: localhost: ssh exited with exit code 1
Starting datanodes
localhost: datanode is running as process 5043. Stop it first and ensure /tmp/hadoop-id20120030-datanode.pid file is empty before retry.
pdsh@thienan: localhost: ssh exited with exit code 1
Starting secondary namenodes [thienan]
thienan: secondarynamenode is running as process 5249. Stop it first and ensure /tmp/hadoop-id20120030-secondarynamenode.pid file is empty before retry.
pdsh@thienan: thienan: ssh exited with exit code 1
2023-03-21 05:10:18,901 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
mkdtr: 'hdfs://localhost:9000/user/id20120030': No such file or directory
[120120030@thienan:~]$ hdfs dfs -mkdir /user
[120120030@thienan:~]$ hdfs dfs -mkdir /user
2023-03-21 05:10:49,688 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
[120120030@thienan:~]$ hdfs dfs -mkdir /input
2023-03-21 05:10:49,688 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
mkdtr: 'hdfs://localhost:9000/user/id20120030': No such file or directory
[120120030@thienan:~]$ 

```

Figure 1.29: Create a folder named 20120030 in /user**Figure 1.30:** Create folder 20120030 successfully!

```
td20120030@thienan:~/hadoop-3.2.4$ start-yarn.sh
Starting resourcemanager
resourcemanager is running as process 18086. Stop it first and ensure /tmp/hadoop-id20120030-resourcemanager.pid file is empty before retry.
Starting nodemanagers
localhost: nodemanager is running as process 18224. Stop it first and ensure /tmp/hadoop-id20120030-nodemanager.pid file is empty before retry.
td20120030@thienan:~/hadoop-3.2.4$ mr-jobhistory-daemon.sh start historyserver
WARNING: Use of this script to start the MR JobHistory daemon is deprecated.
WARNING: Attempting to execute replacement "mapred --daemon start" instead.
td20120030@thienan:~/hadoop-3.2.4$
```

Figure 1.31: Start yarn and historyserver

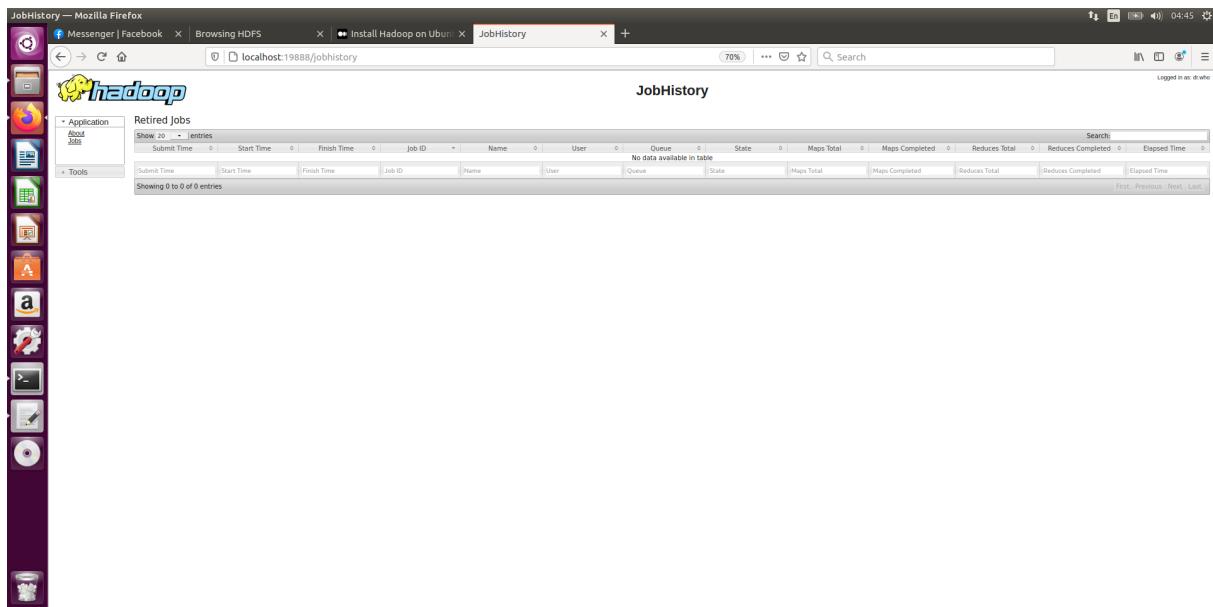
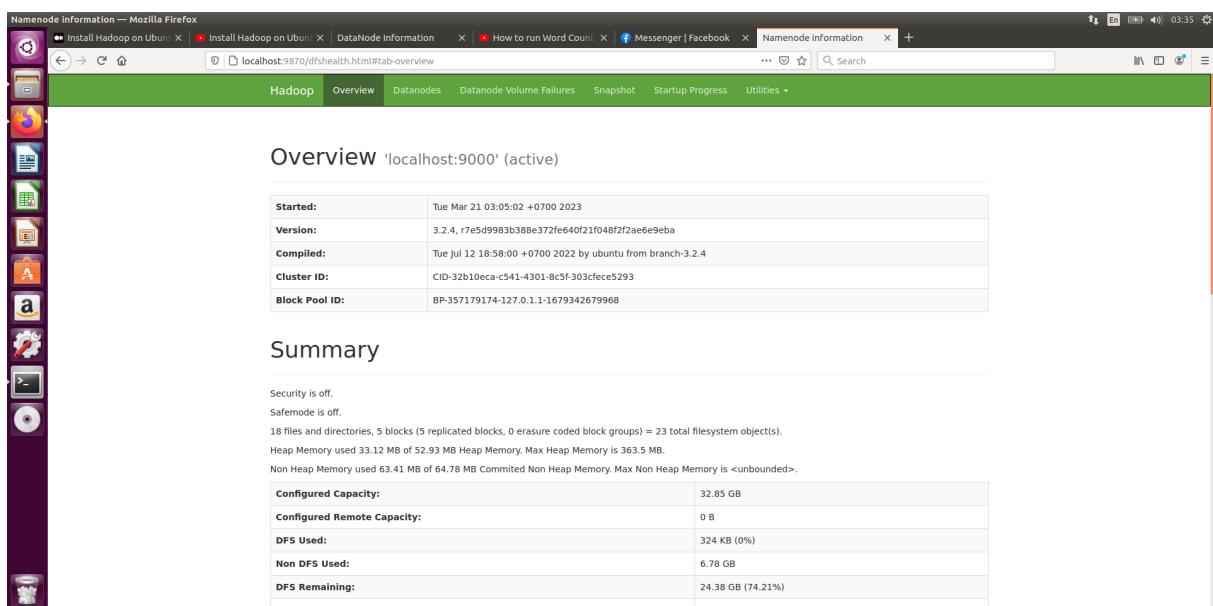
The screenshot shows a Mozilla Firefox browser window with the title "DataNode Information — Mozilla Firefox". The address bar shows the URL "thienan:9864/datanode.html". The main content area is titled "DataNode on thienan:9866". It contains three sections: "Cluster ID:", "Started:", and "Version:". Below these, there is a table titled "Block Pools" with one row. At the bottom, there is a table titled "Volume Information" with one row.

Cluster ID:	CID-32b10eca-c541-4301-8c5f-303cfce5293
Started:	Tue Mar 21 03:05:07 +0700 2023
Version:	3.2.4, r7e5d9983b388e372fe640f21f048f2fae6e9eba

Namenode Address	Block Pool ID	Actor State	Last Heartbeat	Last Block Report	Last Block Report Size (Max Size)
localhost:9000	BP-357179174-127.0.1.1-1679342679968	RUNNING	0s	28 minutes	0 B (64 MB)

Directory	StorageType	Capacity Used	Capacity Left	Capacity Reserved	Reserved Space for Replicas	Blocks
/tmp/hadoop-id20120030/dfs/data	DISK	324 KB	24.38 GB	0 B	0 B	5

Figure 1.32: DataNode

**Figure 1.33:** MapReduce Server**Figure 1.34:** NameNode

The screenshot shows a Firefox browser window with multiple tabs open. The active tab is titled "All Applications" and displays a table of YarnNode metrics. The table has columns for ID, User, Name, Application Type, Queue, Application Priority, StartTime, LaunchTime, FinishTime, State, FinalStatus, Running Containers, Allocated CPU, Allocated Memory MB, Allocated GPUs, Reserved Memory MB, Reserved GPUs, % of Queue, % of Cluster, Progress, Tracking UI, and Blacklisted Nodes. There is one entry in the table:

ID	User	Name	Application Type	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU	Allocated Memory MB	Allocated GPUs	Reserved Memory MB	Reserved GPUs	% of Queue	% of Cluster	Progress	Tracking UI	Blacklisted Nodes
application_1673342716732_0001	id20120030	word count	MAPREDUCE	default	0	Tue Mar 21 03:14:17 2023	+0700 2023	Tue Mar 21 03:14:19 2023	FINISHED	SUCCEEDED	N/A	N/A	N/A	N/A	N/A	0.0	0.0	0.0	0.0	0	

Showing 1 to 1 of 1 entries

Figure 1.35: YarnNode

```
id20120030@thienan:~  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
Starting datanodes  
localhost: datanode is running as process 5043. Stop it first and ensure /tmp/hadoop-id20120030-datanode.pid file is empty before retry.  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
Starting secondary namenodes [thienan]  
thienan: secondarynamenode is running as process 5249. Stop it first and ensure /tmp/hadoop-id20120030-secondarynamenode.pid file is empty before retry.  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
Starting resourcemanager  
resourcemanager is running as process 4183. Stop it first and ensure /tmp/hadoop-id20120030-resourcemanager.pid file is empty before retry.  
Starting nodemanagers  
localhost: nodemanager is running as process 5566. Stop it first and ensure /tmp/hadoop-id20120030-nodemanager.pid file is empty before retry.  
1d20120030@thienan:~$ jps  
4881 NameNode  
5249 SecondaryNameNode  
4043 DataNode  
4183 ResourceManager  
6088 JobHistoryServer  
7977 Jps  
1d20120030@thienan:~$ start-dfs.sh  
Starting namenodes on [localhost]  
localhost: namenode is running as process 4881. Stop it first and ensure /tmp/hadoop-id20120030-namenode.pid file is empty before retry.  
localhost: ssh localhost; ssh exited with exit code 1  
Starting datanodes  
localhost: datanode is running as process 5043. Stop it first and ensure /tmp/hadoop-id20120030-datanode.pid file is empty before retry.  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
pdsh@thienan: ssh localhost; ssh exited with exit code 1  
thienan: secondarynamenode is running as process 5249. Stop it first and ensure /tmp/hadoop-id20120030-secondarynamenode.pid file is empty before retry.  
2023-03-21 05:09:54,763 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
2023-03-21 05:09:54,763 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
mkdir: '/localhost:9000/user/d20120030': No such file or directory  
1d20120030@thienan:~$ hdfs dfs -mkdir /user  
2023-03-21 05:09:54,763 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
1d20120030@thienan:~$ hdfs dfs -mkdir /user/d20120030  
2023-03-21 05:10:49,688 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
1d20120030@thienan:~$ hdfs dfs -mkdir /user/d20120030  
2023-03-21 05:10:49,688 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
1d20120030@thienan:~$ stop-yarn.sh  
Stopping nodemanagers  
Stopping resourcemanager  
localhost: stop-jobhistory-daemon.sh stop historyserver  
WARNING: Use of this script to stop the MR JobHistory daemon is deprecated.  
WARNING: Attempting to execute replacement "mapred --daemon stop" instead.  
1d20120030@thienan:~$ stop-dfs.sh  
localhost: stop-dfs.sh on [localhost]  
Stopping datanodes  
Stopping secondary namenodes [thienan]  
2023-03-21 05:14:16,320 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable  
1d20120030@thienan:~$
```

Figure 1.36: Stop all daemons hadoop

1.4 Member 4: 20120165 - Hong Nhat Phuong

```

Allin [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Thg 3 5 12:30
nhatphuong@Allin:~$ hadoop version
Hadoop 3.3.4
Source code repository https://github.com/apache/hadoop.git -r a585a73c3e02ac62350c136643a5e7f6095a3dbb
Compiled by stevel on 2022-07-29T12:32Z
Compiled with protoc 3.7.1
From source with checksum fb9dd8918a7b8a5b430d61af858f6ec
This command was run using /home/nhatphuong/hadoop-3.3.4/share/hadoop/common/hadoop-common-3.3.4.jar
nhatphuong@Allin:~$ javac -version
javac 11.0.18
nhatphuong@Allin:~$ 
```

Figure 1.37: Install Java and Hadoop then setup the environment path

```

Allin [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Thg 3 4 15:34
nhatphuong@Allin:~$ cd /etc/hadoop/conf
nhatphuong@Allin:~/etc/hadoop/conf$ ls
hadoop-env.cmd      httpfs-log4j.properties      mapred-env.sh          yarn-env.cmd
hadoop-env.sh        httpfs-site.xml           mapred-queues.xml.template  yarn-env.sh
hadoop-metrics2.properties  kms-acls.xml       mapred-site.xml        yarnservice-log4j.properties
hadoop-policy.xml   kms-env.sh                 shellprofile.d         yarn-site.xml
nhatphuong@Allin:~/etc/hadoop/conf$ sudo nano hadoop-env.sh
[sudo] password for nhatphuong:
nhatphuong@Allin:~/etc/hadoop/conf$ sudo nano core-site.xml
nhatphuong@Allin:~/etc/hadoop/conf$ sudo nano hdfs-site.xml
nhatphuong@Allin:~/etc/hadoop/conf$ sudo nano mapred-site.xml
nhatphuong@Allin:~/etc/hadoop/conf$ sudo nano yarn-site.xml
nhatphuong@Allin:~/etc/hadoop/conf$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:nocoxBZcLQ7xIwL6th9pV+bYIK0+MPJvh7lZfiZbqJc.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
nhatphuong@localhost's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-35-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Introducing Expanded Security Maintenance for Applications.
 Receive updates to over 25,000 software packages with your
 Ubuntu Pro subscription. Free for personal use.

 https://ubuntu.com/pro

nhatphuong@Allin:~/etc/hadoop/conf$ 
```

Figure 1.38: Make changes on core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml

The screenshot shows a terminal window titled "Allin [Running] - Oracle VM VirtualBox". The terminal session is as follows:

```
File Machine View Input Devices Help
Activities Terminal
nhatphuong@Allin:~$ 2023-03-04 21:35:26,852 INFO namenode.FSImageFormatProtobuf: Saving image file /tmp/hadoop-nhatphuong/dfs/name/current/fsimage.ckpt_0000000000000000 using no compression
2023-03-04 21:35:27,166 INFO namenode.FSImageFormatProtobuf: Image file /tmp/hadoop-nhatphuong/dfs/name/current/fsimage.ckpt_0000000000000000 of size 405 bytes saved in 0 seconds .
2023-03-04 21:35:27,249 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
2023-03-04 21:35:27,342 INFO namenode.FSNamesystem: Stopping services started for active state
2023-03-04 21:35:27,342 INFO namenode.FSNamesystem: Stopping services started for standby state
2023-03-04 21:35:27,378 INFO namenode.FSImage: FSImageSaver clean checkpoint: txid=0 when meet shutdown.
2023-03-04 21:35:27,379 INFO namenode.NameNode: SHUTDOWN_MSG:
*****STARTUP_MSG*****
SHUTDOWN MSG: Shutting down NameNode at Allin/127.0.1.1
*****STOPPED_MSG*****
20120165: start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as nhatphuong in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [Allin]
2023-03-04 21:36:46,128 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
20120165: jps
3764 Jps
3300 ResourceManager
3414 NodeManager
2745 NameNode
2861 DataNode
3069 SecondaryNameNode
20120165:
```

Figure 1.39: Format namenode and start all daemons hadoop

2 Introduction to MapReduce

1. How do the input keys-values, the intermediate keys-values, and the output keys-values relate?

The input keys-values are the data sets that are processed by the Map function. The input data is divided into key-value pairs that are distributed across the nodes in the Hadoop cluster. The Map function processes each key-value pair and produces an intermediate key-value pair.

The intermediate keys-values are produced by the Map function and are used as input to the Reduce function. The intermediate key-value pairs are sorted by key and partitioned based on the key. Each partition is processed by a separate instance of the Reduce function.

The output keys-values are the final result of the MapReduce process. The Reduce function produces the final key-value pairs which are written to the Hadoop Distributed File System (HDFS) or to an external storage system.

2. How does MapReduce deal with node failures?

When the compute node at which the master is executing fails, a new copy can be started from the last checkpointed state. Only this one node can bring the entire process down; other node failures will be managed by the master.

The master sends heartbeat to every worker periodically. If there is no response from the worker within a certain period of time, the master marks the worker as failed. Map tasks that were assigned by that worker will be reset to their original idle state and eligible for scheduling by other workers, even if they have completed. Completed map tasks are re-executed because their output is stored at that compute node, and is now unavailable to access.

Similarly, reduce task in progress on the failed worker is also reset to idle and will have to be redone. Completed reduce tasks do not need to be re-executed since their output is stored in a global file system. When a map task is executed first by a worker and then later executed by another worker (because the first one failed), all workers executing reduce tasks are notified of the re-execution.

3. What is the meaning and implication of locality? What does it use?

Locality means that taking advantage of the fact that the input data is stored on the local disks of the machine that make up the clusters. As a result, the network bandwidth is conserved, and

when running large MapReduce operations on a large cluster, most input is read locally, consume no network bandwidth, speed processes up significantly.

Each file is divided into same size blocks (default 64 MB), then each block is created multiple copies (typically 3 copies). These blocks are stored on different machine, but highly optimised location (maybe 2 copies in the same rack, and 1 copy in the other rack in case of failing). MapReduce master take the location of the input files and above blocks into account and attempts to schedule a map task on a machine that contains a replica of the corresponding input data. If that is impossible, it tries to schedule map tasks near a replica of that task's input data (maybe on the same rack). Locality is also the result of Rack awareness of Hadoop system.

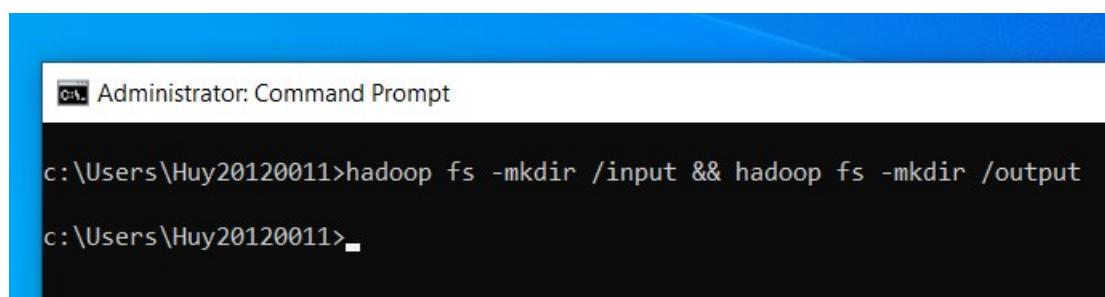
4. Which problem is addressed by introducing a combiner function to the MapReduce model?

The combiner function introduced to the MapReduce model addresses the problem of **network congestion** and **data transfer**. In detail, When data is processed by the Map function, it generates a large number of key-value pairs, which are then sorted and shuffled before being sent to the Reduce function. This sorting and shuffling process can generate a significant amount of network traffic, especially if there are many duplicate keys. The combiner function is introduced after the Map phase and before the Shuffle and Sort phase. Its purpose is to aggregate the output of the Map function for each key, reducing the amount of data that needs to be transferred across the network. This can greatly reduce network congestion and improve the overall performance and scalability of the MapReduce job. Hence, the combiner function reduces network traffic, which is the main problem addressed by its introduction to the MapReduce model.

3 Running a warm-up problem: Word Count

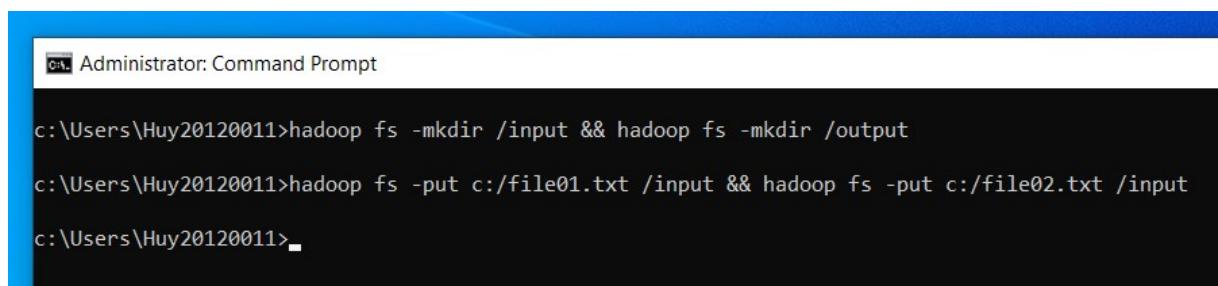
Each member has completed running the WordCount (v.1) program - a simple MapReduce program - on a Single node cluster on their local machine, as described in supplied tutorial. Also, this process is captured in screenshots.

3.1 Member 1: 20120011 - Nguyen Hoang Huy



```
c:\Users\Huy20120011>hadoop fs -mkdir /input && hadoop fs -mkdir /output  
c:\Users\Huy20120011>
```

Figure 3.1: Create input and output directory in hadoop fs



```
c:\Users\Huy20120011>hadoop fs -mkdir /input && hadoop fs -mkdir /output  
c:\Users\Huy20120011>hadoop fs -put c:/file01.txt /input && hadoop fs -put c:/file02.txt /input  
c:\Users\Huy20120011>
```

Figure 3.2: Put file01.txt and file02.txt into input directory

```
c:\Users\Huy20120011>hadoop jar C:\hadoop-3.3.4\share\hadoop\mapreduce\hadoop-mapreduce-examples-3.3.4.jar wordcount /input /output/result.txt
2023-03-10 22:03:33,870 INFO client.DefaultNBHARMFalloverProxyProvider: Connecting to ResourceManager at /0.0.0.0:8032
2023-03-10 22:03:34,200 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Huy/.staging/job_16784602642
12_0001
2023-03-10 22:03:34,200 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/Huy/.staging/job_16784602642
12_0001
```

Figure 3.3: Run WordCount program

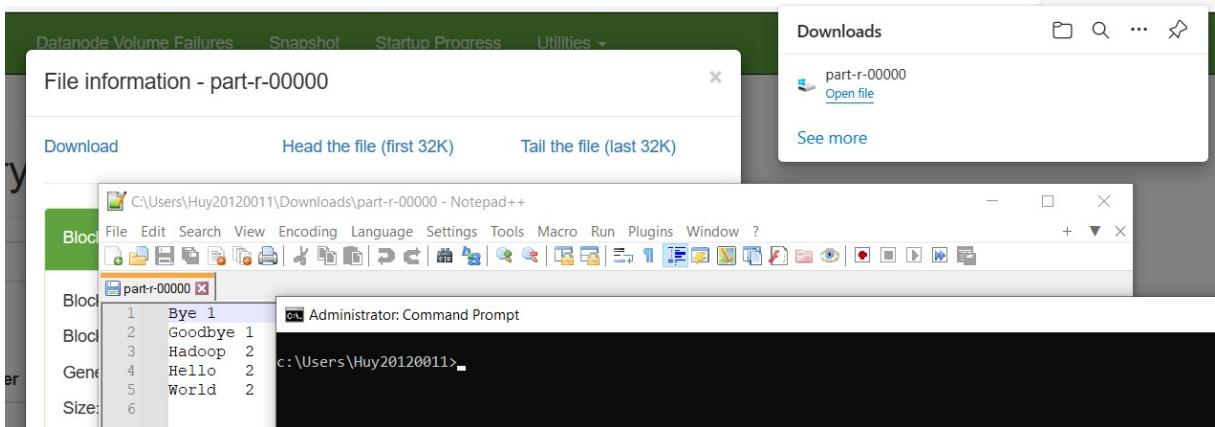


Figure 3.4: Let's show the result

3.2 Member 2: 20120030 - Nguyen Thien An

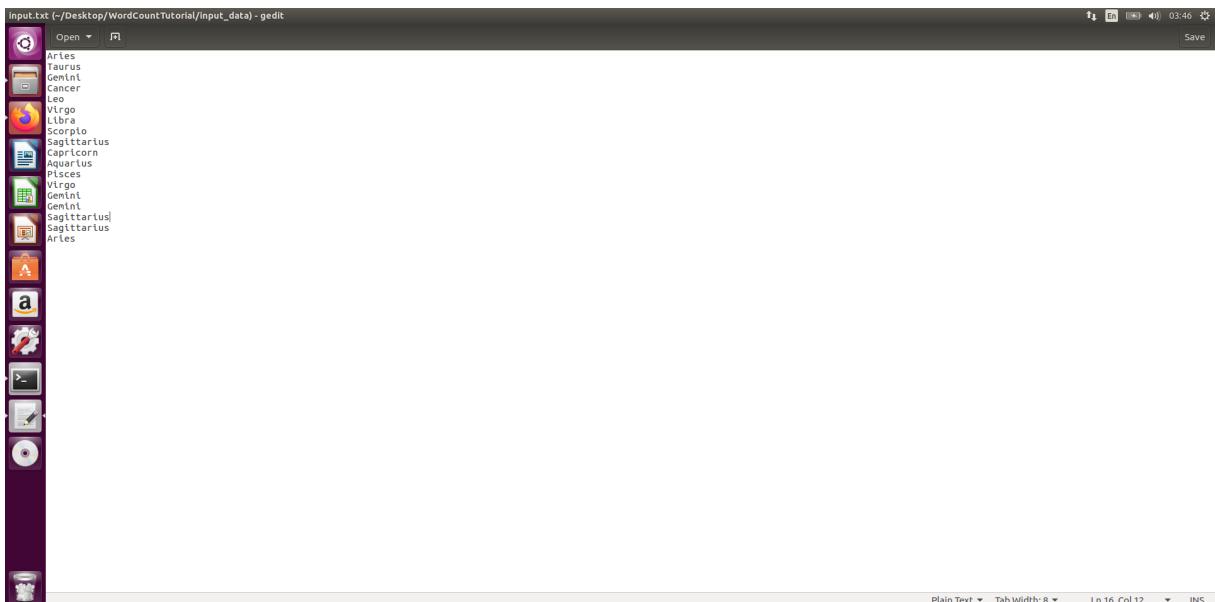
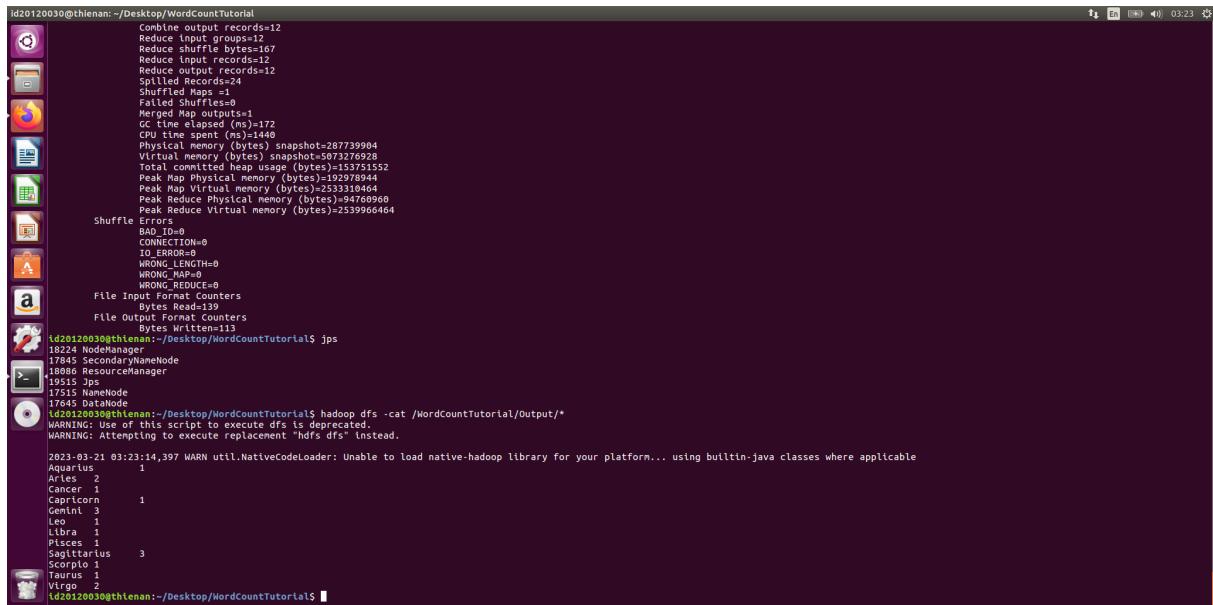


Figure 3.5: File "input.txt"

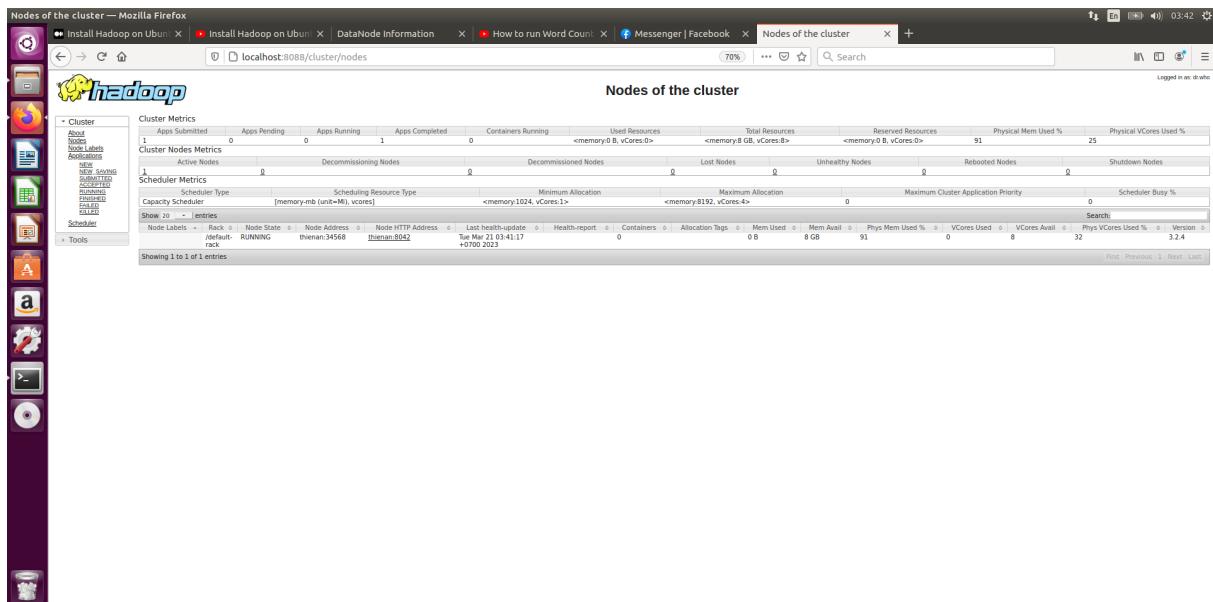


```

id20120030@thienan:~/Desktop/WordCountTutorial$ cat /WordCountTutorial/Output/*
Combine output records=12
Reduce Input groups=12
Reduce shuffle bytes=167
Reduce Input records=12
Reduce Output records=12
Copied Records=24
Shuffled Maps =1
Failed Shuffles=0
Map Input blocks=1
Map time elapsed (ms)=172
CPU time spent (ms)=1440
Physical memory (bytes) snapshot=287739904
Virtual memory (bytes) snapshot=1507372928
Total Physical memory (bytes)=253996464
Peak Map Physical memory (bytes)=192978944
Peak Map Virtual memory (bytes)=2533310464
Peak Reduce Physical memory (bytes)=94768960
Peak Reduce Virtual memory (bytes)=2539964644
Shuffle Errors
  BAD_ID=0
  CONNECT=0
  IO_ERROR=0
  WRONG_LENGTH=0
  WRONG_MAP=0
  WRONG_REDUCE=0
File Input Format Counters
  Bytes Read=139
  File Output Format Counters
    Bytes Written=113
id20120030@thienan:~/Desktop/WordCountTutorial$ jps
19224 NodeManager
17845 SecondaryNameNode
18088 ResourceManager
19515 Jps
19516 org.mortbay.jetty.Server
17645 DataNode
id20120030@thienan:~/Desktop/WordCountTutorial$ hadoop dfs -cat /WordCountTutorial/Output/*
WARNING: Use of this script to execute dfs is deprecated.
WARNING: Attempting to execute replacement 'hdfs dfs' instead.

2023-03-21 03:23:14,397 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Aquarius 1
Aries 2
Cancer 1
Capricorn 1
Gemini 3
Leo 1
Libra 1
Pisces 1
Sagittarius 3
Scorpio 1
Taurus 1
Virgo 2
id20120030@thienan:~/Desktop/WordCountTutorial$ 

```

Figure 3.8: Show the output using -cat command


The screenshot shows the "Nodes of the cluster" page from the Hadoop Web UI. The top navigation bar includes links for "Install Hadoop on Ubuntu", "DataNode Information", "How to run Word Count", and "Nodes of the cluster". The main content area is titled "Nodes of the cluster" and displays a table of node metrics.

	Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources	Reserved Resources	Physical Mem Used %	Physical Vcores Used %
Cluster Metrics	1	0	0	1	0	<memory:0 B, vCores:0>	<memory:8 GB, vCores:8>	<memory:0 B, vCores:0>	91	25
Cluster Nodes Metrics	1	0	0	0	0	0	0	0	0	0
Scheduler Metrics	1	0	0	0	0	0	0	0	0	0
Capacity Scheduler	[memory:mb (unit=M), vcores]				Minimum Allocation	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0	Maximum Cluster Application Priority	0
Sheduler					Maximum Allocation				Scheduler Busy %	
Node Labels	+ default-/default-rack	Node State	Node Address	Node HTTP Address	Last health-update	Health-report	Containers	Allocation Tags	Mem Used	Mem Avail
	RUNNING	thienan:34568	thienan:8042		Tue Mar 21 03:41:17 +0700 2023	0	0	0 B	8 GB	91
								Phys Mem Used %	Vcores Used %	Phys Vcores Used %
								0	8	32
								Version: 0		3.2.4

Below the table, there is a search bar and a link to "First, Previous, Next, Last".

Figure 3.9: Nodes of the cluster

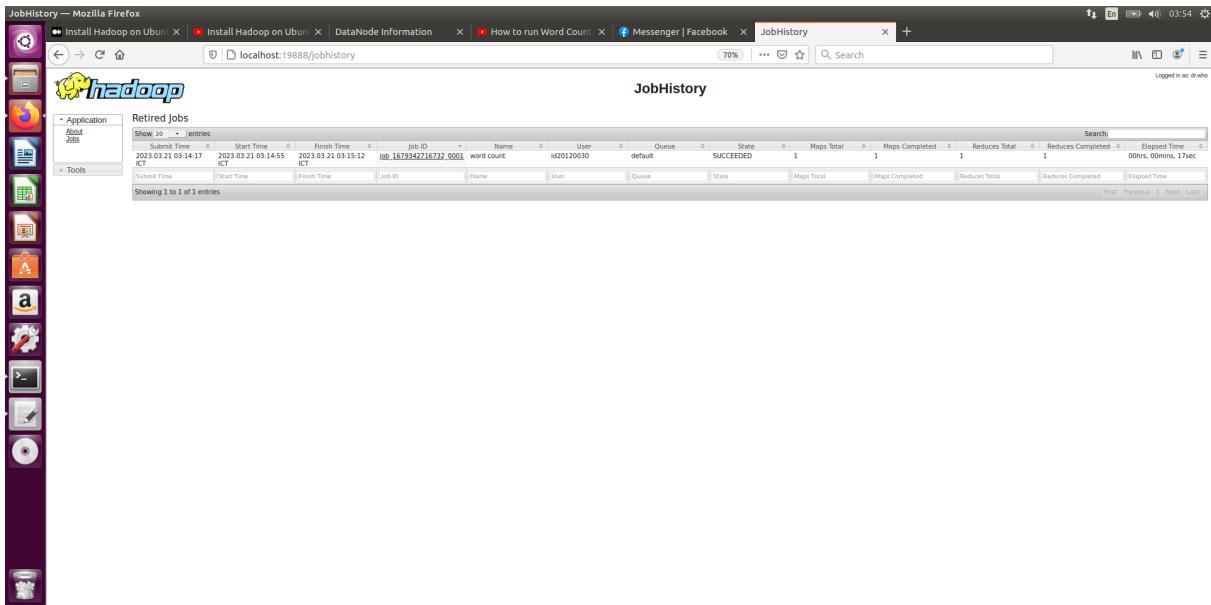


Figure 3.10: MapReduce Server

3.3 Member 3: 20120090 - Nguyen The Hoang

A screenshot of a terminal window titled "20120090@thehoang: ~/BigDataLabs/Lab_1". The window contains the following command-line session:

```
File Edit View Search Terminal Help
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop com.sun.tools.javac.Main WordCount.java
20120090@thehoang:~/BigDataLabs/Lab_1$ jar cf wc.jar WordCount*.class
20120090@thehoang:~/BigDataLabs/Lab_1$ dir
wc.jar wordcount WordCountsIntSumReducer.class WordCountsTokenizerMapper.class WordCount.class WordCount.java
20120090@thehoang:~/BigDataLabs/Lab_1$ _
```

Figure 3.11: Step 1: Create the WordCount v1.0 file as instructed in [^c]. Compile this file to the class and jar file

```
20120090@thehoang:~/BigDataLabs/Lab_1
File Edit View Search Terminal Help
20120090@thehoang:~/BigDataLabs/Lab_1$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as 20120090 in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [thehoang]
Starting resourcemanager
Starting nodemanagers
20120090@thehoang:~/BigDataLabs/Lab_1$ jps
23873 Jps
22722 DataNode
22931 SecondaryNameNode
23156 ResourceManager
23269 NodeManager
22604 NameNode
20120090@thehoang:~/BigDataLabs/Lab_1$ _
```

Figure 3.12: Step 2: Start the Hadoop on the Pseudo-distributed Mode

```
20120090@thehoang:~/BigDataLabs/Lab_1
File Edit View Search Terminal Help
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -mkdir /user
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -mkdir /user/sUSER
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -mkdir /user/sUSER/wordcount
20120090@thehoang:~/BigDataLabs/Lab_1$ echo "Hello World Bye World" | hadoop fs -put - /user/sUSER/wordcount/input/file01
20120090@thehoang:~/BigDataLabs/Lab_1$ echo "Hello World Bye World" | hadoop fs -put - /user/sUSER/wordcount/input/file02
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -rm /user/sUSER/wordcount/input/file02
Deleted /user/20120090/wordcount/input/file02
20120090@thehoang:~/BigDataLabs/Lab_1$ echo "Hello Hadoop Goodbye Hadoop" | hadoop fs -put - /user/sUSER/wordcount/input/file02
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -cat /user/sUSER/wordcount/*
cat: '/user/20120090/wordcount/input': Is a directory
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop fs -cat /user/sUSER/wordcount/input/*
Hello World Bye World
Hello Hadoop Goodbye Hadoop
20120090@thehoang:~/BigDataLabs/Lab_1$ _
```

Figure 3.13: Step 3: As instructed in [^c], create two input files in the distributed filesystem, in the input folder

```
20120090@thehoang:~/BigDataLabs/Lab_1
File Edit View Search Terminal Help
20120090@thehoang:~/BigDataLabs/Lab_1$ hadoop jar wc.jar WordCount /user/sUSER/wordcount/input /user/sUSER/wordcount/output
2023-03-06 13:19:21.029 INFO client.DefaultNoHARMFailoverProxyProvider: Connecting to ResourceManager at localhost/127.0.0.1:8032
2023-03-06 13:19:21.282 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
2023-03-06 13:19:21.295 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /tmp/hadoop-yarn/staging/20120090/.staging/job_16780083289713_0001
083289713_0001
2023-03-06 13:19:21.488 INFO input.FileInputFormat: Total input files to process : 2
2023-03-06 13:19:21.529 INFO mapreduce.JobSubmitter: number of splits:2
2023-03-06 13:19:21.696 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_16780083289713_0001
2023-03-06 13:19:21.696 INFO mapreduce.JobSubmitter: Executing with tokens: []
2023-03-06 13:19:21.811 INFO conf.Configuration: resource-types.xml not found
2023-03-06 13:19:21.812 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2023-03-06 13:19:21.984 INFO impl.YarnClientImpl: Submitted application application_16780083289713_0001
2023-03-06 13:19:22.031 INFO mapreduce.Job: The url to track the job: http://thehoang:8088/proxy/application_16780083289713_0001/
2023-03-06 13:19:22.032 INFO mapreduce.Job: Running job: job_16780083289713_0001
2023-03-06 13:19:28.099 INFO mapreduce.Job: Job job_16780083289713_0001 running in uber mode : false
2023-03-06 13:19:28.102 INFO mapreduce.Job: map 0% reduce 0%
2023-03-06 13:19:34.236 INFO mapreduce.Job: map 50% reduce 0%
2023-03-06 13:19:35.274 INFO mapreduce.Job: map 100% reduce 0%
2023-03-06 13:19:35.274 INFO mapreduce.Job: Job completed successfully
2023-03-06 13:19:35.274 INFO mapreduce.Job: User Wrote: 0
2023-03-06 13:19:35.274 INFO mapreduce.Job: Total Size: 0
2023-03-06 13:19:35.274 INFO mapreduce.Job: Record Count : 0
2023-03-06 13:19:35.274 INFO mapreduce.Job: File Count : 0
2023-03-06 13:19:35.274 INFO mapreduce.Job: Map Output Count : 0
2023-03-06 13:19:35.274 INFO mapreduce.Job: Reduce Output Count : 0
2023-03-06 13:19:35.274 INFO mapreduce.Job:   
20120090@thehoang:~/BigDataLabs/Lab_1$ _
```

Figure 3.14: Step 4: Run the MapReduce program

Application application_1678083289713_0001

Application Overview

User:	20120090
Name:	word count
Application Type:	MAPREDUCE
Application Tags:	
Application Priority:	0 (Higher Integer value indicates higher priority)
YarnApplicationState:	FINISHED
Queue:	default
FinalStatus Reported by AM:	SUCCEEDED
Started:	Mon Mar 06 13:19:21 +0700 2023
Launched:	Mon Mar 06 13:19:22 +0700 2023
Finished:	Mon Mar 06 13:19:39 +0700 2023
Elapsed:	17sec
Tracking URL:	History
Log Aggregation Status:	DISABLED
Application Timeout (Remaining Time):	Unlimited
Diagnostics:	
Unmanaged Application:	false
Application Node Label expression:	<Not set>
AM container Node Label expression:	<DEFAULT_PARTITION>

Application Metrics

Total Resource Preempted:	<memory:0, vCores:0>
Total Number of Non-AM Containers Preempted:	0
Total Number of AM Containers Preempted:	0
Resource Preempted from Current Attempt:	<memory:0, vCores:0>
Number of Non-AM Containers Preempted from Current Attempt:	0
Aggregate Resource Allocation:	61811 MB-seconds, 35 vcore-seconds
Aggregate Preempted Resource Allocation:	0 MB-seconds, 0 vcore-seconds

Nodes of the cluster

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	Total Resources	Reserved Resources	Physical Mem Used %	Physical Vcores Used %
1	0	0	1	0	<memory:0 B, vCores:0>	<memory:8 GB, vCores:8>	<memory:0 B, vCores:0>	59	25

Cluster Nodes Metrics

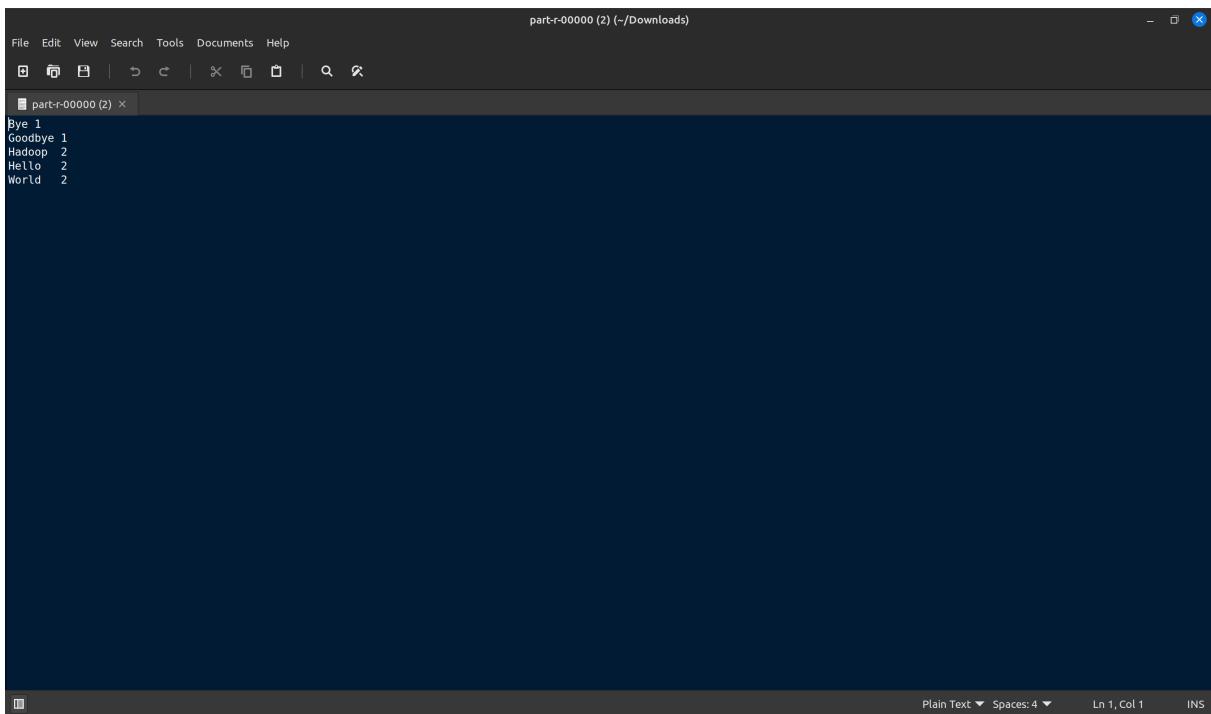
Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes	Rebooted Nodes	Shutdown Nodes
1	0	0	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation	Maximum Cluster Application Priority	Scheduler Busy %
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1024, vCores:1>	<memory:8192, vCores:4>	0	0

Nodes of the cluster

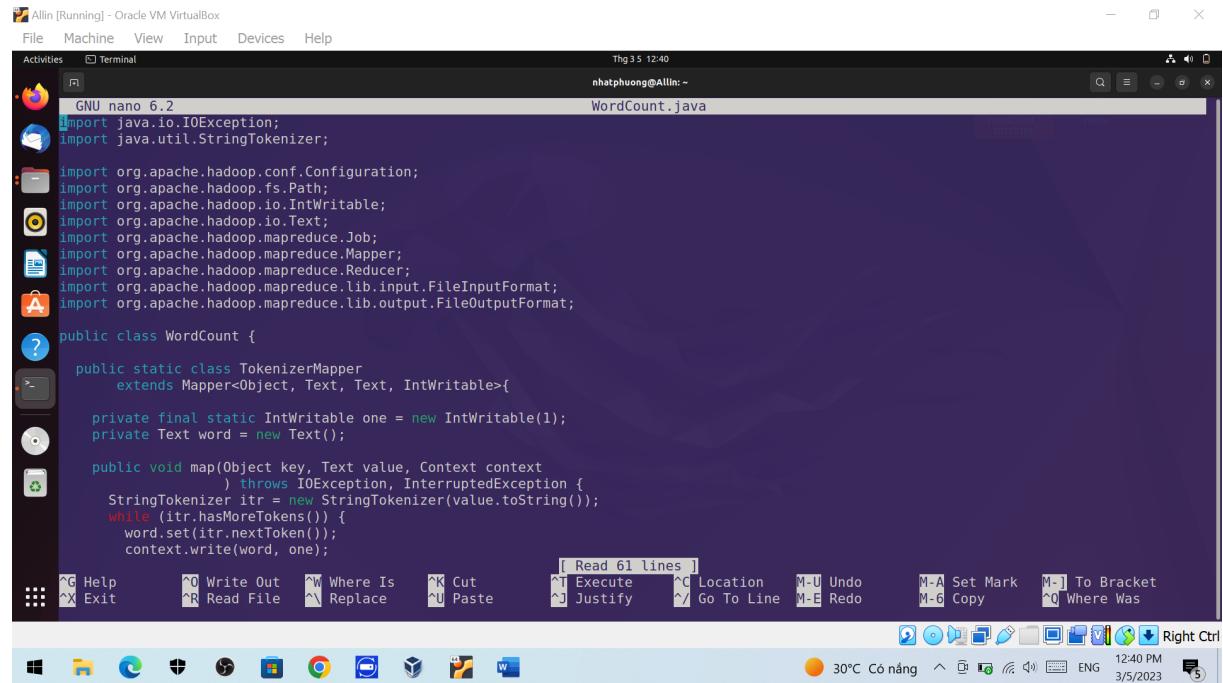
Node Labels	Rack	Node State	Node Address	Node HTTP Address	Last health-update	Health-report	Containers	Allocation Tags	Mem Used	Mem Avail	Phys Mem Used %	Vcores Used	Vcores Avail	Phys Vcores Used %	Version
/default-rack		RUNNING	thehoang:40673	thehoang:8042	Mon Mar 06 13:20:43 +0700 2023	0	0	0 B	8 GB	59	0	8	31	3.3.4	



A screenshot of a text editor window titled "part-r-00000 (2) (-/Downloads)". The window has a dark theme. The menu bar includes File, Edit, View, Search, Tools, Documents, and Help. The toolbar contains icons for New, Open, Save, Cut, Copy, Paste, Find, and Replace. The main text area contains the following two lines of text:

```
Plain Text ▾ Spaces: 4 ▾ Ln 1, Col 1 INS
bye 1
Goodbye 1
Hadoop 2
Hello 2
World 2
```

3.4 Member 4: 20120165 - Hong Nhat Phuong



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "WordCount.java" and the command "nano 6.2". The code in the terminal is:

```
GNU nano 6.2
import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class WordCount {
    public static class TokenizerMapper
        extends Mapper<Object, Text, Text, IntWritable>{
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();

        public void map(Object key, Text value, Context context
                       ) throws IOException, InterruptedException {
            StringTokenizer itr = new StringTokenizer(value.toString());
            while (itr.hasMoreTokens()) {
                word.set(itr.nextToken());
                context.write(word, one);
            }
        }
    }

    public static class TokenizerReducer
        extends Reducer<Text, IntWritable, Text, IntWritable> {
        private IntWritable result = new IntWritable();

        public void reduce(Text key, Iterable<IntWritable> values,
                           Context context) throws IOException, InterruptedException {
            int sum = 0;
            for (IntWritable val : values) {
                sum += val.get();
            }
            result.set(sum);
            context.write(key, result);
        }
    }
}
```

The terminal window also shows the status "Read 61 lines". Below the terminal is a standard Linux desktop toolbar with icons for file operations like Help, Exit, Write Out, Read File, Replace, Cut, Paste, Execute, Location, Undo, Set Mark, Copy, and Where Was.

Figure 3.15: Create WordCount.java

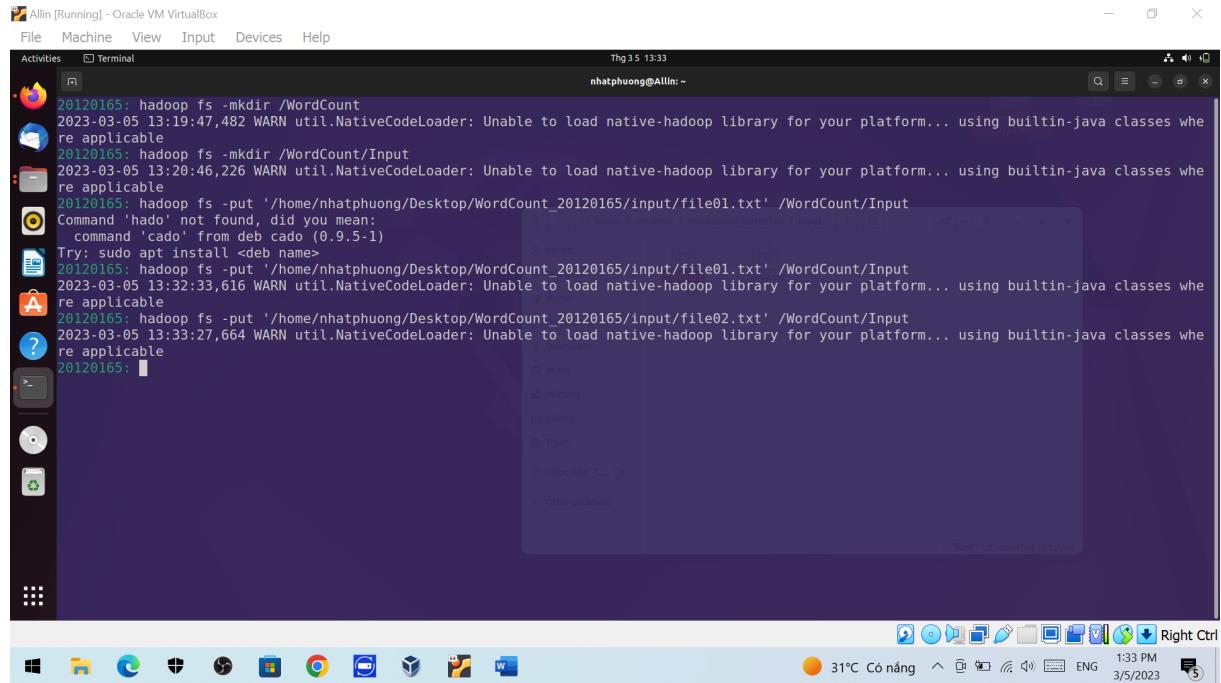


Figure 3.16: Create folder input and put file01.txt, file02.txt into input directory

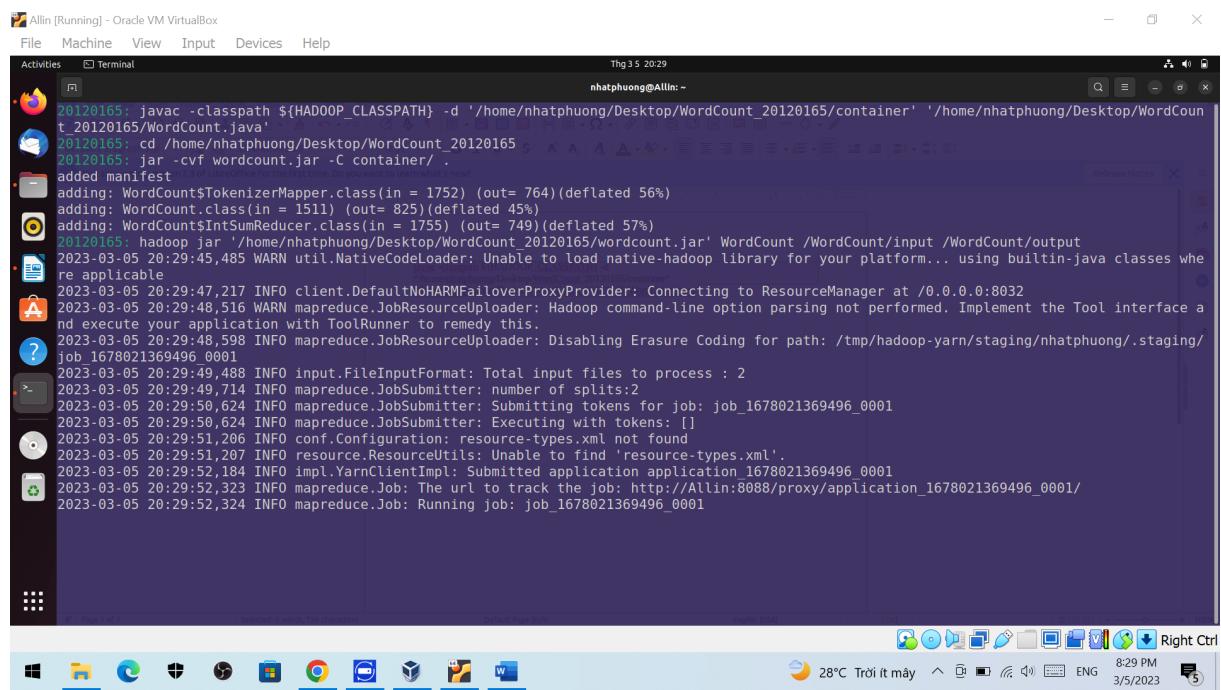
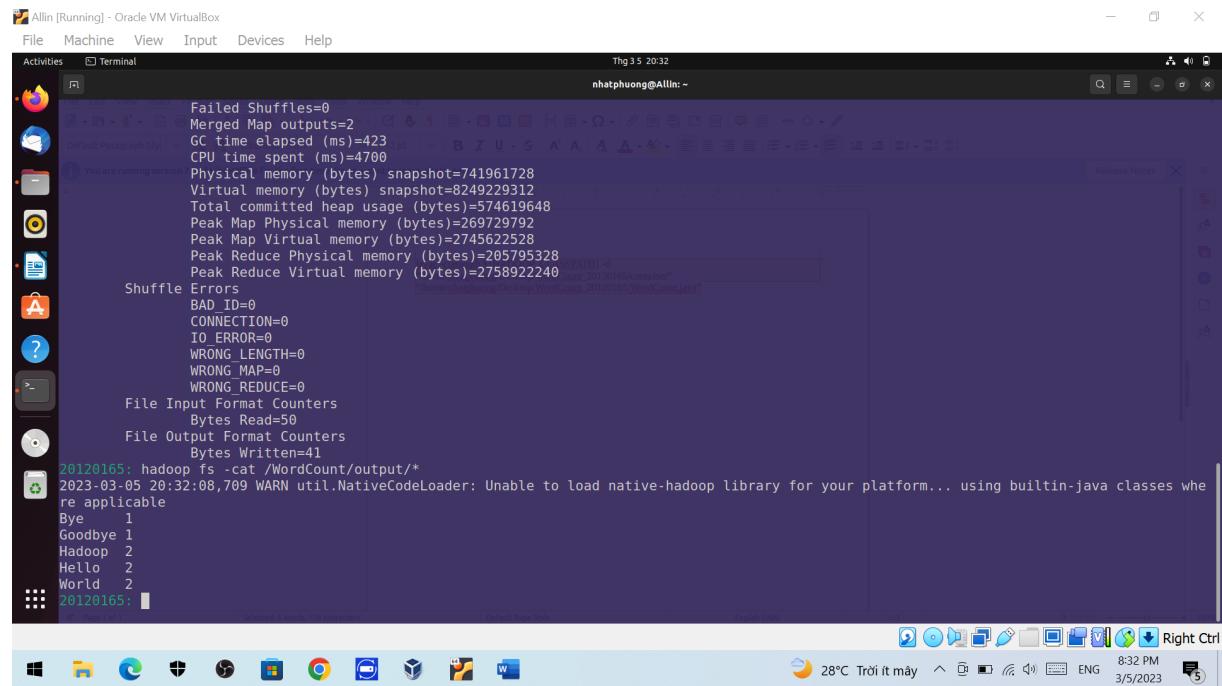


Figure 3.17: Compile WordCount.java, create a jar and run WordCount program



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "Allin [Running] - Oracle VM VirtualBox". The terminal content displays the output of a Hadoop WordCount job. The output includes various system statistics and the word count results:

```
Failed Shuffles=0
Merged Map outputs=2
GC time elapsed (ms)=423
CPU time spent (ms)=4700
Physical memory (bytes) snapshot=741961728
Virtual memory (bytes) snapshot=8249229312
Total committed heap usage (bytes)=574619648
Peak Map Physical memory (bytes)=269729792
Peak Map Virtual memory (bytes)=2745622528
Peak Reduce Physical memory (bytes)=205795328
Peak Reduce Virtual memory (bytes)=2758922240
Shuffle Errors
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters
Bytes Read=50
File Output Format Counters
Bytes Written=41
20120165: hadoop fs -cat /WordCount/output/*
2023-03-05 20:32:08,709 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Bye 1
Goodbye 1
Hadoop 2
Hello 2
World 2
20120165:
```

Figure 3.18: Show the output using -cat command

4 Bonus

4.1 Bad Relationship

The **Unhealthy_relationship.java** program includes two phases:

1. The Mapper implementation (UnhealthyRelationshipMapper class), via the `map()` method, processes the input files one line at a time, read them as `Text` type object. It then splits the line into two words (separated by whitespace): `word_1` and `word_2`. It then emits two key-value pair of `< word_1, 1 >` and `< word_2, -1 >`. The first pair means that: `word_1` has one positive relation with other objects, so it will contribute value '1' to its final result. On the other hand, the second pair means that: `word_2` has one negative relation with other objects, so it will contribute value '-1' to its final result. In short, by considering each word's position in the input files, we can count each positive and negative relation of each word, and then we aggregate these values to know the final relation of each word.

So the `map()` emits the output has type `<Text, IntWritable>`. This pair of key-value will be grouped by key and all the value of one key will be put into one list.

2. The Reducer implementation (UnhealthyRelationshipReducer), via the `reduce()` method, receive the key-value pairs `<word, [list_of_1_or_-1]>` for each distinct word. The `reduce()` method just aggregate all the value in `[list_of_1_or_-1]`. The final value sum will be used to determine if that word has negative/positive/equality relation. In specific:

- `sum > 0`: That word has positive relation
- `sum = 0`: That word has equality relation
- `sum < 0`: That word has negative relation

The `reduce()` method then just emit the final output for each word in the form key-value `<word, type_of_relation>`, with type `<Text, Text>`.

5 References

[^b] T. White, “Appendix A. Installing Apache Hadoop,” in Hadoop: The definite guide, 2012, p. 680.

[^c] Apache Hadoop, “MapReduce Tutorial,” Jul. 29, 2022. <https://hadoop.apache.org/docs/stable/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html>.