CEG-7370-01

Distributed Computing

Fall 2015

Project #2

Write-up File

DHRUVKUMAR NAVINCHANDRA PATEL

U00791652

Hadoop Map/Reduce – An Open-Source Software for Reliable, Scalable, and Distributed Processing of Big Data

1) **What is Hadoop?**
   - Hadoop is an open-source framework that allows to interact with the large data sets using single node or multi node clusters of computer using programming models.
   - It is used to develop a model from single server to thousands of computers for storage and computation.
   - Hadoop is a framework used to access big datasets which is not be processed using general computing techniques.
   - Big Data is a data coming from different devices and applications. For example, Social Media data, Search Engine Data, Sensor Data and menu more.
   - Using Hadoop one can capturing data, storage data, searching and sharing a data and transferring, analysis, presenting those data.
   - Hadoop runs applications using the MapReduce Algorithm. Word Count Example is tested using Hadoop Framework.
   - Hadoop Framework is written in Java programming language. It includes following Four Modules.

   1) **Hadoop Common**: These libraries used for filesystem and OS level which contains java file and scripts to start Hadoop.
   2) **Hadoop YARN**: This is a framework for job scheduling and cluster resource manager.
   3) **HDFS**: - Hadoop Distributed File System that provides good output to access application data.
   4) **Hadoop MapReduce:** This is YARN-based system is used to concurrent processing of large data sets.

Following are the importance of java process display when jps command hit.

1) Resource Manager
2) NameNode
3) DataNode
4) Jps
5) SecondaryNameNode
6) NodeManager

1) **Resource Manager: -** Hadoop Resource Manager is used for allow user to collect information about status on the cluster, metrics on the cluster, scheduler information and information about nodes and applications on cluster.
   It is used for tracking the resources in a cluster and scheduling applications.
2) **NameNode :-** Hadoop NameNode is a work like a master server. It used to manage Hadoop file system namespace. Client's access to file and NameNode is responsible for file system operation renaming, closing, opening files and directories
3) **DataNode:-** Every node in a cluster has a DataNode . It used for Data Storage in a System. It is responsible for read and write operation of the system. That perform Block creation, deletion and Replication according to instruction.
4) **SecondaryNameNode:-** It is a backup and helper node of NameNode. It put checkpoint in file system which would be help NameNode to execute flexible.
5) **NodeManager:-** It is used for individual compute node in Hadoop cluster. It used for keep up to date information of Resource Manager , Monitoring Memory and cpu usage.

Hadoop 2.6.0 installation:

1) **Configured Linux Ubuntu 64 bit in Virtual Box.**
2) **Installed Java in Ubuntu.**

```
Oracle JRE 8 browser plugin installed
Setting up gsfonts-x11 (0.22) ...
dhruv@dhruv-VirtualBox:~$ java -version
java version "1.8.0_60"
Java(TM) SE Runtime Environment (build 1.8.0_60-b27)
Java HotSpot(TM) 64-Bit Server VM (build 25.60-b23, mixed mode)
dhruv@dhruv-VirtualBox:~$
```

Installed Oracle Java version 1.8.0_60 for eclipse configuration.

3) Created hduser in a new group for Hadoop installation.

```
dhruv@dhruv-VirtualBox:~$ sudo adduser --ingroup hadoop hduser
Adding user `hduser' ...
Adding new user `hduser' (1001) with group `hadoop' ...
Creating home directory `/home/hduser' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for hduser
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
dhruv@dhruv-VirtualBox:~$ sudo adduser hduser sudo
Adding user `hduser' to group `sudo' ...
Adding user hduser to group sudo
Done.
dhruv@dhruv-VirtualBox:~$ sudo su hduser
hduser@dhruv-VirtualBox:/home/dhruv$
```

4) Installation and Configuring SSH

Generated SSH key generation

```
Setting up ssh-import-id (3.21-0ubuntu1) ...
Processing triggers for libc-bin (2.19-0ubuntu6.6) ...
Processing triggers for ureadahead (0.100.0-16) ...
Processing triggers for ufw (0.34~rc-0ubuntu2) ...
hduser@dhruv-VirtualBox:~$ ssh-keygen -t rsa -P ""
Generating public/private rsa key pair.
Enter file in which to save the key (/home/hduser/.ssh/id_rsa):
Created directory '/home/hduser/.ssh'.
Your identification has been saved in /home/hduser/.ssh/id_rsa.
Your public key has been saved in /home/hduser/.ssh/id_rsa.pub.
The key fingerprint is:
ee:f5:36:8e:43:ca:6a:27:66:6c:37:ea:1b:b1:68:91 hduser@dhruv-VirtualBox
The key's randomart image is:
+--[ RSA 2048]----+
|                 |
|                 |
|                 |
|     .           |
|    E . S        |
|     o +  .      |
|    o.o..o.      |
|   .  Bo*..oo    |
|     *=B...+o.   |
+-----------------+
```

Tested SSH Localhost

```
hduser@dhruv-VirtualBox:~$ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_ke
ys
hduser@dhruv-VirtualBox:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is cf:d2:53:d7:a9:0e:58:f2:d1:47:e5:68:87:fa:82:04.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

5) Installing and Configuring Hadoop

Downloading and extracting Hadoop

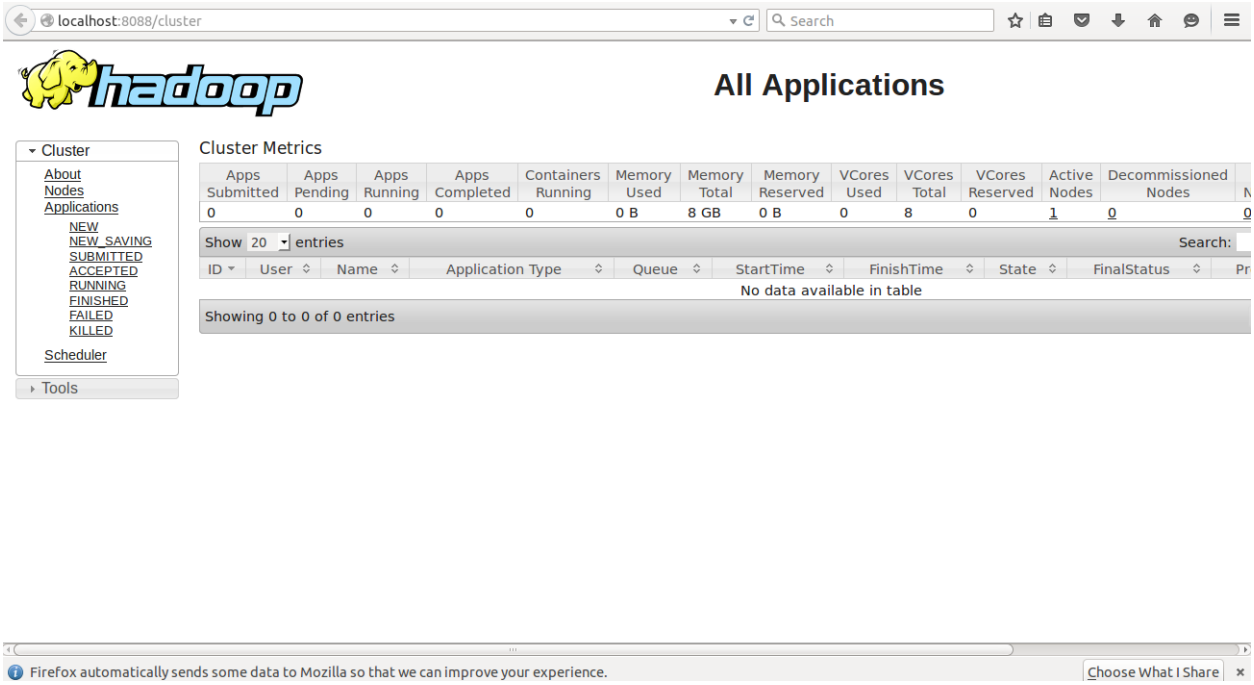
Use different steps including edit different files.

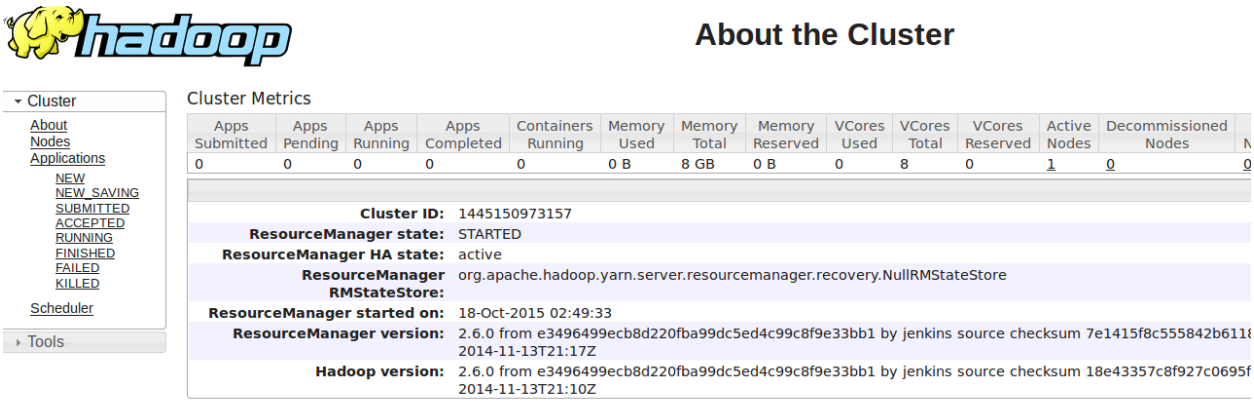
I got one error while updating java path in Hadoop-env.sh file. Must include export before JAVA_HOME variable.

Following screenshot describing five java processes when one hit jps command after successfully installing Hadoop.

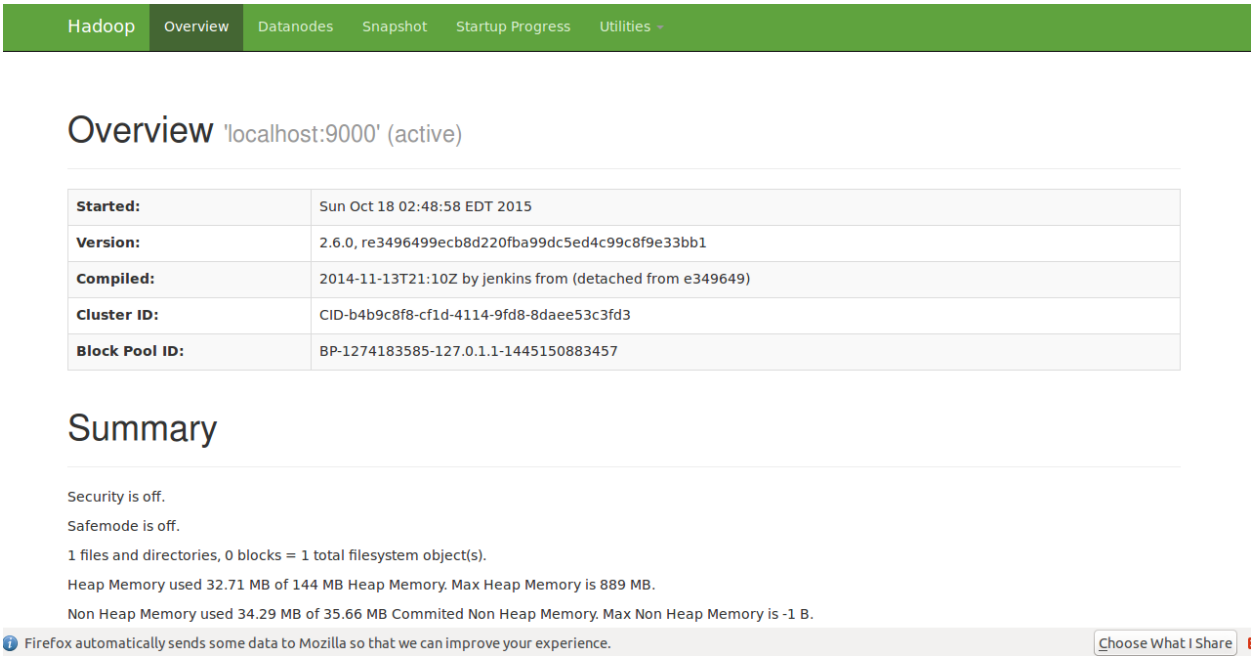
```
.
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hd
user-secondarynamenode-dhruv-VirtualBox.out
15/10/18 02:49:17 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
hduser@dhruv-VirtualBox:/usr/local/hadoop/sbin$ start-yarn.sh
starting yarn daemons
starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-hduser-resource
manager-dhruv-VirtualBox.out
localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-hduser-n
odemanager-dhruv-VirtualBox.out
hduser@dhruv-VirtualBox:/usr/local/hadoop/sbin$ jps
6098 ResourceManager
5620 NameNode
5765 DataNode
6519 Jps
5945 SecondaryNameNode
6222 NodeManager
```

6) Open Resource Manager using localhost 8088



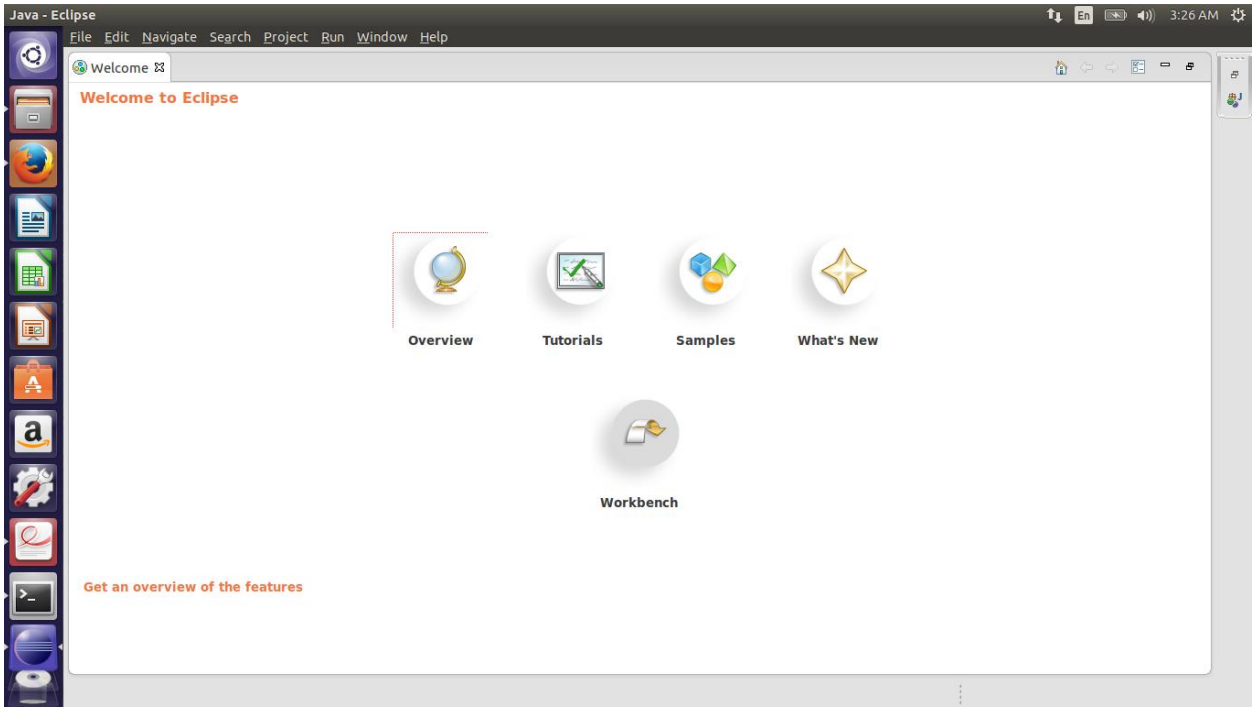


7) Open Name Node using Localhost 50070



8) Run the Eclipse standard Kepler SR2 64 bit from terminal

I got an error message because I installed Ubuntu 64 bit operating system and I was trying to install eclipse 32 bit. After that I solved that error and install eclipse Kepler sr2 package 64 bit which is compatible with Ubuntu 64 bit operating system.

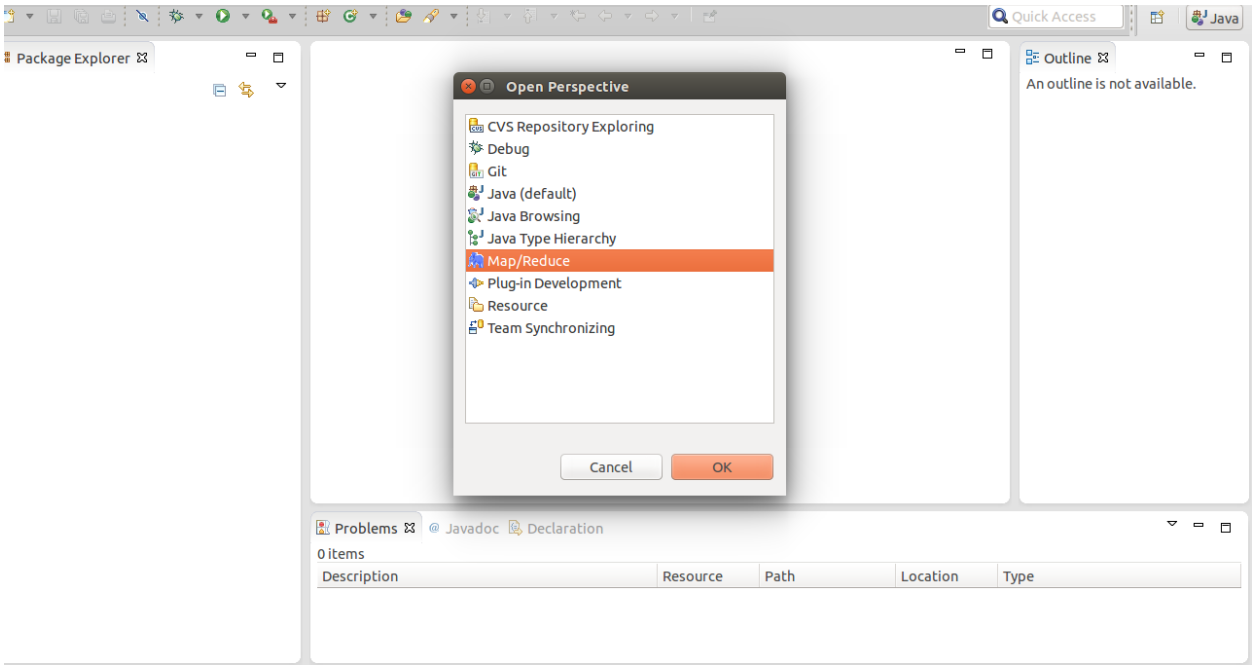


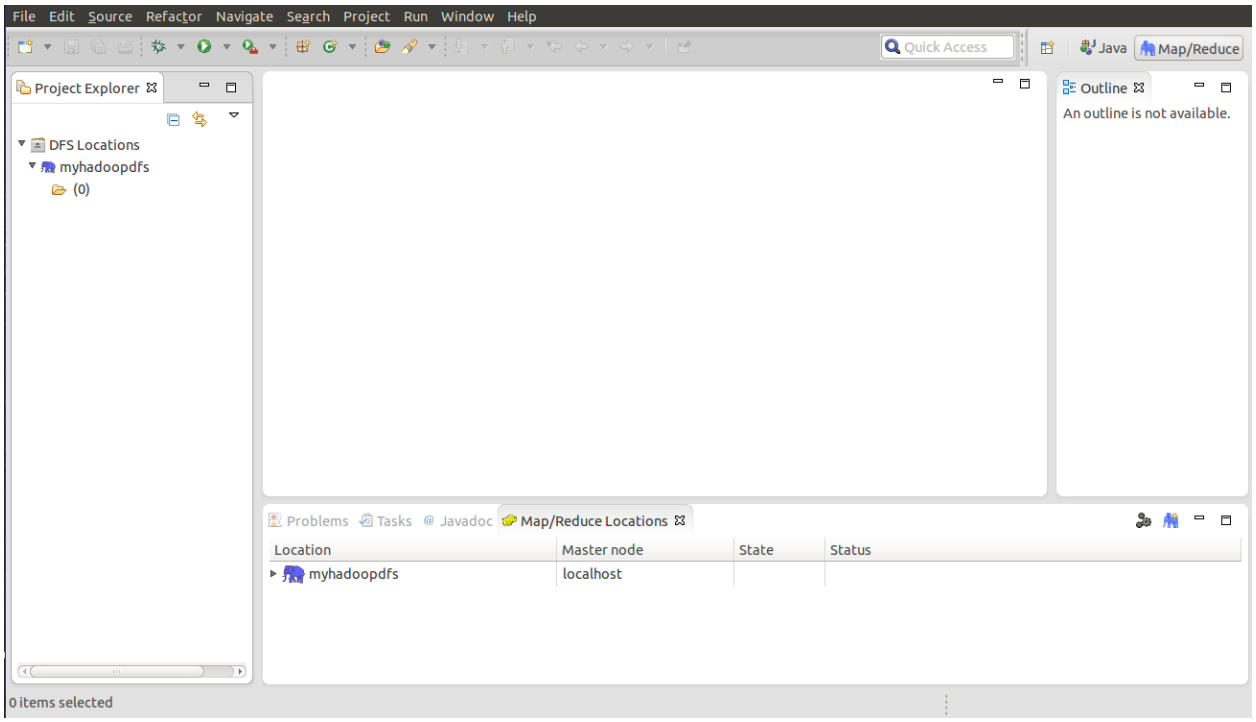
9) Installation of Eclipse Plugin

```
o /home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/build/contrib/eclipse-plu
gin/lib/guava-11.0.2.jar
     [copy] Copying 1 file to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-mast
er/build/contrib/eclipse-plugin/lib
     [copy] Copying /usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.6.0.
jar to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/build/contrib/eclips
e-plugin/lib/hadoop-auth-2.6.0.jar
     [copy] Copying 1 file to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-mast
er/build/contrib/eclipse-plugin/lib
     [copy] Copying /usr/local/hadoop/share/hadoop/common/lib/netty-3.6.2.Final.
jar to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/build/contrib/eclips
e-plugin/lib/netty-3.6.2.Final.jar
     [copy] Copying 1 file to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-mast
er/build/contrib/eclipse-plugin/lib
     [copy] Copying /usr/local/hadoop/share/hadoop/common/lib/htrace-core-3.0.4.
jar to /home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/build/contrib/eclips
e-plugin/lib/htrace-core-3.0.4.jar
      [jar] Building jar: /home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/b
uild/contrib/eclipse-plugin/hadoop-eclipse-plugin-2.6.0.jar

BUILD SUCCESSFUL
Total time: 3 minutes 17 seconds
hduser@dhruv-VirtualBox:/home/dhruv/Downloads/hadoop2x-eclipse-plugin-master/src
/contrib/eclipse-plugin$
```

Create MapReduce prospective in eclipse and create myhadoopdfs directory.





10) Now I did tasks and try to run Word Count Program

Save the source code of word count program in WordCount.java. Create a wordcountclasses folder for classes. To compile the java file use javac command to generate class files. **Create wordcount.jar file in command line**

Create an Input Directory under hdfs.

```
        at java.lang.reflect.Method.invoke(Method.java:497)
        at org.eclipse.equinox.launcher.Main.invokeFramework(Main.java:636)
        at org.eclipse.equinox.launcher.Main.basicRun(Main.java:591)
        at org.eclipse.equinox.launcher.Main.run(Main.java:1450)
        at org.eclipse.equinox.launcher.Main.main(Main.java:1426)
log4j:WARN No appenders could be found for logger (org.apache.hadoop.security.au
thentication.util.KerberosName).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more in
fo.
Oct 18, 2015 3:51:10 AM org.apache.hadoop.util.NativeCodeLoader <clinit>
WARNING: Unable to load native-hadoop library for your platform... using builtin
-java classes where applicable
hduser@dhruv-VirtualBox:/usr/local/bin$ cd
hduser@dhruv-VirtualBox:~$ cd /usr/local/hadoop/bin
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ hadoop dfs -mkdir /input
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

15/10/18 03:55:53 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
```

Create a hello.txt file and put under input directory

```
15/10/18 03:55:53 WARN util.NativeCodeLoader: Unable to load native-hadoop libra
ry for your platform... using builtin-java classes where applicable
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ touch hello.txt
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ ls
container-executor  hdfs       mapred      test-container-executor
hadoop              hdfs.cmd   mapred.cmd  yarn
hadoop.cmd          hello.txt  rcc         yarn.cmd
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$
```

Create a wordcountclasses directory for classes and WordCount.java file

```
hduser@dhruv-VirtualBox:~$ sudo nano ~/.bashrc
hduser@dhruv-VirtualBox:~$ source ~/.bashrc
hduser@dhruv-VirtualBox:~$ cd /usr/local/hadoop/bin
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ ls
container-executor  hdfs.cmd    rcc                      WordCount.java~
hadoop              hello.txt   test-container-executor  yarn
hadoop.cmd          mapred      wordcountclasses         yarn.cmd
hdfs                mapred.cmd  WordCount.java
```

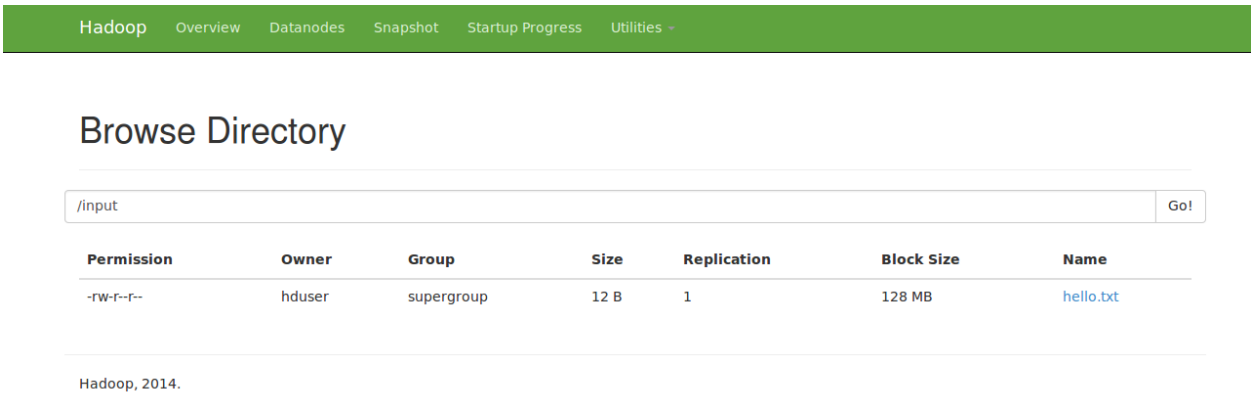
Compile WordCount.java

```
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ javac -classpath ${HADOOP_CLASSPA
TH} -d wordcountclasses/WordCount.java
javac: directory not found: wordcountclasses/WordCount.java
Usage: javac <options> <source files>
use -help for a list of possible options
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ javac -classpath ${HADOOP_CLASSPA
TH} -d wordcountclasses WordCount.java
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ cd wordcountclasses/
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin/wordcountclasses$ ls
WordCount.class  WordCount$IntSumReducer.class  WordCount$TokenizerMapper.class
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin/wordcountclasses$
```

wordcount.jar

```
cd..: command not found
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin/wordcountclasses$ cd ..
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ jar -cvf wordcount.jar -C wordcou
ntclasses/ .
added manifest
adding: WordCount$IntSumReducer.class(in = 1739) (out= 739)(deflated 57%)
adding: WordCount.class(in = 1491) (out= 814)(deflated 45%)
adding: WordCount$TokenizerMapper.class(in = 1736) (out= 754)(deflated 56%)
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ ls
container-executor  hdfs.cmd    rcc                     WordCount.java
hadoop              hello.txt   test-container-executor WordCount.java~
hadoop.cmd          mapred      wordcountclasses        yarn
hdfs                mapred.cmd  wordcount.jar           yarn.cmd
```

11) hello.txt is an input file it contains word abc 3 times and run wordcount.jar and output will be store into output directory.



```
                Total time spent by all maps in occupied slots (ms)=1890
                Total time spent by all reduces in occupied slots (ms)=2387
                Total time spent by all map tasks (ms)=1890
                Total time spent by all reduce tasks (ms)=2387
                Total vcore-seconds taken by all map tasks=1890
                Total vcore-seconds taken by all reduce tasks=2387
                Total megabyte-seconds taken by all map tasks=1935360
                Total megabyte-seconds taken by all reduce tasks=2444288
        Map-Reduce Framework
                Map input records=1
                Map output records=3
                Map output bytes=24
                Map output materialized bytes=16
                Input split bytes=102
                Combine input records=3
                Combine output records=1
                Reduce input groups=1
                Reduce shuffle bytes=16
                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)=93
                CPU time spent (ms)=1060
                Physical memory (bytes) snapshot=437940224
                Virtual memory (bytes) snapshot=3838742528
                Total committed heap usage (bytes)=302514176
        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=12
        File Output Format Counters
                Bytes Written=6
```

Output inside output directory

```
        File Input Format Counters
                Bytes Read=12
        File Output Format Counters
                Bytes Written=6
hduser@dhruv-VirtualBox:/usr/local/hadoop/bin$ hdfs dfs -cat /output/*
15/10/18 16:34:16 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applica
ble
abc     3
```

Hadoop Overview Datanodes Snapshot Startup Progress Utilities

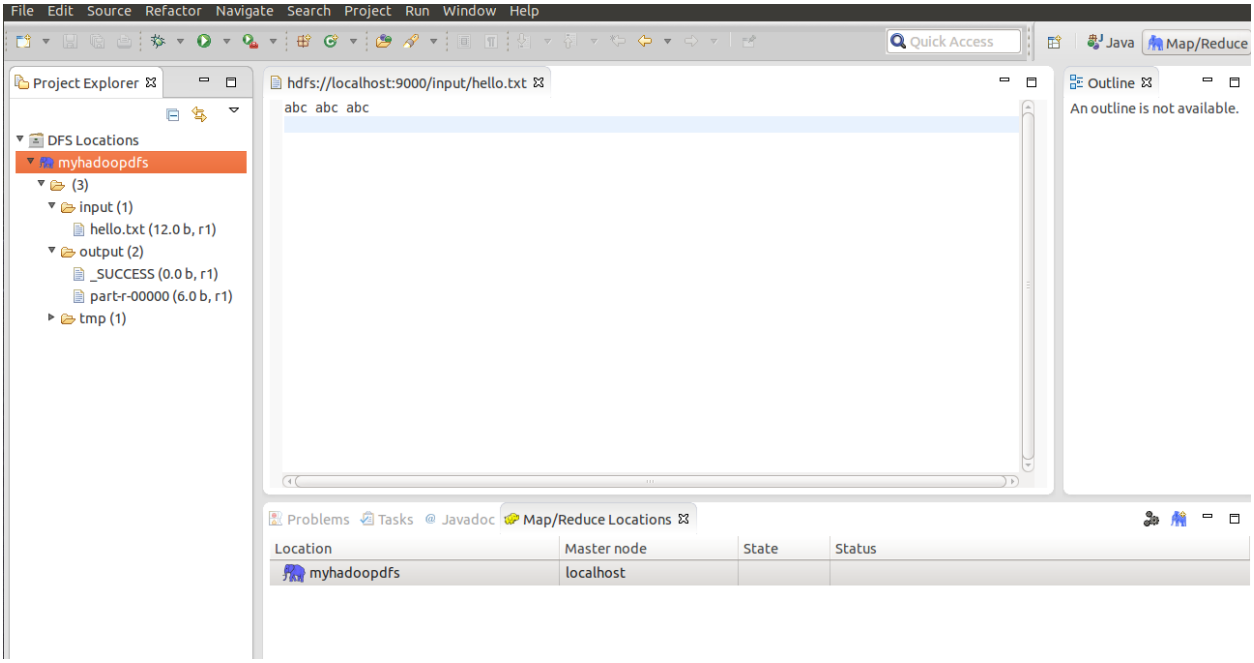
Browse Directory

/output Go!

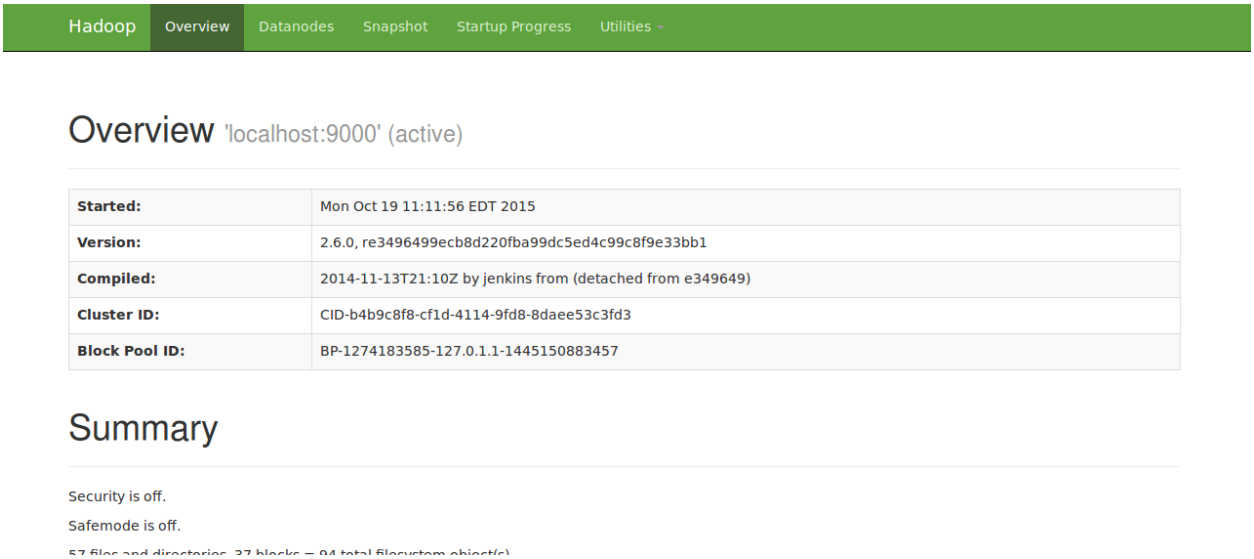
Permission	Owner	Group	Size	Replication	Block Size	Name
-rw-r--r--	hduser	supergroup	0 B	1	128 MB	_SUCCESS
-rw-r--r--	hduser	supergroup	6 B	1	128 MB	part-r-00000

Hadoop, 2014.

Eclipse Output after executing word count program



12) cluster summery after executing word count program using Localhost 50070



Configured Capacity:	25.82 GB
DFS Used:	1.11 MB
Non DFS Used:	6.73 GB
DFS Remaining:	19.09 GB
DFS Used%:	0%
DFS Remaining%:	73.92%
Block Pool Used:	1.11 MB
Block Pool Used%:	0%
DataNodes usages% (Min/Median/Max/stdDev):	0.00% / 0.00% / 0.00% / 0.00%
Live Nodes	1 (Decommissioned: 0)
Dead Nodes	0 (Decommissioned: 0)
Decommissioning Nodes	0
Number of Under-Replicated Blocks	16

NameNode Journal Status

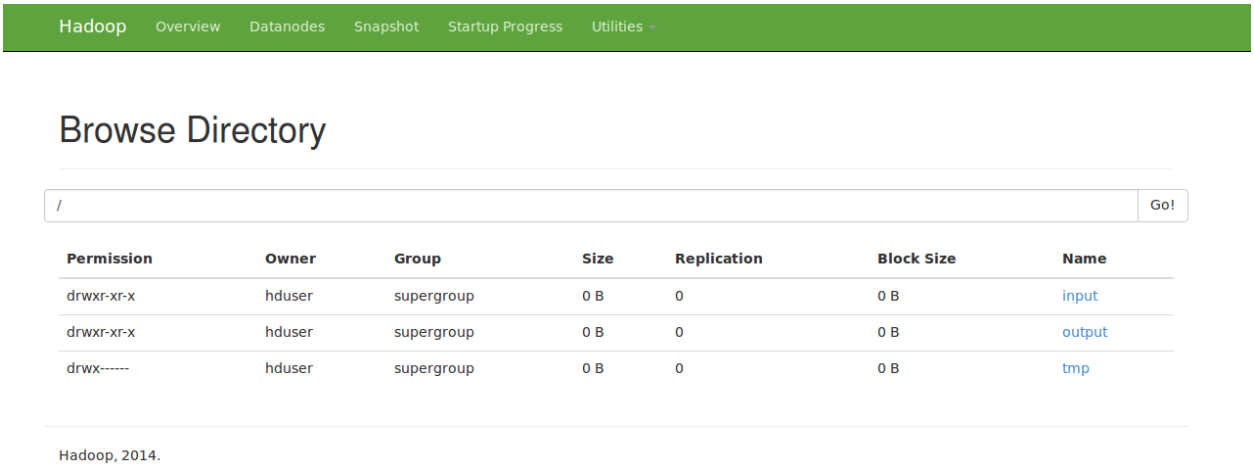
Current transaction ID: 364

Journal Manager	State
FileJournalManager(root=/usr/local/hadoop_tmp/hdfs/namenode)	EditLogFileOutputStream(/usr/local/hadoop_tmp/hdfs/namenode/current/edits_inprogress_0000000000000000364)

NameNode Storage

Storage Directory	Type	State
/usr/local/hadoop_tmp/hdfs/namenode	IMAGE_AND_EDITS	Active

Display Browse Directory under utilities.



I downloaded hello.txt from input directory and output under output directory.

Along With Write-up I included following files

1) Write-up contains all explanation and screen shots of Hadoop 2.6.0.
2) wordcount.jar file
3) Files under input and Output Directory
4) Cluster summary job and all the other screenshots included in Write-up file.