**Task 1: Setting up the Hadoop environment on the Vlab cloud**

**Single Node Setup**

The team followed the following steps to set up a pseudo-distributed, single-node Hadoop cluster backed by the Hadoop Distributed File System, on Ubuntu Linux machine.

1. Installed Java 1.7 as Hadoop requires a working Java 1.5+ installation.

2. Added a dedicated Hadoop system user - ‘hduser’

3. Configured SSH access to localhost for the user- ‘hduser’

4. Installed hadoop package and made ‘hduser’ the owner of the files.

5. updated the configuration files appropriately which includes changes to JAVA\_HOME variable.

6. Formatted the HDFS implemented on top of the local filesystem via namenode

7. Started a single node cluster

8. Ensured that Hadoop is listening on the configured ports.

9. Used three ebooks from Project Gutenberg to run the wordcount example. The ebooks are available in the following links

<http://www.gutenberg.org/ebooks/20417>

<http://www.gutenberg.org/ebooks/5000>

<http://www.gutenberg.org/ebooks/4300>

10. Copied the local data to HDFS

11. Created an executable jar from the wordcount java project, and ran the mapreduce job from Command Line Interface

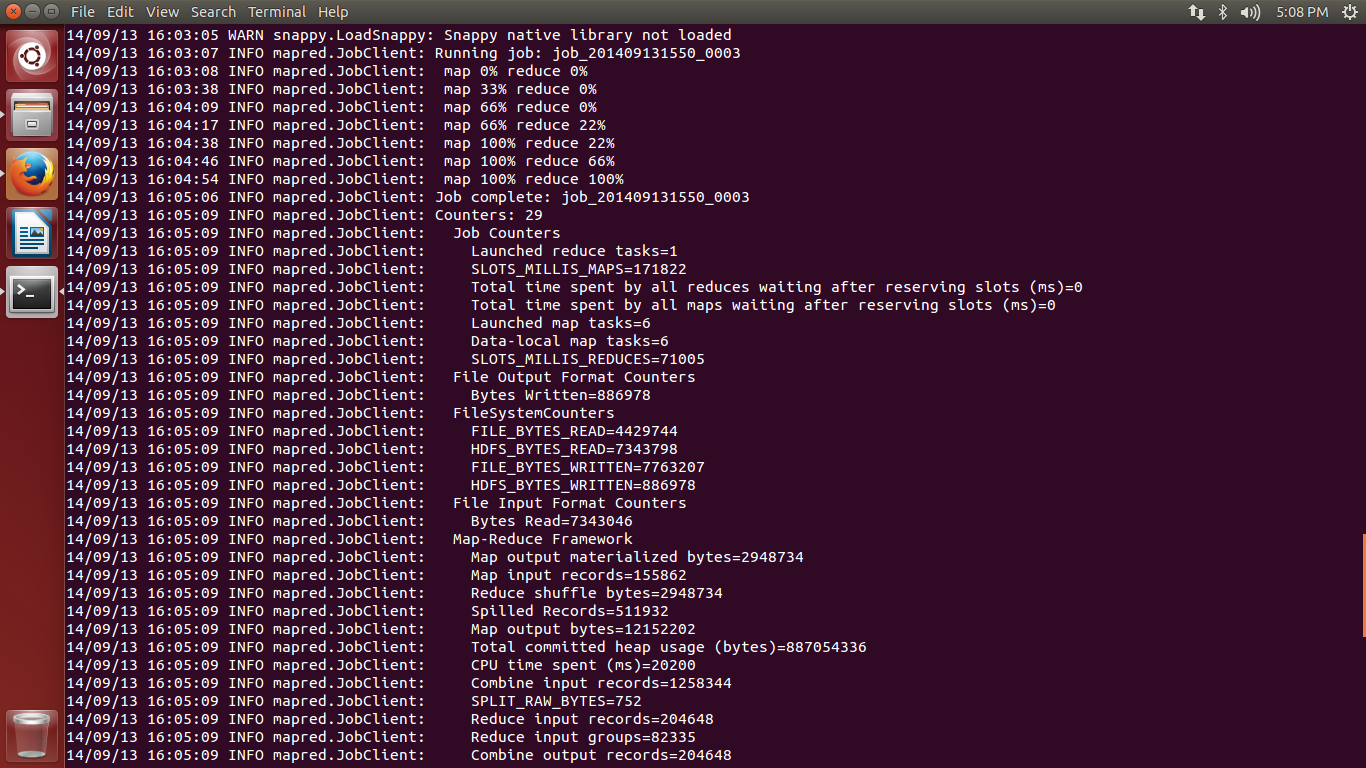
13. Observed the concise information provided about the hadoop cluster summary using the below web interfaces.

<http://localhost:50070/>- web UI of the NameNode daemon

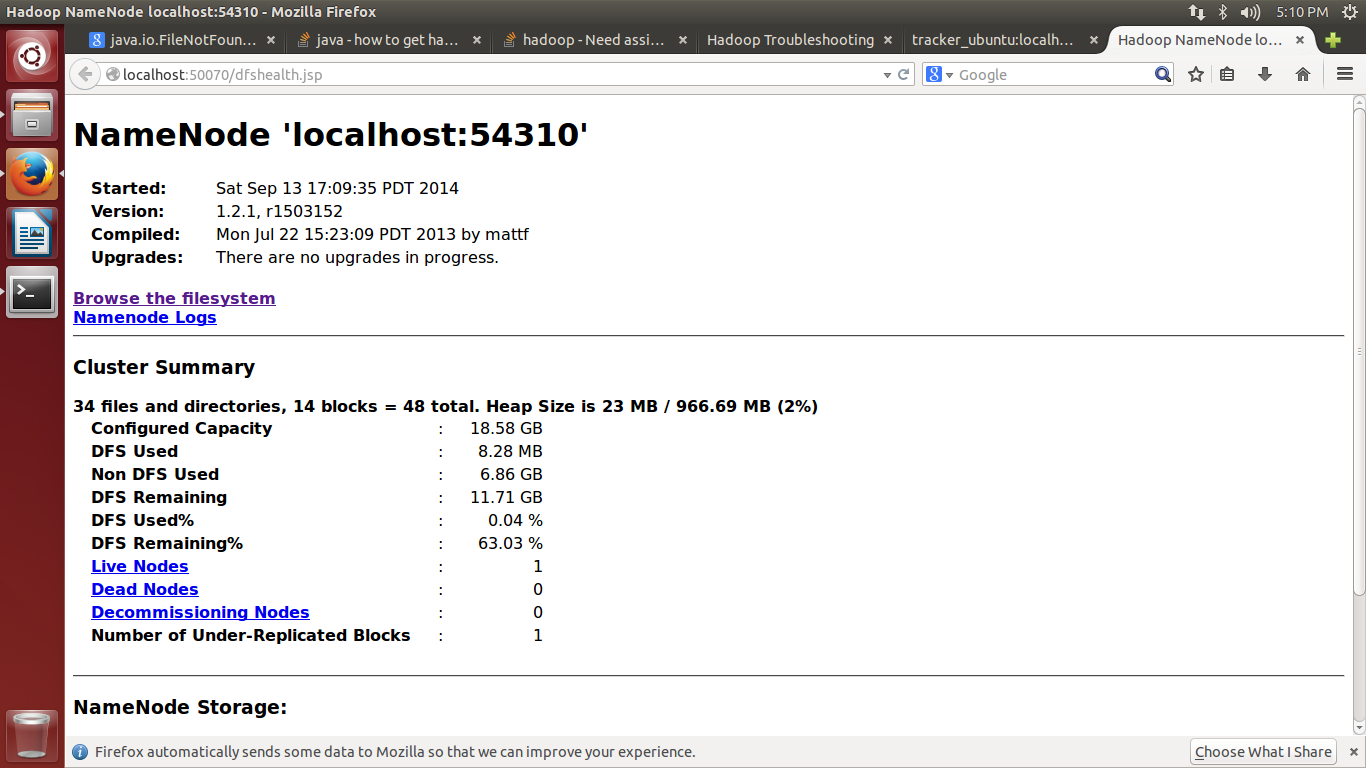
<http://localhost:50030/> – web UI of the JobTracker daemon

<http://localhost:50060/> – web UI of the TaskTracker daemon

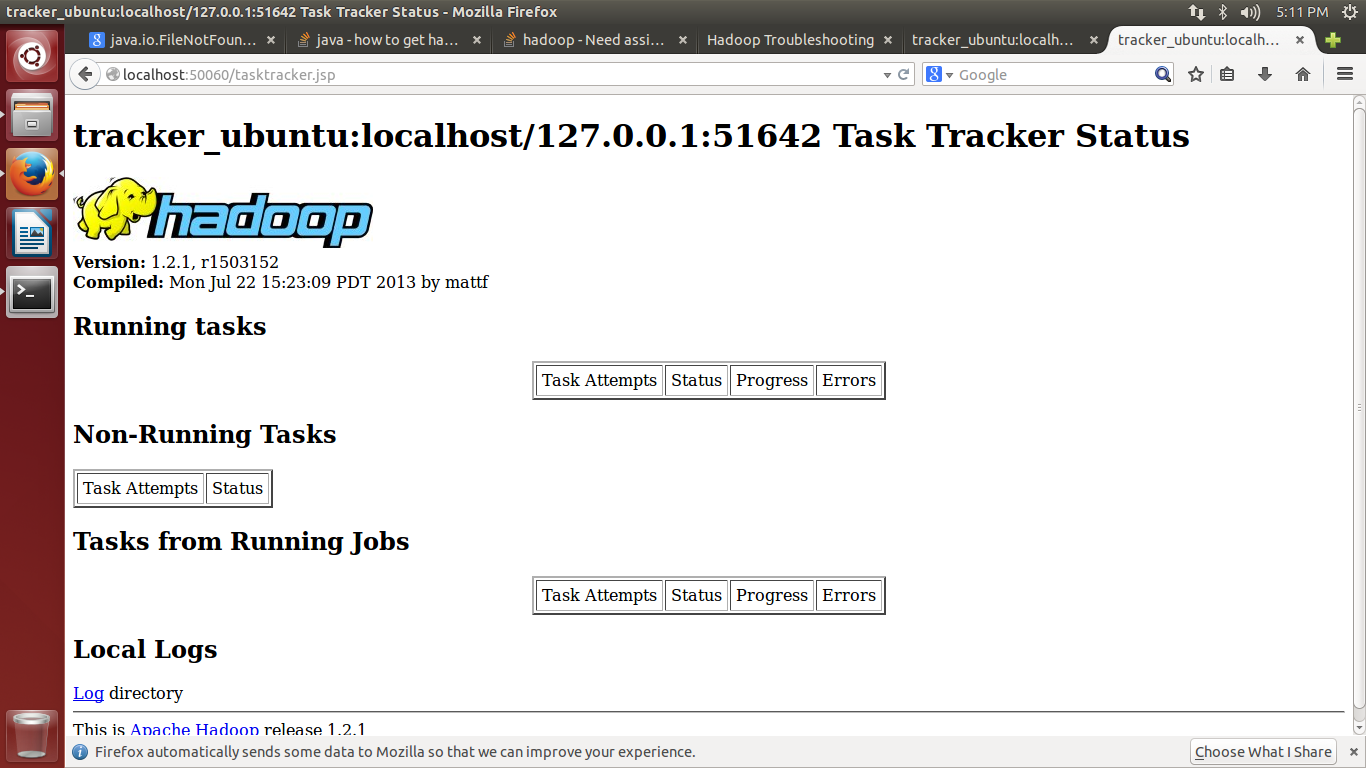
Please find screenshots of the results for running the word count example using hadoop.



Screenshot of Namenode Summary



Screenshot of Task Tracker Status



**Multi-Node Setup**

The following steps were followed to set up a pseudo-distributed, multi-node Hadoop cluster backed by the Hadoop Distributed File System, on Ubuntu Linux machine

1. Two single node Hadoop clusters were setup on VM 591103 and 591102
2. To ensure that both the machines are reachable over the network, we assigned IP addresses on a common network and updated the /etc/hosts file on both the machines accordingly.
3. Tested if ‘hduser’ on master is able to connect to ‘hduser’ on the slave node via the password-less login. Added the ‘hduser’ on master node`s public key to the ‘authorized\_keys’ file of the slave node.
4. Saved the slave’s host key fingerprint to the hduser@master’s ‘known\_hosts’ file.
5. Tested the SSH setup by connecting master node to itself and connecting master node to the slave node using the below commands.

>> ssh master

>> ssh slave

1. Configured the master node to run the “master” daemons namely –NameNode for HDFS layer and ResourceManager for MapReduce processing layer
2. Configured both the nodes to run the “slave” daemons namely- DataNode for HDFS layer and NodeManager for MapReduce processing layer.
3. On master node, we defined the machine on which Hadoop will start the NameNode and ResourceManager in ‘conf/masters’ file.
4. On master node, we also defined the machine on which Hadoop will run the DataNode and NodeManager in ‘conf/slaves’ file.
5. Changed the configuration file ‘conf/core-site.xml’, ‘conf/mapred-site.xml’ and ‘conf/hdfs-site.xml’ on all the machines to include host and port details of the master node.
6. Formatted Hadoop’s distributed filesystem (HDFS) via the NameNode.
7. Started the multi-node cluster using the below two steps

* NameNode daemon is started on master and DataNode daemon are started on all the slave nodes
* ResourceManager is started on master and NodeManger daemons are started on all the slave nodes.

1. Then we run the HDFS daemons and MapReduce Daemons on the master node.

14. Used three ebooks from Project Gutenberg to run the wordcount example. The ebooks are available in the following links

<http://www.gutenberg.org/ebooks/20417>

<http://www.gutenberg.org/ebooks/5000>

<http://www.gutenberg.org/ebooks/4300>

<http://www.gutenberg.org/ebooks/132>

<http://www.gutenberg.org/ebooks/1661>

<http://www.gutenberg.org/ebooks/972>

http://www.gutenberg.org/ebooks/19699

15. Copied the local data to HDFS

16. Created an executable jar from the wordcount java project, and ran the mapreduce job.

17. Observed the concise information provided about the hadoop cluster summary using the below web interfaces.

**Task 2: Choosing the best learning algorithm for predicting the rating based on the yelp dataset considering accuracy as the primary goal.**

The Random Forest Classifier to classify the user input review

## Task 3: Implementation of the above chosen algorithm using map-reduce framework.