Welcome, this document is a record of my installation of Hadoop for study purpose.

- Hadoop version: 2.2.0
- OS: Ubantu 32 bit

```
hduser@ubuntu:~$ cat /etc/issue

Ubuntu 12.10 \n \l

Host: VMWARE workstation on Windows 8
```

Please be advice this is a draft, and just fit my own circumstance.

References:

http://blog.csdn.net/focusheart/article/details/14005893

http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster/

http://cs.smith.edu/dftwiki/index.php/Hadoop_Tutorial_1_--_Running_WordCount

http://hadoop.apache.org/docs/r2.2.0/hadoop-project-dist/hadoop-common/CommandsManual.html

Developer

http://www.osedu.net/article/nosql/2012-05-02/435.html

http://www.slideshare.net/waue/hadoop-map-reduce-3019713

!!To-Do list!!

Following things need to be verified:

```
1. Job tracker interface can't show

2. Where is job tracker?
```

```
It seems they changed job tracker to YARN (Next generation Map-reduce)

Use local:8088 instead of localhost 50030

http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-hadoop-2.2.0-features.html

https://hadoop.apache.org/docs/current2/hadoop-project-dist/hadoop-common/ClusterSetup.html
```

Web Interfaces

Once the Hadoop cluster is up and running check the web-ui of the components as described below:

NameNode	http://*nn_host:port/*
ResourceManager	http://*rm_host:port/*
MapReduce JobHistory Server	http://*jhs_host:port/*

Apache Hadoop NextGen MapReduce (YARN)

MapReduce has undergone a complete overhaul in hadoop-0.23 and we now have, what we call, MapReduce 2.0 (MRv2) or YARN.

So when check with JPS, there is no longer has jobtracker

http://blog.cloudera.com/blog/2013/11/migrating-to-mapreduce-2-on-yarn-for-users/

Web UI

In MR1, the JobTracker web UI served detailed information about the state of the cluster and the jobs currently and recently running on it. It also contained the job history page, which served information from disk about older jobs.

The MR2 web UI provides the same information structured in the same way, but has been revamped with a new look and feel. The ResourceManager UI, which includes information about running applications and the state of the cluster, is now located by default at:8088. The job history UI is now located by default at:19888. Jobs can be searched and viewed there just as they could in MR1.

Because the ResourceManager is meant to be agnostic to many of the concepts in MapReduce, it cannot host job information directly. Instead, it proxies to a web UI that can. If the job is running, this is the relevant MapReduce Application Master; if it has completed, this is the JobHistoryServer. In this sense, the user experience is similar to that of MR1, but the information is coming from different places.

Some conceptions

What Is Apache Hadoop?

The Apache™ Hadoop® project develops open-source software for reliable, scalable, distributed computing.

The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

The project includes these modules:

- **Hadoop Common**: The common utilities that support the other Hadoop modules.
- **Hadoop Distributed File System (HDFS™)**: A distributed file system that provides high-throughput access to application data.
- **Hadoop YARN**: A framework for job scheduling and cluster resource management.
- **Hadoop MapReduce**: A YARN-based system for parallel processing of large data sets.

Other Hadoop-related projects at Apache include:

- **Ambari™**: A web-based tool for provisioning, managing, and monitoring Apache Hadoop clusters which includes support for Hadoop HDFS, Hadoop MapReduce, Hive, HCatalog, HBase, ZooKeeper, Oozie, Pig and Sqoop. Ambari also provides a dashboard for viewing cluster health such as heatmaps and ability to view MapReduce, Pig and Hive applications visually alongwith features to diagnose their performance characteristics in a user-friendly manner.
- **Avro™**: A data serialization system.
- **Cassandra™**: A scalable multi-master database with no single points of failure.
- **Chukwa™**: A data collection system for managing large distributed systems.
- **HBase™**: A scalable, distributed database that supports structured data storage for large tables.
- **Hive™**: A data warehouse infrastructure that provides data summarization and ad hoc querying.
- **Mahout™**: A Scalable machine learning and data mining library.
- **Pig™**: A high-level data-flow language and execution framework for parallel computation.
- **Spark™**: A fast and general compute engine for Hadoop data. Spark provides a simple and expressive programming model that supports a wide range of applications, including ETL, machine learning, stream processing, and graph computation.
- **ZooKeeper™**: A high-performance coordination service for distributed applications.

Hadoop follows the idea of job map and reduce.

It has a DFS file system to support distribution

Important ports

~~1.Job Tracker：50030 (no longer in 2.20?)~~

2.HDFS ： 50070

3.HDFS communication：9000

4.MapReducecommunication：9001

Management

1. HDFS web interface

    http://hostname:50070

~~2. MapReduce interface~~

~~http://hostname:50030~~

Install Ubuntu

You may like to install it by a flash disk (on a new tower)

Install required packages

For example: JRE/JDK

```
$ sudo apt-get update
$ sudo apt-get install sun-java6-jdk
```

Add user and user group

```
$ sudo addgroup hadoop
$ sudo adduser --ingroup hadoop hduser
```

This will add the user `hduser` and the group `hadoop` to your local machine.

Configuring SSH

```
user@ubuntu:~$ su - hduser
hduser@ubuntu:~$ 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:

9b:82:ea:58:b4:e0:35:d7:ff:19:66:a6:ef:ae:0e:d2 hduser@ubuntu

The key's randomart image is:

[...snipp...]

hduser@ubuntu:~$
```

If success, you may ssh to your host without input credentials

Disable IPV6 (not proved whether necessary)

```
# disable ipv6

net.ipv6.conf.all.disable_ipv6 = 1
net.ipv6.conf.default.disable_ipv6 = 1
net.ipv6.conf.lo.disable_ipv6 = 1
```

You have to reboot your machine in order to make the changes take effect.
You can check whether IPv6 is enabled on your machine with the following command:

```
$ cat /proc/sys/net/ipv6/conf/all/disable_ipv6
```

Install Hadoop files

```
$ cd /usr/local

$ sudo tar xzf hadoop-1.0.3.tar.gz

$ sudo mv hadoop-1.0.3 hadoop
$ sudo chown -R hduser:hadoop hadoop
```

You may also think to make you directory where to place DFS data file, in my case it is /hadoopfs.

If necessary, grant owner to hduser:hadoop

Update environment variables

```
vim /home/hduser/.rcbash


export HADOOP_HOME=/usr/local/hadoop

unalias fs &> /dev/null

alias fs="hadoop fs"

unalias hls &> /dev/null

alias hls="fs -ls"


lzohead () {

    hadoop fs -cat $1 | lzop -dc | head -1000 | less

}


export PATH=$PATH:$HADOOP_HOME/bin

export JAVA_HOME=/usr/lib/jvm/default-java
```

Configure site

Modify core-site.xml

```
hduser@ubuntu:/usr/local/hadoop/etc/hadoop$ pwd

/usr/local/hadoop/etc/hadoop
```

Add following lines:

```
?xml version="1.0" encoding="UTF-8"?>

<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>

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

<configuration>

<property>
```

```
  <name>hadoop.tmp.dir</name>
  <value>/hadoopfs/tmp</value>
  <description>A base for other temporary directories.</description>
</property>

<property>
  <name>fs.default.name</name>
  <value>hdfs://localhost:9000</value>
  <description>The name of the default file system.  A URI whose
  scheme and authority determine the FileSystem implementation.  The
  uri's scheme determines the config property (fs.SCHEME.impl) naming
  the FileSystem implementation class.  The uri's authority is used to
  determine the host, port, etc. for a filesystem.</description>
<final>true</final>
</property>
</configuration>
```

Configure DHS

Modify hdsf-site.xml. This file indicates information to DFS

```
duser@ubuntu:/usr/local/hadoop/etc/hadoop$ cat hdfs-site.xml
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0
```

```
  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

<configuration>
<property>
  <name>dfs.replication</name>
  <value>1</value>
  <description>Default block replication.
  The actual number of replications can be specified when the file is created.
  The default is used if replication is not specified in create time.
  </description>
</property>

<property>
         <name>dfs.permissions</name>
         <value>false</value>
</property>

<property>
         <name>dfs.namenode.name.dir</name>
         <value>file:/hadoopfs/dfs/name</value>
         <final>true</final>
</property>
```

```
<property>
        <name>dfs.datanode.data.dir</name>
        <value>file:/hadoopfs/dfs/data</value>
        <final>true</final>
</property>


</configuration>
```

Configure Map-reduce

```
hduser@ubuntu:/usr/local/hadoop/etc/hadoop$ cat mapred-site.xml
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<!--
  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License. See accompanying LICENSE file.
-->

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

```
<configuration>


<property>
         <name>mapreduce.framework.name</name>
         <value>yarn</value>
</property>


<property>
         <name>mapreduce.jobtracker.system.dir</name>
         <value>file:/hadoopfs/dfs/system</value>
         <final>true</final>
</property>



<property>
  <name>mapreduce.jobtracker.address</name>
  <value>localhost:9001</value>
  <description>The host and port that the MapReduce job tracker runs
  at.  If "local", then jobs are run in-process as a single map
  and reduce task.
  </description>
</property>


<property>
        <name>mapreduce.jobtracker.http.address</name>
        <value>localhost:50030</value>
</property>



<property>
```

```
        <name>mapreduce.cluster.local.dir</name>
        <value>file:/hadoopfs/dfs/local</value>
        <final>true</final>
</property>


  <property>
    <name>mapreduce.cluster.temp.dir</name>
    <value>file:/hadoopfs/dfs/tmp</value>
    <description>No description</description>
    <final>true</final>
  </property>
</configuration>
```

Yarn-site.xml

```
<?xml version="1.0"?>
<configuration>


<!-- Site specific YARN configuration properties -->
  <property>
    <name>yarn.resourcemanager.hostname</name>
    <value>localhost</value>
    <description>the host is the hostname of the ResourceManager and the port is the
port on
    which the clients can talk to the Resource Manager. </description>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services.mapreduce_shuffle.class</name>
    <value>org.apache.hadoop.mapred.ShuffleHandler</value>
```

```
    <description>shuffle service that needs to be set for Map Reduce to run
</description>
  </property>
  <property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce.shuffle</value>
  </property>
</configuration>
```

Initialize DFS

```
$ hdfs namenode -format
```

In some cases if there is some problem with node and namenode DFS, you need to re-format the DFS

```
$ rm -rf /hadoopfs/dfs/*
$ rm -rf /hadoopfs/tmp/*
```

Then re-format namenode DFS.

Start and stop the server(s)

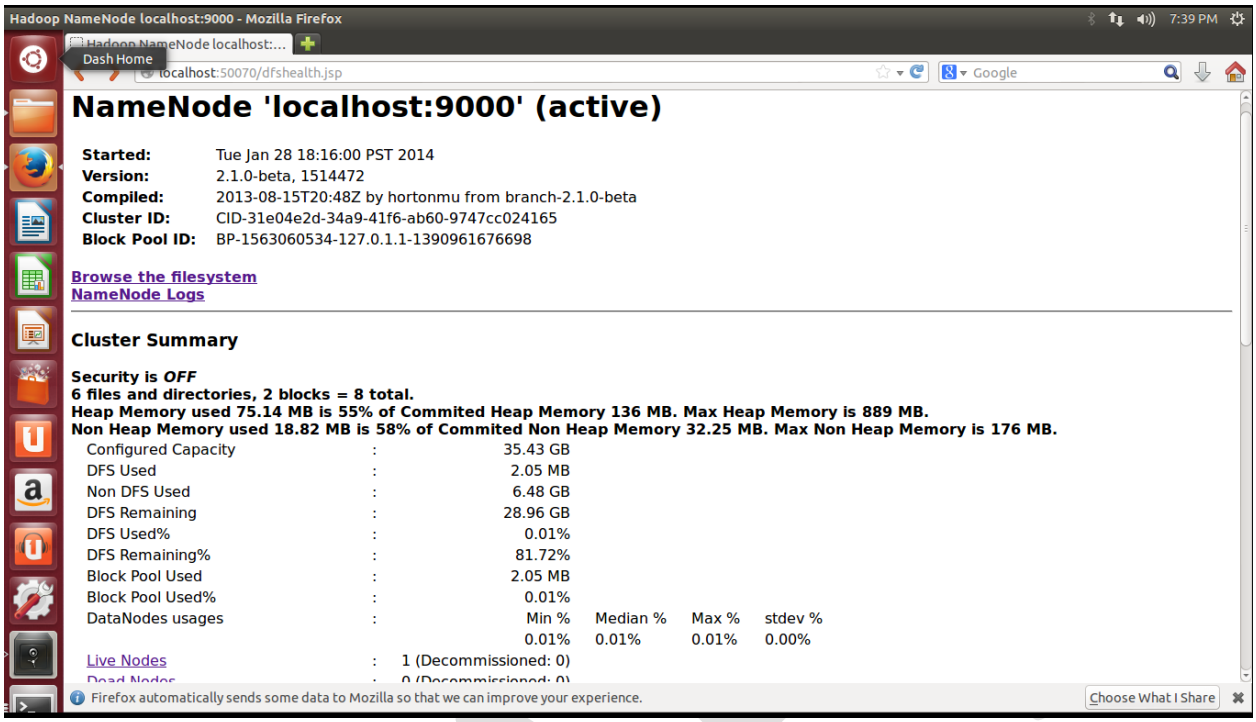
Should be done one by one – namenode, datanode and Yarn(resource manager), but you can use start-all.sh and stop-all.sh to enable one time start/stop

/usr/local/hadoop/sbin

```
$ hadoop-daemon.sh start namenode
$ hadoop-daemon.sh start datanode
```

Monitor the node(s)

General nodes information can be checked by web browser at port 50070 (by default)



You can also use jps to see following instances are running

```
hduser@ubuntu:/usr/local/hadoop/sbin$ jps

3135 ResourceManager

2678 DataNode

2428 NameNode

5044 Jps

2961 SecondaryNameNode
```

Run a test

Let's try this word count example. Don't try to run the randomwriter example by default if you want waste 10 GB

http://wiki.apache.org/hadoop/RandomWriter
RandomWriter example writes 10 gig (by default) of random data/host to DFS using Map/Reduce.

Try to find some txt such like this cool stuff

```
$ wget http://www.gutenberg.org/files/4300/4300.zip
$ unzip 4300.zip
```

```
hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop dfs -mkdir /tmp

hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop dfs -copyFromLocal
4300.txt /tmp
```

Now ready to roll!

```
hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop jar hadoop-mapreduce-
examples-2.1.0-beta.jar wordcount /tmp/4300.txt /tmp/output3

14/01/28 20:04:56 WARN conf.Configuration: session.id is deprecated. Instead, use
dfs.metrics.session-id

14/01/28 20:04:56 INFO jvm.JvmMetrics: Initializing JVM Metrics with
processName=JobTracker, sessionId=

14/01/28 20:04:56 INFO input.FileInputFormat: Total input paths to process : 1

14/01/28 20:04:56 INFO mapreduce.JobSubmitter: number of splits:1

14/01/28 20:04:56 WARN conf.Configuration: user.name is deprecated. Instead, use
mapreduce.job.user.name

14/01/28 20:04:56 WARN conf.Configuration: mapred.jar is deprecated. Instead, use
mapreduce.job.jar

14/01/28 20:04:56 WARN conf.Configuration: mapred.output.value.class is deprecated.
Instead, use mapreduce.job.output.value.class

14/01/28 20:04:56 WARN conf.Configuration: mapreduce.combine.class is deprecated.
Instead, use mapreduce.job.combine.class

14/01/28 20:04:56 WARN conf.Configuration: mapreduce.map.class is deprecated. Instead,
use mapreduce.job.map.class

14/01/28 20:04:56 WARN conf.Configuration: mapred.job.name is deprecated. Instead, use
mapreduce.job.name

14/01/28 20:04:56 WARN conf.Configuration: mapreduce.reduce.class is deprecated.
Instead, use mapreduce.job.reduce.class

14/01/28 20:04:56 WARN conf.Configuration: mapred.input.dir is deprecated. Instead,
use mapreduce.input.fileinputformat.inputdir

14/01/28 20:04:56 WARN conf.Configuration: mapred.output.dir is deprecated. Instead,
use mapreduce.output.fileoutputformat.outputdir

14/01/28 20:04:56 WARN conf.Configuration: mapred.map.tasks is deprecated. Instead,
use mapreduce.job.maps

14/01/28 20:04:56 WARN conf.Configuration: mapred.output.key.class is deprecated.
Instead, use mapreduce.job.output.key.class
```

```
14/01/28 20:04:56 WARN conf.Configuration: mapred.working.dir is deprecated. Instead,
use mapreduce.job.working.dir

14/01/28 20:04:56 INFO mapreduce.JobSubmitter: Submitting tokens for job:
job_local513207422_0001

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: mapreduce.local.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: dfs.namenode.name.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: mapreduce.job.end-
notification.max.retry.interval;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: dfs.datanode.data.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: mapreduce.job.end-
notification.max.attempts;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: fs.defaultFS;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/staging/hduser513207422/.staging/job_local513207422_0001/job
.xml:an attempt to override final parameter: mapreduce.system.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: mapreduce.local.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: dfs.namenode.name.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: mapreduce.job.end-notification.max.retry.interval;
Ignoring.
```

```
14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: dfs.datanode.data.dir;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: mapreduce.job.end-notification.max.attempts;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: fs.defaultFS;  Ignoring.

14/01/28 20:04:56 WARN conf.Configuration:
file:/hadoopfs/tmp/mapred/local/localRunner/job_local513207422_0001.xml:an attempt to
override final parameter: mapreduce.system.dir;  Ignoring.

14/01/28 20:04:56 INFO mapreduce.Job: The url to track the job: http://localhost:8080/

14/01/28 20:04:56 INFO mapreduce.Job: Running job: job_local513207422_0001

14/01/28 20:04:56 INFO mapred.LocalJobRunner: OutputCommitter set in config null

14/01/28 20:04:56 INFO mapred.LocalJobRunner: OutputCommitter is
org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter

14/01/28 20:04:56 INFO mapred.LocalJobRunner: Waiting for map tasks

14/01/28 20:04:56 INFO mapred.LocalJobRunner: Starting task:
attempt_local513207422_0001_m_000000_0

14/01/28 20:04:56 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]

14/01/28 20:04:56 INFO mapred.MapTask: Processing split:
hdfs://localhost:9000/tmp/4300.txt:0+1573078

14/01/28 20:04:57 INFO mapred.MapTask: Map output collector class =
org.apache.hadoop.mapred.MapTask$MapOutputBuffer

14/01/28 20:04:57 INFO mapred.MapTask: (EQUATOR) 0 kvi 26214396(104857584)

14/01/28 20:04:57 INFO mapred.MapTask: mapreduce.task.io.sort.mb: 100

14/01/28 20:04:57 INFO mapred.MapTask: soft limit at 83886080

14/01/28 20:04:57 INFO mapred.MapTask: bufstart = 0; bufvoid = 104857600

14/01/28 20:04:57 INFO mapred.MapTask: kvstart = 26214396; length = 6553600

14/01/28 20:04:57 INFO mapred.LocalJobRunner:

14/01/28 20:04:57 INFO mapred.MapTask: Starting flush of map output

14/01/28 20:04:57 INFO mapred.MapTask: Spilling map output
```

```
14/01/28 20:04:57 INFO mapred.MapTask: bufstart = 0; bufend = 2601826; bufvoid =
104857600

14/01/28 20:04:57 INFO mapred.MapTask: kvstart = 26214396(104857584); kvend =
25142480(100569920); length = 1071917/6553600

14/01/28 20:04:57 INFO mapreduce.Job: Job job_local513207422_0001 running in uber
mode : false

14/01/28 20:04:57 INFO mapreduce.Job:  map 0% reduce 0%

14/01/28 20:04:58 INFO mapred.MapTask: Finished spill 0

14/01/28 20:04:58 INFO mapred.Task: Task:attempt_local513207422_0001_m_000000_0 is
done. And is in the process of committing

14/01/28 20:04:58 INFO mapred.LocalJobRunner: map

14/01/28 20:04:58 INFO mapred.Task: Task 'attempt_local513207422_0001_m_000000_0'
done.

14/01/28 20:04:58 INFO mapred.LocalJobRunner: Finishing task:
attempt_local513207422_0001_m_000000_0

14/01/28 20:04:58 INFO mapred.LocalJobRunner: Map task executor complete.

14/01/28 20:04:58 INFO mapred.Task:  Using ResourceCalculatorProcessTree : [ ]

14/01/28 20:04:58 INFO mapred.Merger: Merging 1 sorted segments

14/01/28 20:04:58 INFO mapred.Merger: Down to the last merge-pass, with 1 segments
left of total size: 725062 bytes

14/01/28 20:04:58 INFO mapred.LocalJobRunner:

14/01/28 20:04:58 WARN conf.Configuration: mapred.skip.on is deprecated. Instead, use
mapreduce.job.skiprecords

14/01/28 20:04:58 INFO mapreduce.Job:  map 100% reduce 0%

14/01/28 20:04:58 INFO mapred.Task: Task:attempt_local513207422_0001_r_000000_0 is
done. And is in the process of committing

14/01/28 20:04:58 INFO mapred.LocalJobRunner:

14/01/28 20:04:58 INFO mapred.Task: Task attempt_local513207422_0001_r_000000_0 is
allowed to commit now

14/01/28 20:04:58 INFO output.FileOutputCommitter: Saved output of task
'attempt_local513207422_0001_r_000000_0' to
hdfs://localhost:9000/tmp/output3/_temporary/0/task_local513207422_0001_r_000000

14/01/28 20:04:58 INFO mapred.LocalJobRunner: reduce > reduce
```

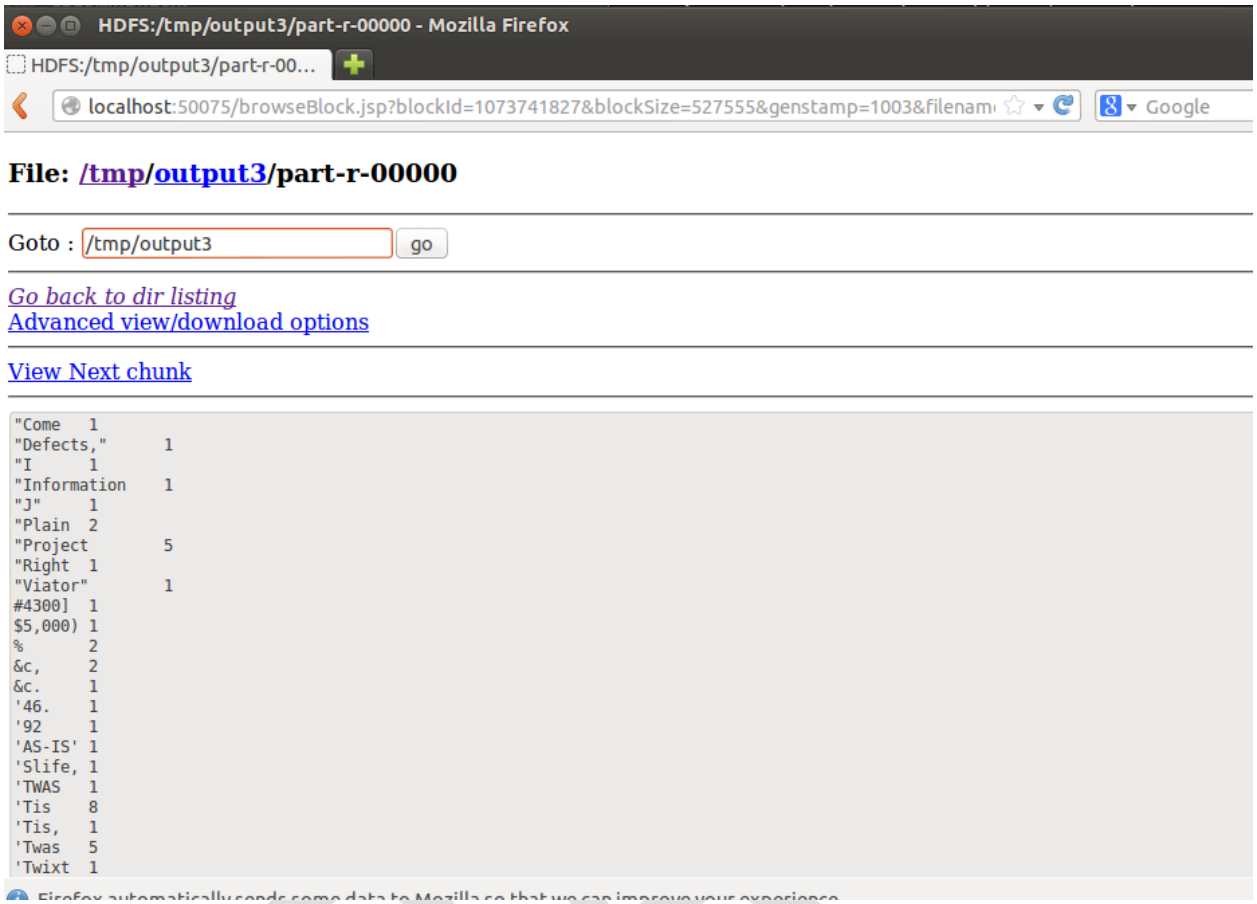
```
14/01/28 20:04:58 INFO mapred.Task: Task 'attempt_local513207422_0001_r_000000_0' done.

14/01/28 20:04:59 INFO mapreduce.Job:  map 100% reduce 100%

14/01/28 20:04:59 INFO mapreduce.Job: Job job_local513207422_0001 completed successfully

14/01/28 20:04:59 INFO mapreduce.Job: Counters: 32

        File System Counters

                FILE: Number of bytes read=1268626

                FILE: Number of bytes written=2363894

                FILE: Number of read operations=0

                FILE: Number of large read operations=0

                FILE: Number of write operations=0

                HDFS: Number of bytes read=3146156

                HDFS: Number of bytes written=527555

                HDFS: Number of read operations=13

                HDFS: Number of large read operations=0

                HDFS: Number of write operations=4

        Map-Reduce Framework

                Map input records=33056

                Map output records=267980

                Map output bytes=2601826

                Map output materialized bytes=725074

                Input split bytes=99

                Combine input records=267980

                Combine output records=50095

                Reduce input groups=50095

                Reduce shuffle bytes=0

                Reduce input records=50095

                Reduce output records=50095

                Spilled Records=100190
```

```
                Shuffled Maps =0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=37
                CPU time spent (ms)=0
                Physical memory (bytes) snapshot=0
                Virtual memory (bytes) snapshot=0
                Total committed heap usage (bytes)=508559360
        File Input Format Counters
                Bytes Read=1573078
        File Output Format Counters
                Bytes Written=527555
hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$
```

see the result by

```
hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop dfs -cat
/tmp/output3/part*
```

Or from web console



Remove a directory

```
hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop dfs -rmr /tmp/output

DEPRECATED: Use of this script to execute hdfs command is deprecated.

Instead use the hdfs command for it.


rmr: DEPRECATED: Please use 'rm -r' instead.

14/01/30 09:26:27 INFO fs.TrashPolicyDefault: Namenode trash configuration: Deletion
interval = 0 minutes, Emptier interval = 0 minutes.

Deleted /tmp/output

hduser@ubuntu:/usr/local/hadoop/share/hadoop/mapreduce$ hadoop dfs -rmr /tmp/output2

DEPRECATED: Use of this script to execute hdfs command is deprecated.

Instead use the hdfs command for it.
```

```
rmr: DEPRECATED: Please use 'rm -r' instead.

14/01/30 09:26:56 INFO fs.TrashPolicyDefault: Namenode trash configuration: Deletion
interval = 0 minutes, Emptier interval = 0 minutes.

Deleted /tmp/output2
```