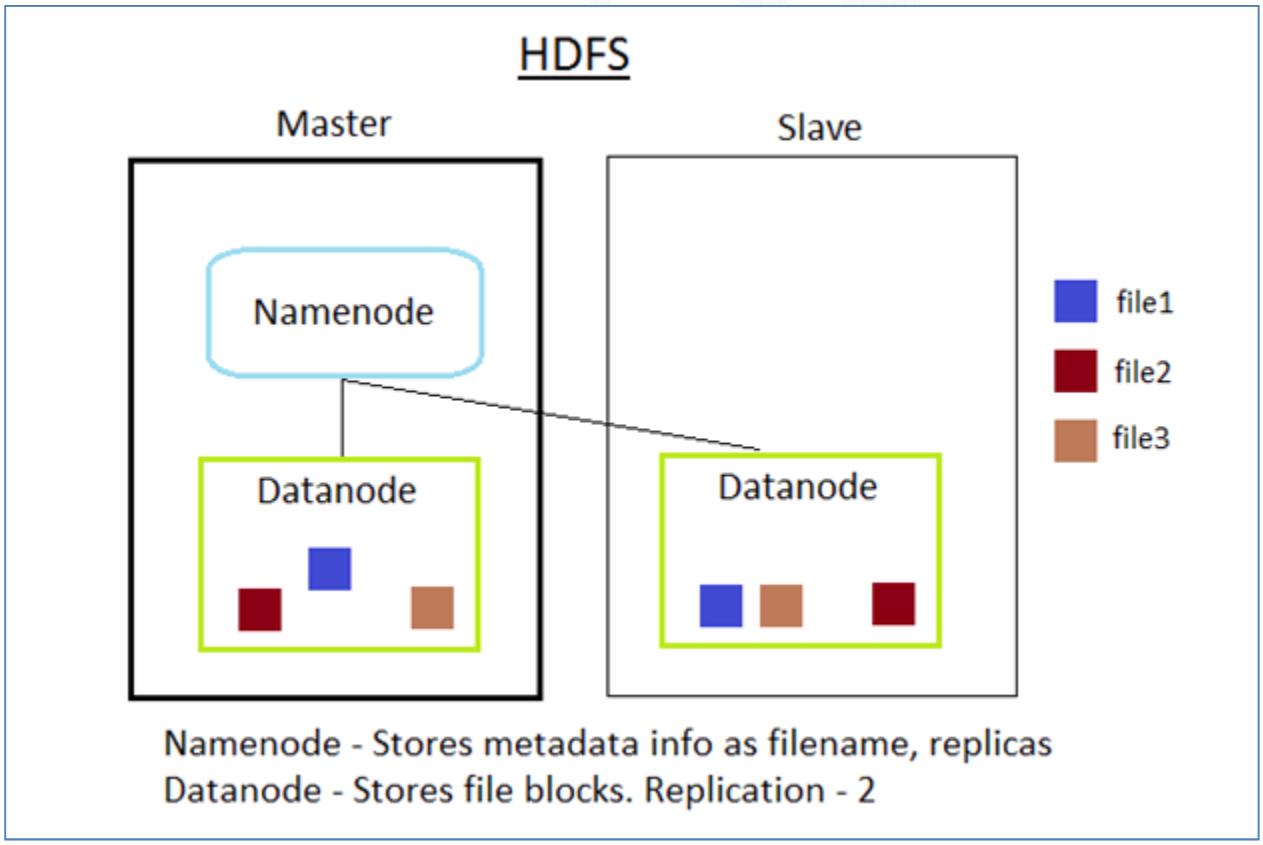
Hadoop Distributed File System (HDFS) Overview

Here is a quick overview of HDFS before we get started with hands on HDFS Lab.

HDFS

- It is a primary distributed file storage system and it forms the base of Hadoop Architecture.
- It consists of
  - *NameNode* which manages the file system and
  - *DataNode* which stores the actual data.
- *Namenode* holds the metadata, takes care of replication.
- The client contacts *NameNode* for file metadata and perform actual file I/O directly with *DataNodes*.
- It is fault tolerant, scalable and extremely simple to expand.
- Hadoop supports shell like commands to interact with HDFS directly.



For more details about HDFS design refer to the HDFS User Guide at
http://hadoop.apache.org/docs/r0.17.1/hdfs_design.html

HDFS Lab

This lab covers the frequently used DFS Shell commands. In this lab you will learn how to Create, Read, Edit/Update, Copy files and directories in HDFS.

HDFS commands are executed from $HADOOP_HOME/bin directory. With $HADOOP_HOME directory already in path, the following commands can be executed from any directory.

1. Get the list of HDFS commands or help for HDFS commands

```
$ hadoop dfs
Usage: java FsShell
        [-ls <path>]
        [-lsr <path>]
        [-du <path>]
        [-dus <path>]
        [-count[-q] <path>]
        [-mv <src> <dst>]
        [-cp <src> <dst>]
        [-rm [-skipTrash] <path>]
        [-rmr [-skipTrash] <path>]
        [-expunge]
        [-put <localsrc> ... <dst>]
        [-copyFromLocal <localsrc> ... <dst>]
        [-moveFromLocal <localsrc> ... <dst>]
        [-get [-ignoreCrc] [-crc] <src> <localdst>]
        [-getmerge <src> <localdst> [addnl]]
        [-cat <src>]
        [-text <src>]
        [-copyToLocal [-ignoreCrc] [-crc] <src> <localdst>]
        [-moveToLocal [-crc] <src> <localdst>]
        [-mkdir <path>]
        [-setrep [-R] [-w] <rep> <path/file>]
        [-touchz <path>]
        [-test -[ezd] <path>]
        [-stat [format] <path>]
        [-tail [-f] <file>]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-chgrp [-R] GROUP PATH...]
        [-help [cmd]]

Generic options supported are
-conf <configuration file>     specify an application configuration file
-D <property=value>            use value for given property
-fs <local|namenode:port>      specify a namenode
-jt <local|jobtracker:port>    specify a job tracker
-files <comma separated list of files>    specify comma separated files to be
copied to the map reduce cluster
```

```
-libjars <comma separated list of jars>    specify comma separated jar files to
include in the classpath.
-archives <comma separated list of archives>    specify comma separated
archives to be unarchived on the compute machines.

The general command line syntax is
bin/hadoop command [genericOptions] [commandOptions]
```

2. Create a local dir

```
$ mkdir /home/hadoop/input
```

For this tutorial we will use this dir as local input directory.

3. Create a local input file

```
$ vim /home/hadoop/input/input.txt
```

Add some content
This is a test input file
Welcome to Big Data Hands on Lab
Save and close the file (press :q!)

4. Create an hdfs directory

```
$ hadoop fs -mkdir /home/hadoop/dfsinput
```

For this lab we will use the /home/hadoop/dfsinput as fs input location.

5. Check if the directory is created

```
$ ls /home/hadoop/dfsinput
ls: /home/hadoop/dfsinput: No such file or directory
```

dfsinput being a hdfs file system directory, it will not be visible using normal file commands. You will have to use fs commands to check the directory as files are distributed.

6. Check hdfs dir using hdfs command

```
$ hadoop fs -ls /home/hadoop
Found 1 item
drwxr-xr-x   - hduser supergroup          0 2012-10-21 10:34
/home/hadoop/dfsinput
```

7. Copy the local input.txt file to hdfs directory

```
$ hadoop fs -copyFromLocal /home/hadoop/input/input.txt
/home/hadoop/dfsinput
```

For this lab we will use the /home/hadoop/dfsinput as hdfs input location.

8. Check if the file is created

```
$ hadoop fs -ls /home/hadoop/dfsinput
Found 1 items
-rw-r--r--   2 hduser supergroup         59 2012-10-21 10:34
/home/hadoop/dfsinput/input.txt
```

9. View the content of the file using fs command

```
$ hadoop fs -cat /home/hadoop/dfsinput/input.txt
    This is a test input file
    Welcome to Big Data Hands on Lab
```

10. Create an output directory

```
$ mkdir /home/hadoop/output
```

11. Copy the hdfs file to local file system

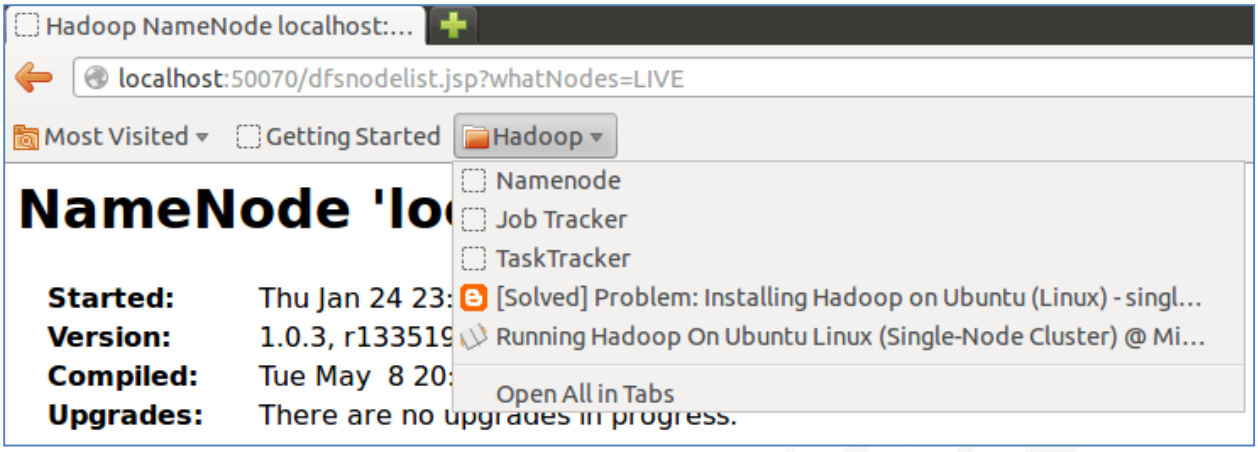
```
$ hadoop fs -copyToLocal /home/hadoop/dfsinput/input.txt /home/hadoop/output
```

12. Check the output dir

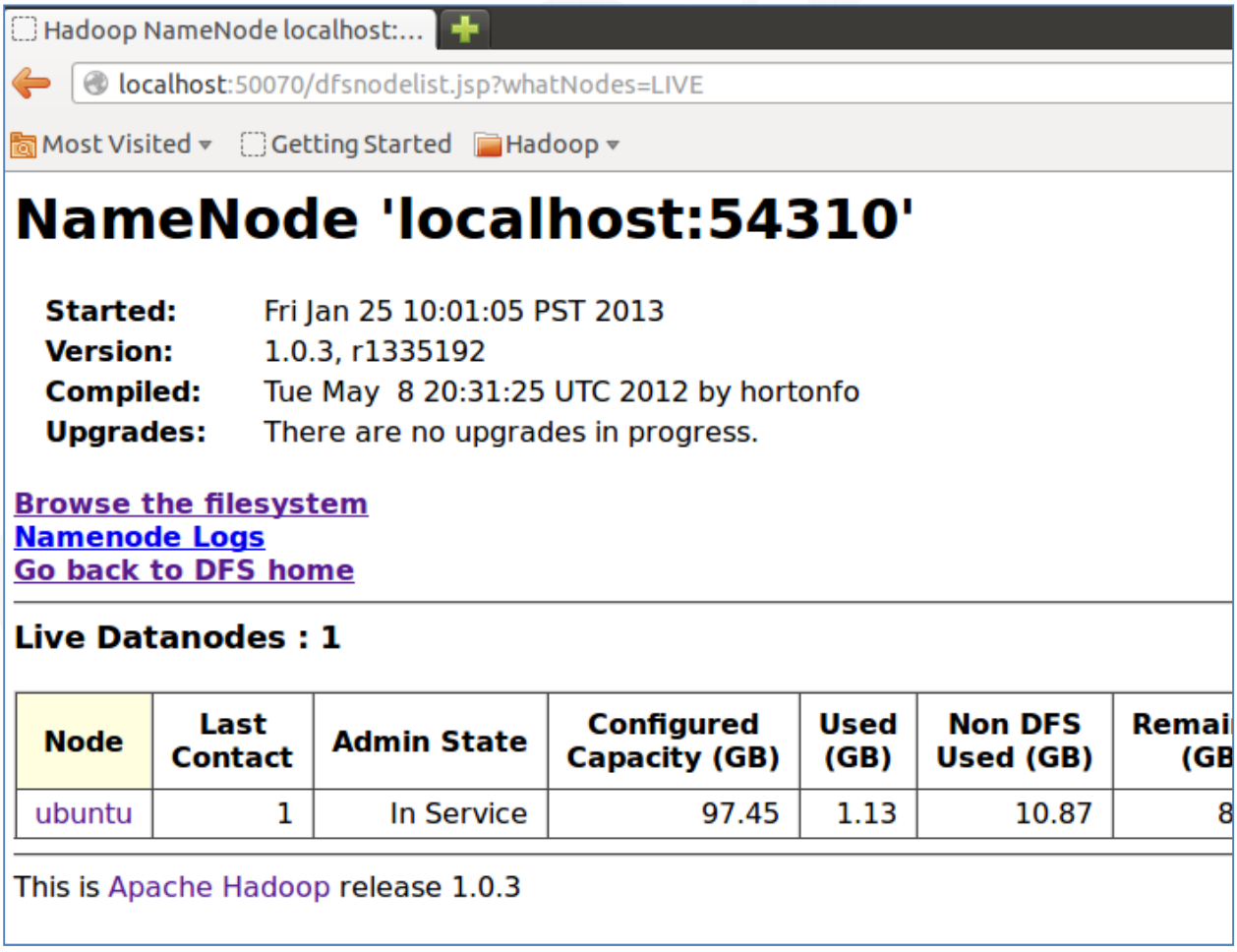
```
$ ls /home/hadoop/output/
input.txt
```

13. **Browser interface** to access the hdfs file system

<TBD NOT WORKING> http://localhost:50070/dfshealth.jsp
Browser links are bookmarked in the VM firefox browser.



This is a namenode home page. Here you can check the status of all the nodes in the hadoop system.
You can also access the hdfs file system.
Click on the 'Browse the filesystem' -> home -> hadoop -> dfsinput -> input.txt



14. Remove the hdfs file

```
$ hadoop fs -rm /home/hadoop/dfsinput/input.txt
Deleted hdfs://adc2120094:54310/home/hadoop/dfsinput/input.txt
```

15. Check if file is deleted

```
$ hadoop fs -ls /home/hadoop/dfsinput/input.txt
ls: Cannot access /home/hadoop/dfsinput/input.txt: No such file or directory.
```

16. Removing the hdfs directory

```
$ hadoop fs -rmr /home/hadoop/dfsinput
Deleted hdfs://adc2120094:54310/home/hadoop/dfsinput
```

Gaja Data