

Upgrade existing Hadoop Cluster from Hadoop 1 to Hadoop 2

A guide to Upgrade existing Hadoop Cluster from Hadoop 1 to Hadoop 2

edureka!

edureka!

Software Requirements

- ✓ VMware Player or Oracle Virtual Box
- ✓ CentOS Virtual Machine

Hardware Requirements

- ✓ Intel Core i3 processor or higher
- ✓ **8** GB RAM Recommended
- ✓ **300** GB for VM Recommended (By default 40 GB is taken)

edureka!

Introduction

This setup and configuration document is a guide upgrade the Hadoop-1 cluster to hadoop-2 on a CentOS virtual machine on your PC.

The guide describes the whole process in three parts:

[Section 1: Taking back up of the Hadoop-1.x cluster](#)

In this section we describe how to take the back of the cluster, and which information need to take the back up from the hadoop-1.x cluster.

[Section 2: Setting up the hadoop-2.6.0](#)

In this section we describe setting up hadoop-2.6.0 environment variables and changing configuration files to hadoop cluster.

[Section 3: Upgrading hadoop cluster](#)

In this section we describe what are steps need to take care to upgrade the hadoop cluster and finalizing hadoop upgrade.

Note: The configuration described here is intended for learning purposes only.

Introduction:

An upgrade of HDFS makes a copy of the previous version's metadata and data. Doing an upgrade does not double the storage requirements of the cluster, as the data nodes use hard links to keep two references (for the current and previous version) to the same block of data.

This design makes it straightforward to roll back to the previous version of the filesystem, should you need to. You should understand that any changes made to the data on the upgraded system will be lost after the rollback completes.

You can keep only the previous version of the filesystem: you can't roll back several versions. Therefore, to carry out another upgrade to HDFS data and metadata, you will need to delete the previous version, a process called finalizing the upgrade.

Once an upgrade is finalized, there is no procedure for rolling back to a previous version.

edureka!

Section 1: Taking back up of the Hadoop-1.x cluster

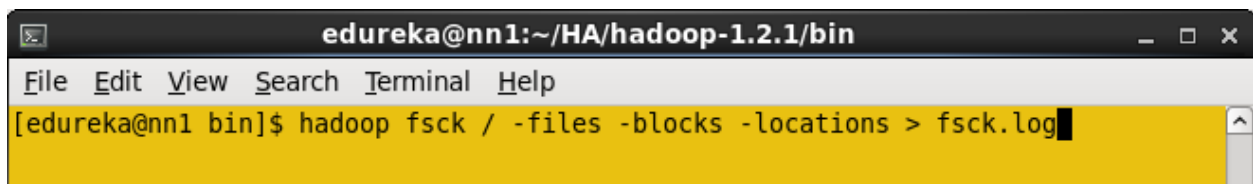
1.1: Before upgrading the Hadoop cluster stop all the Map-reduce application running on the cluster and take the back of the HDFS namespace, clusters data node report, and health of file system details.

Make sure that any previous upgrade is finalized before proceeding with another upgrade.

Command: `hadoop dfsadmin -upgradeProgress status`

1.2: Take the backup of the HDFS Files list, Block report and location of blocks by runs a HDFS file system checking utility.

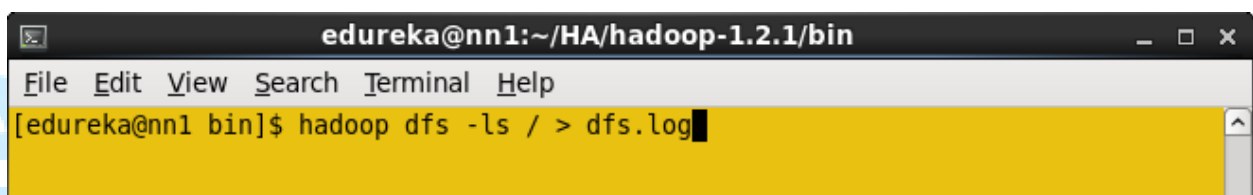
Command: `hadoop / -files -blocks -locations > fsck.log`



```
edureka@nn1:~/HA/hadoop-1.2.1/bin
File Edit View Search Terminal Help
[edureka@nn1 bin]$ hadoop fsck / -files -blocks -locations > fsck.log
```

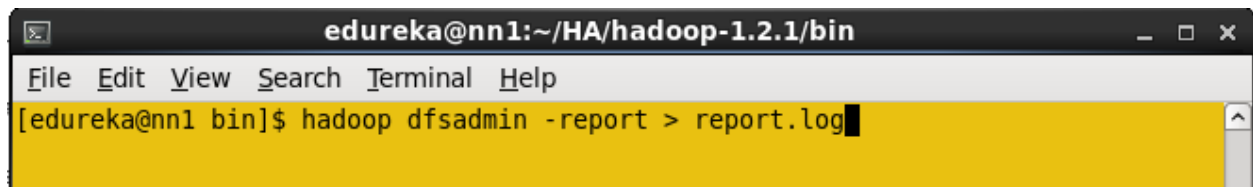
1.3: Take the backup of the file system name space.

Command: `hadoop dfs -ls / > dfs.log`



```
edureka@nn1:~/HA/hadoop-1.2.1/bin
File Edit View Search Terminal Help
[edureka@nn1 bin]$ hadoop dfs -ls / > dfs.log
```

1.4: Save the data nodes report, to check the data nodes details after upgrade.



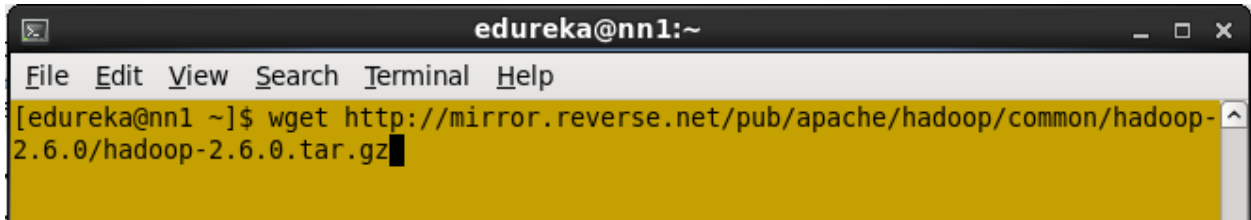
```
edureka@nn1:~/HA/hadoop-1.2.1/bin
File Edit View Search Terminal Help
[edureka@nn1 bin]$ hadoop dfsadmin -report > report.log
```

1.5: Stop all the Daemons running in hadoop cluster using `stop-all.sh` or `stop-dfs.sh` and `stop-mapred.sh` scripts.

Section 2: Setting up the hadoop-2.6.0

2.1: Download the New version of Hadoop to upgrade the cluster.

Command: wget



```
edureka@nn1:~  
File Edit View Search Terminal Help  
[edureka@nn1 ~]$ wget http://mirror.reverse.net/pub/apache/hadoop/common/hadoop-2.6.0/hadoop-2.6.0.tar.gz
```

2.2: Extract the hadoop tar ball.

Command: hadoop tar -xvf hadoop-2.6.0.tar.gz

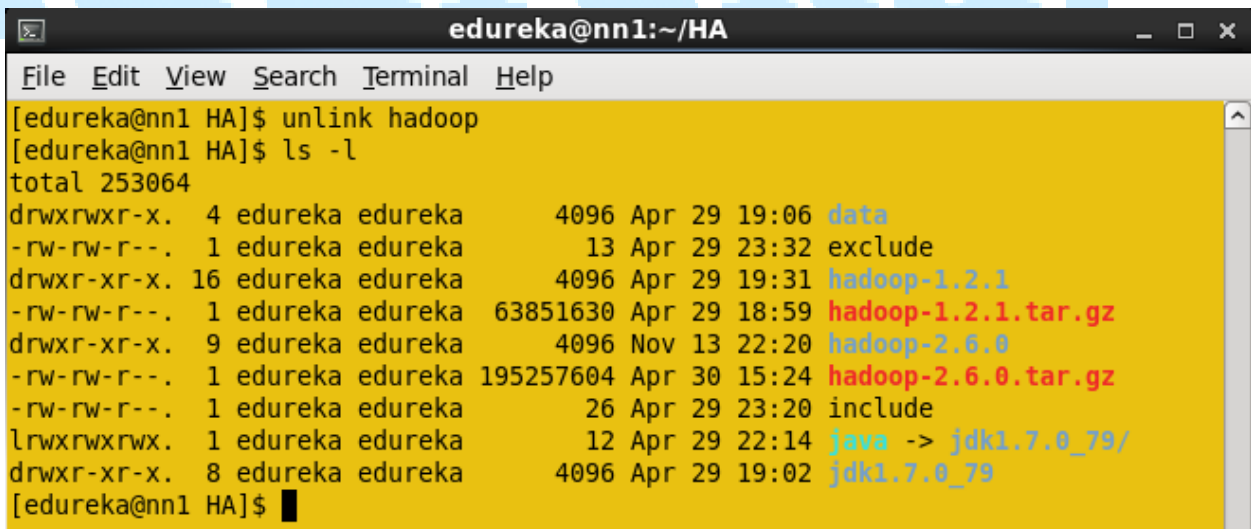


```
edureka@nn1:~  
File Edit View Search Terminal Help  
[edureka@nn1 ~]$ tar -xvf hadoop-2.6.0.tar.gz
```

2.3: If you set up any soft link to hadoop-1.2.1 directory delete the soft link and change the soft link to hadoop-2.6.0 directory.

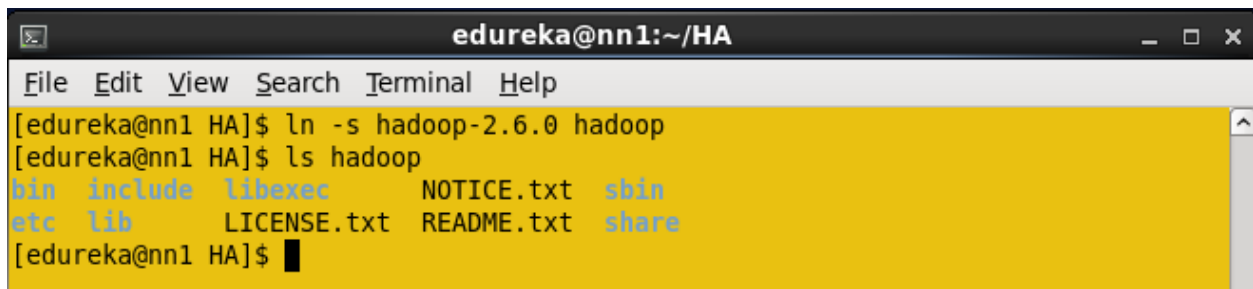
I have given hadoop soft link to hadoop-1.2.1 directory

Command: unlink hadoop



```
edureka@nn1:~/HA  
File Edit View Search Terminal Help  
[edureka@nn1 HA]$ unlink hadoop  
[edureka@nn1 HA]$ ls -l  
total 253064  
drwxrwxr-x.  4 edureka edureka    4096 Apr 29 19:06 data  
-rw-rw-r--.  1 edureka edureka     13 Apr 29 23:32 exclude  
drwxr-xr-x. 16 edureka edureka    4096 Apr 29 19:31 hadoop-1.2.1  
-rw-rw-r--.  1 edureka edureka 63851630 Apr 29 18:59 hadoop-1.2.1.tar.gz  
drwxr-xr-x.  9 edureka edureka    4096 Nov 13 22:20 hadoop-2.6.0  
-rw-rw-r--.  1 edureka edureka 195257604 Apr 30 15:24 hadoop-2.6.0.tar.gz  
-rw-rw-r--.  1 edureka edureka     26 Apr 29 23:20 include  
lrwxrwxrwx.  1 edureka edureka     12 Apr 29 22:14 java -> jdk1.7.0_79/  
drwxr-xr-x.  8 edureka edureka    4096 Apr 29 19:02 jdk1.7.0_79  
[edureka@nn1 HA]$
```

2.4: Add the soft link to hadoop-2.6.0.



```
edureka@nn1:~/HA
File Edit View Search Terminal Help
[edureka@nn1 HA]$ ln -s hadoop-2.6.0 hadoop
[edureka@nn1 HA]$ ls hadoop
bin  include  libexec  NOTICE.txt  sbin
etc  lib      LICENSE.txt  README.txt  share
[edureka@nn1 HA]$
```

2.5: Update the bashrc file with new hadoop-2.6.0 environment variables.

The paths you need to add in bashrc is:

export HADOOP_HOME=< Path to your Hadoop-2.6.0 directory>

export HADOOP_MAPRED_HOME=\$HADOOP_HOME

export HADOOP_COMMON_HOME=\$HADOOP_HOME

export HADOOP_HDFS_HOME=\$HADOOP_HOME

export YARN_HOME=\$HADOOP_HOME

export HADOOP_CONF_DIR=\$HADOOP_HOME/etc/hadoop

export YARN_CONF_DIR=\$HADOOP_HOME/etc/hadoop

export JAVA_HOME=<Path to your Java Directory>

export PATH=\$PATH: \$JAVA_HOME/bin: \$HADOOP_HOME/bin: \$HADOOP_HOME/sbin

```
[edureka@nn1 HA]$ sudo gedit ~/.bashrc
[sudo] password for edureka:

*.bashrc (/home/edureka) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
*.bashrc
. /etc/bashrc
fi
# User specific aliases and functions
export JAVA_HOME=/home/edureka/HA/jdk1.7.0_79
export PATH=$PATH:$JAVA_HOME/bin

export HADOOP_HOME=/home/edureka/HA/hadoop

export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_CONF_DIR=$HADOOP_HOME/etc/hadoop
export YARN_CONF_DIR=$HADOOP_HOME/etc/hadoop

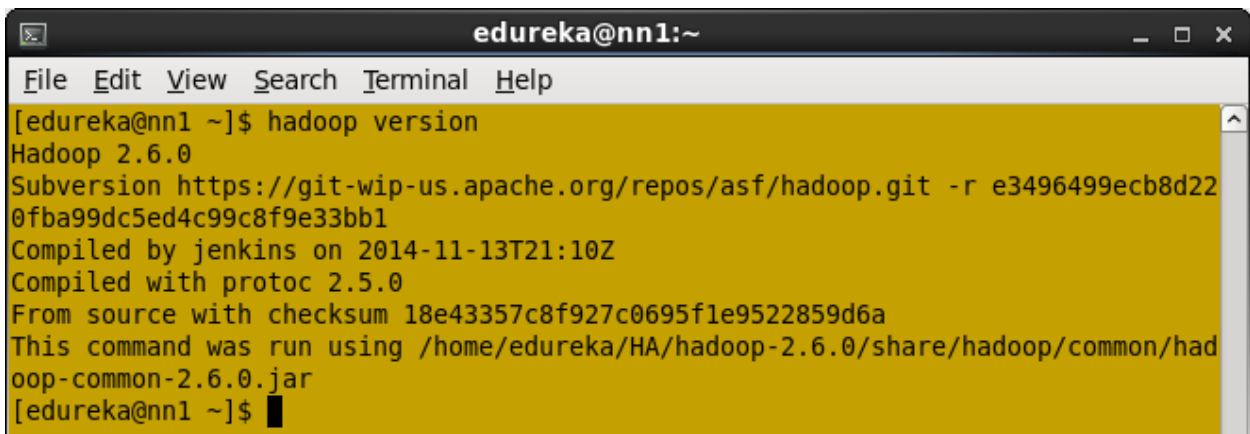
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
```

2.6: To apply all these changes to current running Terminal run the source command.

Command: source ~/.bashrc

```
edureka@nn1:~
File Edit View Search Terminal Help
[edureka@nn1 ~]$ source ~/.bashrc
[edureka@nn1 ~]$
```


Check the Hadoop version.



```
edureka@nn1:~  
File Edit View Search Terminal Help  
[edureka@nn1 ~]$ hadoop version  
Hadoop 2.6.0  
Subversion https://git-wip-us.apache.org/repos/asf/hadoop.git -r e3496499ecb8d22  
0fba99dc5ed4c99c8f9e33bb1  
Compiled by jenkins on 2014-11-13T21:10Z  
Compiled with protoc 2.5.0  
From source with checksum 18e43357c8f927c0695f1e9522859d6a  
This command was run using /home/edureka/HA/hadoop-2.6.0/share/hadoop/common/had  
oop-common-2.6.0.jar  
[edureka@nn1 ~]$
```

2.7: Now Change Hadoop-2.6.0 configuration files.

All the Hadoop configuration files are located in Hadoop-2.6.0/etc/hadoop directory.

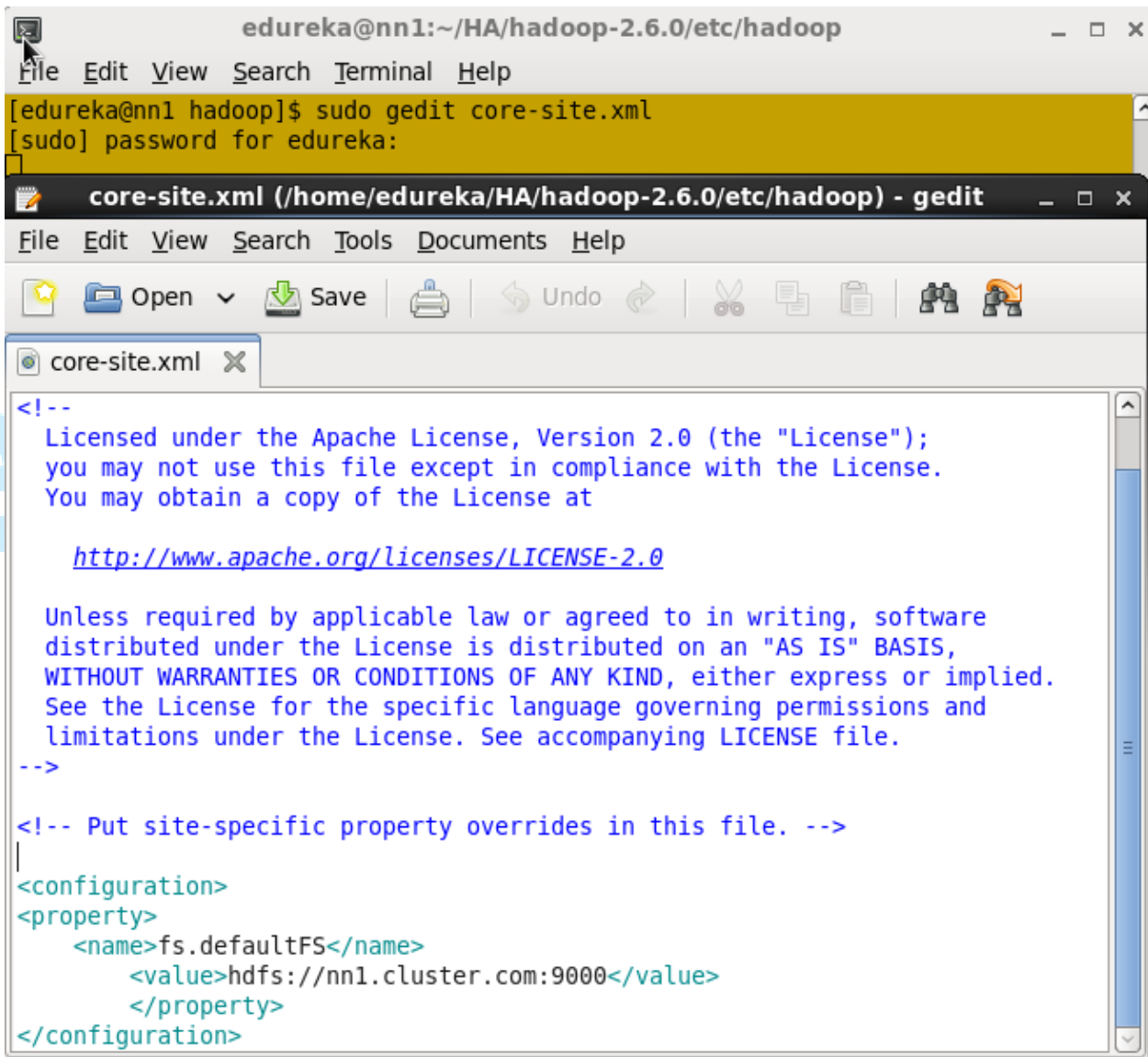
The configuration files that need to change is:

Configuration Filenames	Description
hadoop-env.sh	Environment variables that are used in the scripts to run Hadoop.
core-site.xml	Configuration settings for Hadoop Core such as I/O settings that Are common to HDFS and MapReduce.
hdfs-site.xml	Configuration settings for HDFS daemons, the name node, The secondary namenode and the data nodes.
mapred-site.xml	Configuration settings for MapReduce Applications.
yarn-site.xml	Configuration settings for ResourceManager and NodeManager.
Slaves	Contain the Each Datanode IP address to identify the slave nodes.

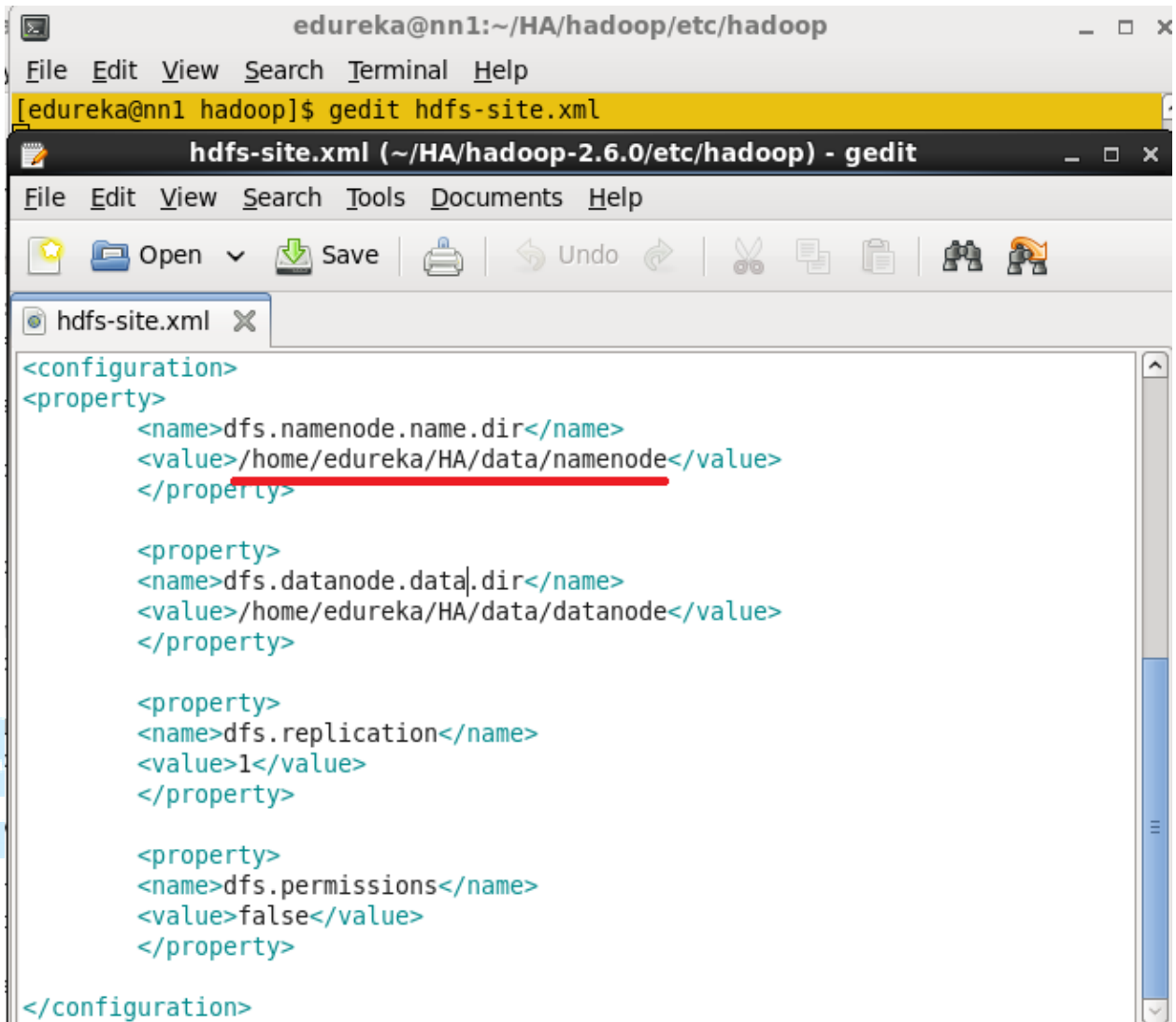
2.8: In the core-site.xml we have to mention the namenode hostname to identify the namenode daemons.

```
<property>
  <name>fs.defaultFS</name>
  <value>hdfs://<NameNode Hostname>:<port number></value>
</property>
```

In my case name node Hostname is **nn1.cluster.com**



2.9: In Namenode hdfs-site.xml Add the Name node's previous hadoop Meta data location for dfs.namenode.name.dir.



The screenshot shows a terminal window at the top with the command `gedit hdfs-site.xml` executed. Below it, the gedit editor displays the XML configuration for `hdfs-site.xml`. The configuration includes several properties, with the first one, `dfs.namenode.name.dir`, highlighted in red. The value for this property is `/home/edureka/HA/data/namenode`.

```
<configuration>
<property>
  <name>dfs.namenode.name.dir</name>
  <value>/home/edureka/HA/data/namenode</value>
</property>

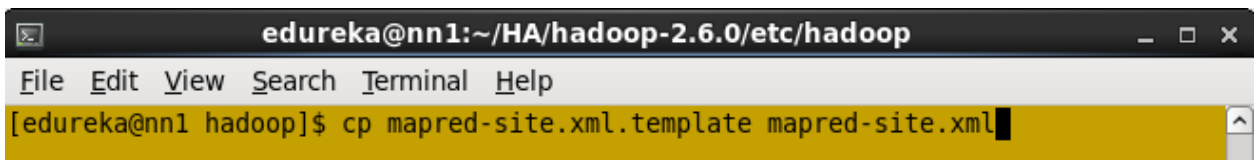
  <property>
    <name>dfs.datanode.data.dir</name>
    <value>/home/edureka/HA/data/datanode</value>
  </property>

  <property>
    <name>dfs.replication</name>
    <value>1</value>
  </property>

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

2.10: Mapred-site.xml

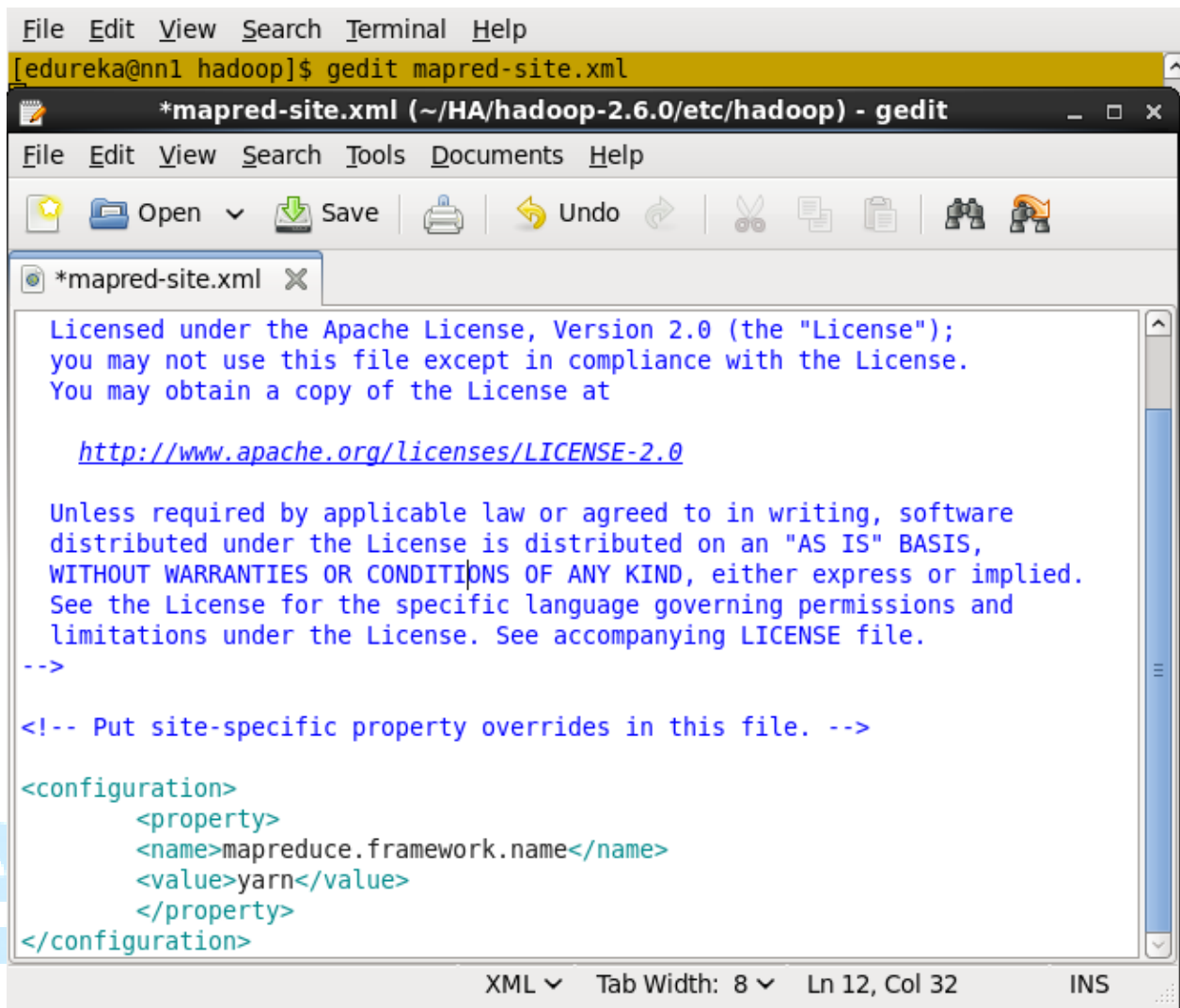
In some cases mapred-site.xml file will not available, you have to create the mapred-site.xml using mapred-site.xml template.



The screenshot shows a terminal window with the command `cp mapred-site.xml.template mapred-site.xml` executed.

```
edureka@nn1:~/HA/hadoop-2.6.0/etc/hadoop
[edureka@nn1 hadoop]$ cp mapred-site.xml.template mapred-site.xml
```

Open the mapred-site.xml file.



```
File Edit View Search Terminal Help
[edureka@nn1 hadoop]$ gedit mapred-site.xml

*mapred-site.xml (~/HA/hadoop-2.6.0/etc/hadoop) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
*mapred-site.xml X
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>
</configuration>

XML Tab Width: 8 Ln 12, Col 32 INS
```

2.11: Edit yarn-site.xml

Open the yarn-site.xml file to add the resource manager properties.

```
<property>
```

```
<name>yarn.nodemanager.aux-services</name>
```

```
<value>mapreduce_shuffle</value>
```

```
</property>
```

```
<property>
```

```
<name>yarn.nodemanager.aux-services.mapreduce_shuffle.class</name>
```

```
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
```

```
</property>
```

```
<property>
```

```
<name>yarn.resourcemanager.resource-tracker.address</name>
```

```
<value><Resource manager Hostname>:9001</value>
```

```
</property>
```

```
<property>
```

```
<name>yarn.resourcemanager.scheduler.address</name>
```

```
<value><Resource manager Hostname>:9002</value>
```

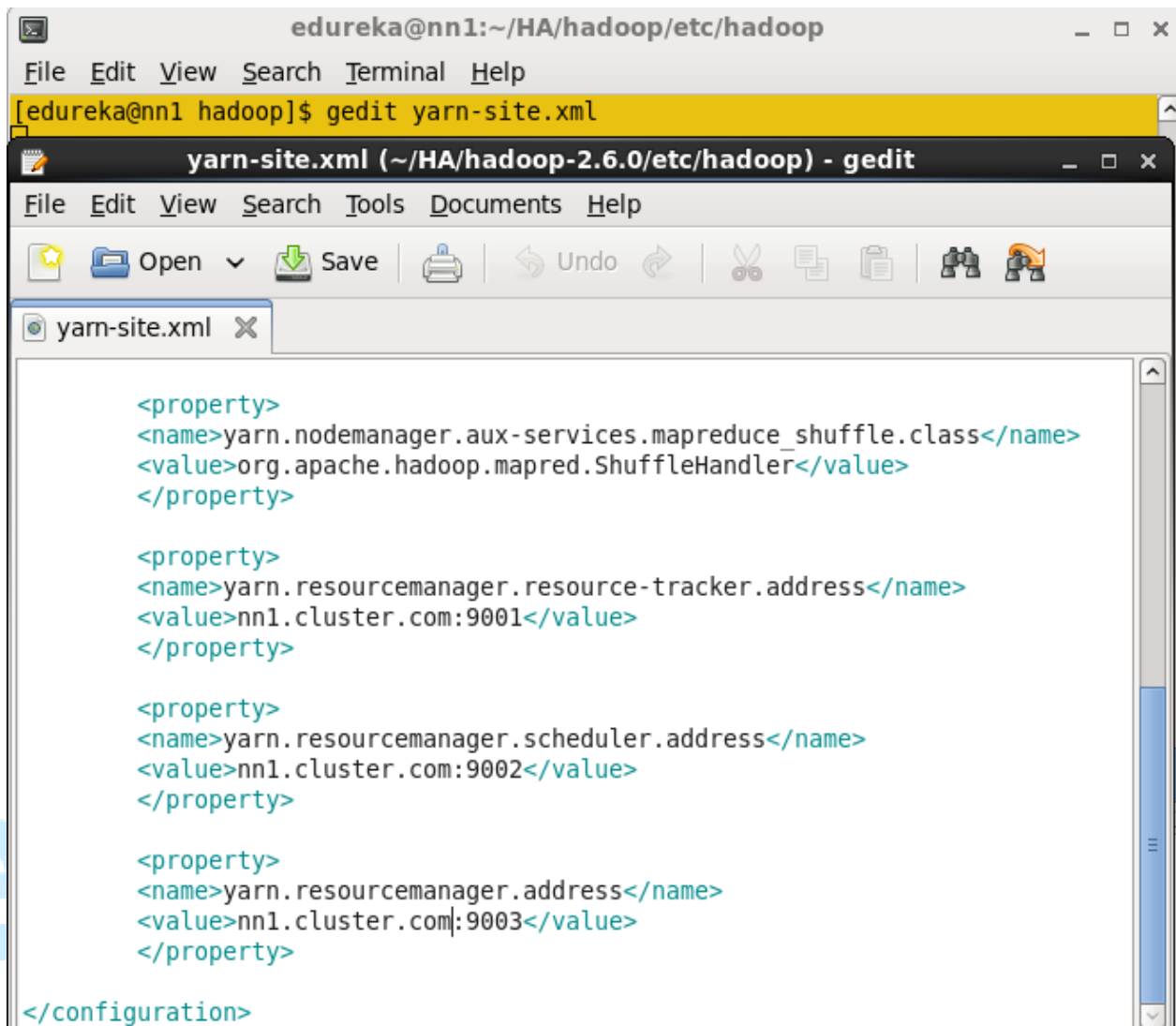
```
</property>
```

```
<property>
```

```
<name>yarn.resourcemanager.address</name>
```

```
<value><Resource manager Hostname>:9003</value>
```

```
</property>
```



The screenshot shows a terminal window at the top with the prompt `edureka@nn1:~/HA/hadoop/etc/hadoop` and the command `[edureka@nn1 hadoop]$ gedit yarn-site.xml`. Below the terminal is a gedit editor window titled `yarn-site.xml (~/HA/hadoop-2.6.0/etc/hadoop) - gedit`. The editor displays the following XML configuration:

```
<property>
<name>yarn.nodemanager.aux-services.mapreduce_shuffle.class</name>
<value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>

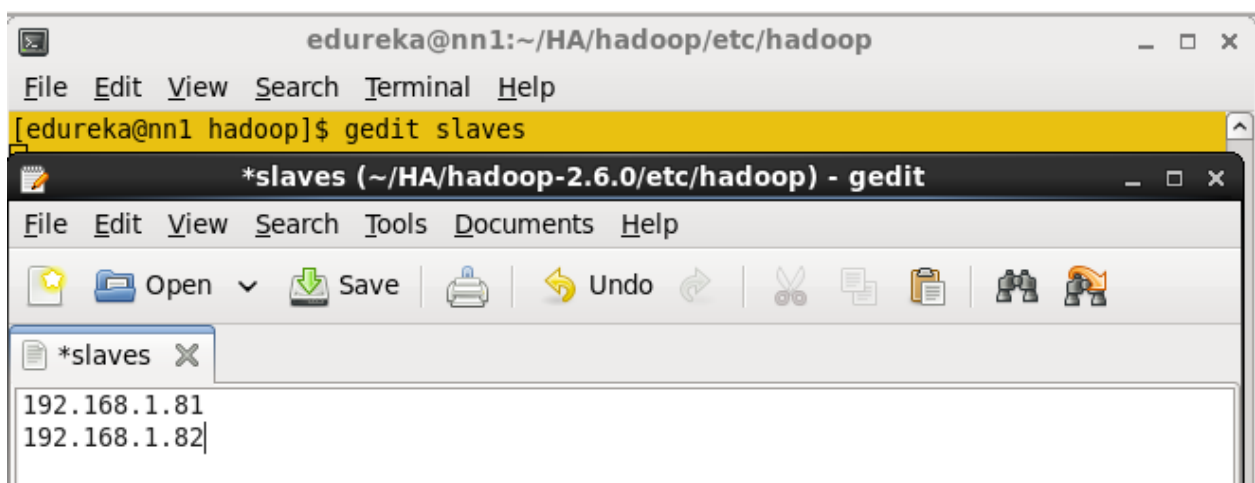
<property>
<name>yarn.resourcemanager.resource-tracker.address</name>
<value>nn1.cluster.com:9001</value>
</property>

<property>
<name>yarn.resourcemanager.scheduler.address</name>
<value>nn1.cluster.com:9002</value>
</property>

<property>
<name>yarn.resourcemanager.address</name>
<value>nn1.cluster.com:9003</value>
</property>

</configuration>
```

2.12: Open slaves file and add the Data nodes IP address.

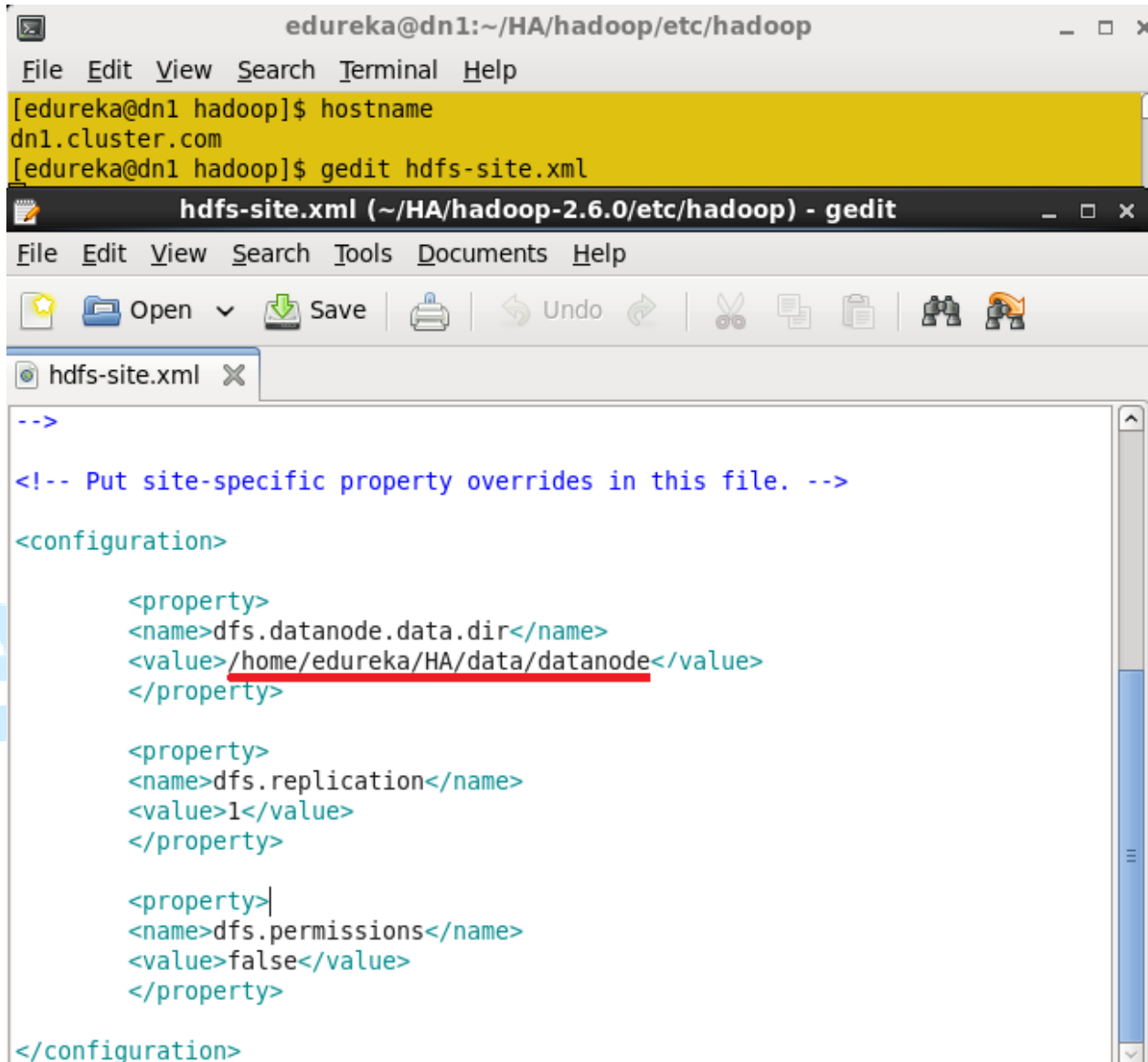


The screenshot shows a terminal window at the top with the prompt `edureka@nn1:~/HA/hadoop/etc/hadoop` and the command `[edureka@nn1 hadoop]$ gedit slaves`. Below the terminal is a gedit editor window titled `*slaves (~/HA/hadoop-2.6.0/etc/hadoop) - gedit`. The editor displays the following content:

```
*slaves
192.168.1.81
192.168.1.82
```

2.13: Copy your Hadoop-2.6.0 directory and .bashrc file to all the datanodes and Edit the hdfs-site.xml, slaves file according to the datanode.

In a datanode the hdfs-site.xml file contain the **dfs.datanode.data.dir** property should contain the previous data node block location.



The screenshot shows a terminal window and a gedit editor window. The terminal window displays the following commands and output:

```
edureka@dn1:~/HA/hadoop/etc/hadoop
File Edit View Search Terminal Help
[edureka@dn1 hadoop]$ hostname
dn1.cluster.com
[edureka@dn1 hadoop]$ gedit hdfs-site.xml
```

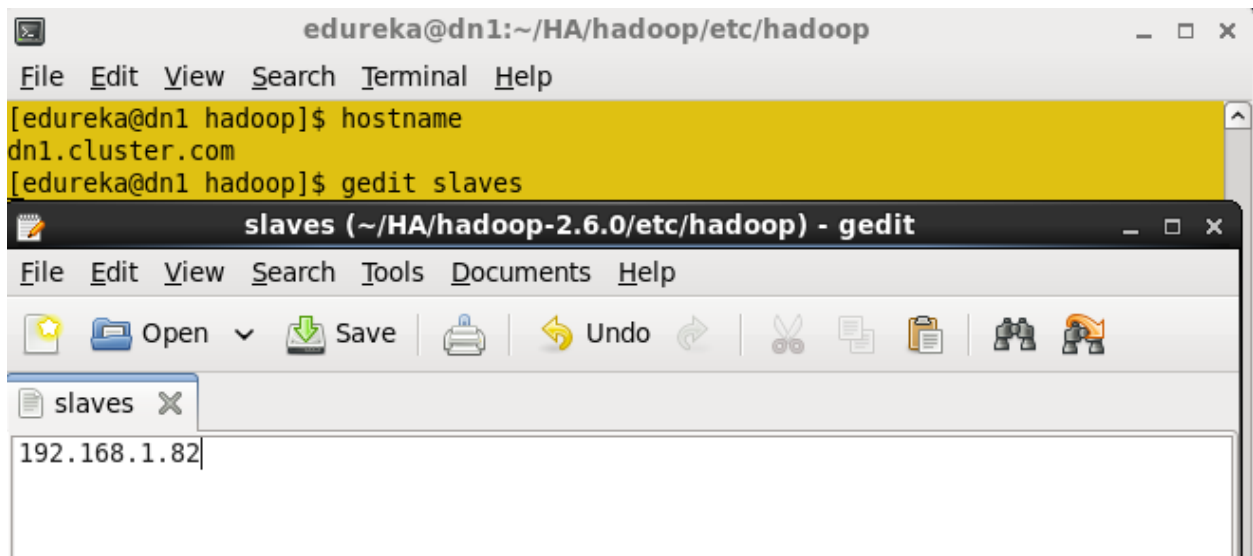
The gedit editor window shows the contents of the hdfs-site.xml file:

```
-->
<!-- Put site-specific property overrides in this file. -->
<configuration>
    <property>
    <name>dfs.datanode.data.dir</name>
    <value>/home/edureka/HA/data/datanode</value>
    </property>

    <property>
    <name>dfs.replication</name>
    <value>1</value>
    </property>

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

2.14: Open the slaves file in each data node and add its IP address.

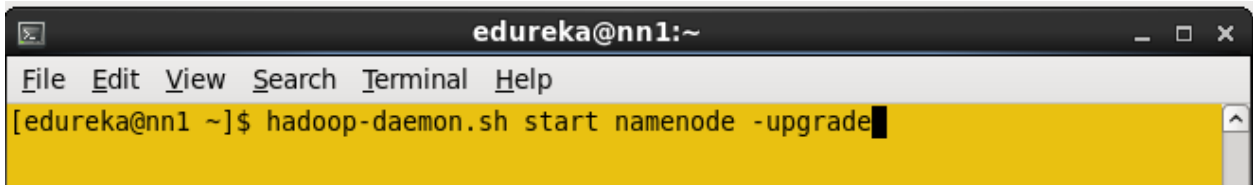


edureka!

Section 3: Upgrading hadoop cluster

3.1: Now start the Name node using below command.

Command: `hadoop-daemon.sh start namenode -upgrade`



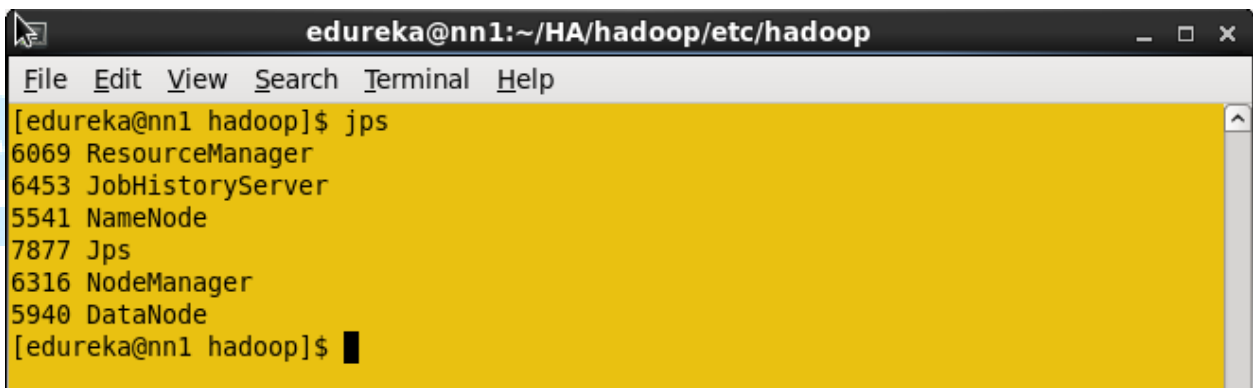
```
edureka@nn1:~  
File Edit View Search Terminal Help  
[edureka@nn1 ~]$ hadoop-daemon.sh start namenode -upgrade
```

3.2: Start all the daemons in using `start-all.sh` or `start-dfs.sh` and `start-yarn.sh` scripts.

Once you run the `start-dfs.sh` file you can see the Name node daemon in master and Data node daemon in slave machines.

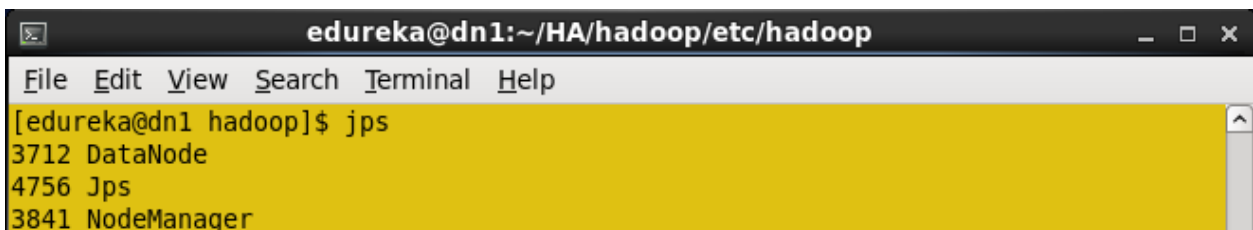
3.3: Once you run the `start-yarn.sh` file you can see the Resource manager daemon in master and Node manager daemon in slave machines.

Daemons in Name node virtual machine is:



```
edureka@nn1:~/HA/hadoop/etc/hadoop  
File Edit View Search Terminal Help  
[edureka@nn1 hadoop]$ jps  
6069 ResourceManager  
6453 JobHistoryServer  
5541 NameNode  
7877 Jps  
6316 NodeManager  
5940 DataNode  
[edureka@nn1 hadoop]$
```

Daemons in datanode virtual machine is:

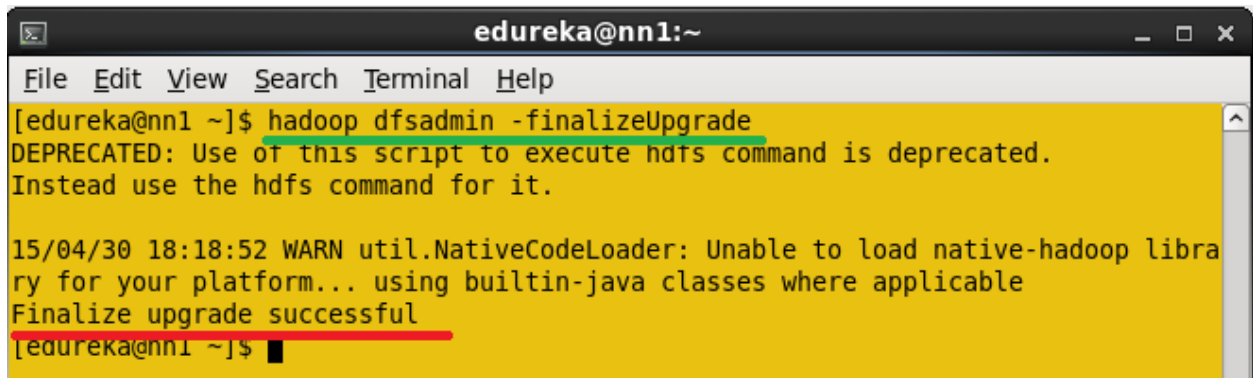


```
edureka@dn1:~/HA/hadoop/etc/hadoop  
File Edit View Search Terminal Help  
[edureka@dn1 hadoop]$ jps  
3712 DataNode  
4756 Jps  
3841 NodeManager
```

3.4: Now once you upgrade the hadoop cluster check whether you got all the files or not, by checking all the HDFS files and its name space details with back up files.

If you think the upgrade the process is fine then you have to finalize the upgrade process.

If you run the finalize command you cannot roll back to previous hadoop version.



```
edureka@nn1:~  
File Edit View Search Terminal Help  
[edureka@nn1 ~]$ hadoop dfsadmin -finalizeUpgrade  
DEPRECATED: Use of this script to execute hdfs command is deprecated.  
Instead use the hdfs command for it.  
  
15/04/30 18:18:52 WARN util.NativeCodeLoader: Unable to load native-hadoop libra  
ry for your platform... using builtin-java classes where applicable  
Finalize upgrade successful  
[edureka@nn1 ~]$
```

edureka!