Created and Edited: Arinjoy Basak, 27/06/2015
Last Edit on 6th October, 2015

The configuration currently used on the system is as follows:

Hadoop: version 2.6.0
Hive: version 1.2.0
Spark: version 1.2.1
and MySQL (for Hive metastore): 5.5

The order of installation is as follows (Schematically):

1. Download and setup Hadoop either as a single node or a cluster.
2. Download and install Hive, create a Hive metastore in MySQL, and make sure Hive is connecting to it.
3. Install Spark.
4. Download the source of Spark, and build it using Hive (configure it first properly, and then using the Simple Build Tool).

General Installation:

The very first thing you need to do is install a version of Java, as both Hadoop, Hive and Spark all use Java. Doing this carefully saves a step in the installation of all the rest.

1. First, install Java on your Linux machine. Try to install Oracle Java, but even if you have OpenJDK, (which comes by default with Ubuntu), it will work just fine. Try to get a hold of openjdk 7 for Java 7.

   The command to install OpenJDK 7 runtime is:

   $ sudo apt-get install openjdk-7-jre
   $ sudo apt-get install openjdk-7-jdk is for the jdk.

   Note that this will source Java 7 in /usr/lib/jvm/java-7-openjdk-amd64.

2. Then, open .bashrc:

   $ vim ~/.bashrc

   and add the following line to the file to specify the path for Java.

   export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64

3. Then, save the file and source it to make it available:

   $. ~/.bashrc

4. Check the Java version to see if it is working properly:

$ java -version

Best Case output:
java version "1.7.0_79"
OpenJDK Runtime Environment (IcedTea 2.5.5) (7u79-2.5.5-0ubuntu0.12.04.1)
OpenJDK 64-Bit Server VM (build 24.79-b02, mixed mode)
$

Congratulations! Java is ready!

5. Firstly, set up a new user account and group strictly for use with Hadoop, Hive and Spark. This is for security issues, and so that any problems in the account or any changes in profiles would not affect the your working user (although, all installations will be in /usr/local folder, for a global access).

   Use the following commands to create the new group 'hadoop', the new user 'hduser' and add hduser to the group hadoop.

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

   This step is actually part of the hadoop installation, but is equally important. Also, remember to grant sudo access to the user 'hduser'.

   $ sudo usermod -a -G sudo hduser

Note: Always login to the hduser account while downloading and installing (especially installing) the things from the terminal, so that you are careful to make changes in the .bashrc file when needed. Add the $JAVA_HOME variable and path to the bashrc file of hduser also. And finally, for running hadoop clusters, hive metastore, spark tasks, or even starting the hive shell, always open through userid 'hduser', where all the settings made.

Whatever be your current login id, you can login to your hduser account using:

$ su – hduser, followed by entering your password.

Note: This is particularly useful if one is using proxy settings and needs to turn them off , which is needed especially during the download processes, during building and sudo apt-get installations. Just disable the proxy settings, and log out and log into the user account for the changes to take place.

Hadoop Installation: (For version 2.6.0, single node clusters)

1. Installing SSH: This is for Hadoop to access all the slave nodes, and most importantly – To start and manage all the HDFS and MapReduce daemons.

   $ sudo apt-get install openssh-server

2. Now, we need to configure the ssh server setup for this machine, in order to be able to access it from this as well as other machines. Log into hduser account (su – hduser), and do the following steps:

   1. Generate ssh key for hduser account

      $ ssh-keygen -t rsa -P ""

   2. Copy id_rsa.pub to authorized keys from hduser

      $ cat $HOME/.ssh/id_rsa.pub >> $HOME/.ssh/authorized_keys

3. Download Hadoop version 2.6.0 from one of the download mirrors available on the Apache website: http://mirrors.gigenet.com/apache/hadoop/common/hadoop-2.6.0/hadoop-2.6.0.tar.gz

4. Go to the location of the installation of hadoop (namely, /usr/local). There, extract the source of Hadoop, and rename it to the folder hadoop:

   $ sudo tar -xzvf $path_to_file_downloaded/hadoop-2.6.0.tar.gz
   $ sudo mv hadoop-2.6.0 /usr/local/hadoop

   Assign ownership of this folder to Hadoop user:

   $ sudo chown hduser:hadoop -R /usr/local/hadoop

   Create Hadoop temp directories for Namenode and Datanode

   $ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/namenode
   $ sudo mkdir -p /usr/local/hadoop_tmp/hdfs/datanode

   Again assign ownership of this Hadoop temp folder to Hadoop user

   $ sudo chown hduser:hadoop -R /usr/local/hadoop_tmp/

5. Now add the following lines to the .bashrc files of hduser:

   export HADOOP_INSTALL=/usr/local/hadoop

```
export PATH=$PATH:$HADOOP_INSTALL/bin
export PATH=$PATH:$HADOOP_INSTALL/sbin
export HADOOP_MAPRED_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_HOME=$HADOOP_INSTALL
export HADOOP_HDFS_HOME=$HADOOP_INSTALL
export YARN_HOME=$HADOOP_INSTALL
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_INSTALL/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_INSTALL/lib"
export PATH=$PATH:$HADOOP_HOME/bin
```

The save and source the file ($. ~/.bashrc)

6. Now, update the following files with the lines as they are mentioned (look for the configurations tag, and paste the properties into them).

   $ cd /usr/local/hadoop/etc/hadoop

Configuration file : core-site.xml

```
## To edit file, fire the below given command
$ /usr/local/hadoop/etc/hadoop$ sudo gedit core-site.xml

## Paste these lines into <configuration> tag
<property>
<name>fs.default.name</name>
<value>hdfs://localhost:9000</value>
</property>
```

Configuration file : hdfs-site.xml

```
## To edit file, fire the below given command
$ /usr/local/hadoop/etc/hadoop$ sudo gedit hdfs-site.xml

## Paste these lines into <configuration> tag
<property>
    <name>dfs.replication</name>
    <value>1</value>
 </property>
 <property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
 </property>
 <property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
 </property>
```

Configuration file : yarn-site.xml

```
## To edit file, fire the below given command
$ /usr/local/hadoop/etc/hadoop$ sudo gedit yarn-site.xml

## Paste these lines into <configuration> tag
```

```
<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>
```

Configuration file : mapred-site.xml

```
## Copy template of mapred-site.xml.template file
cp /usr/local/hadoop/etc/hadoop/mapred-site.xml.template  /usr/local/hadoop/etc/hadoop/mapred-site.xml

## To edit file, fire the below given command
hduser@pingax:/usr/local/hadoop/etc/hadoop$ sudo gedit mapred-site.xml

## Paste these lines into <configuration> tag
<property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
</property>
```

7. Then, format the namenode: $ hdfs namenode -format
8. Start the hadoop and mapreduce daemons through the following commands:

    $ start-dfs.sh
    $ start-yarn.sh

    Which should give the output (start-all.sh also does the same thing):

    Starting namenodes on [localhost]
    localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-namenode-arinjoy-Inspiron-3521.out
    localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-datanode-arinjoy-Inspiron-3521.out
    Starting secondary namenodes [0.0.0.0]
    0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-hduser-secondarynamenode-arinjoy-Inspiron-3521.out
    starting yarn daemons
    starting resourcemanager, logging to /usr/local/hadoop/logs/yarn-hduser-resourcemanager-arinjoy-Inspiron-3521.out
    localhost: starting nodemanager, logging to /usr/local/hadoop/logs/yarn-hduser-nodemanager-arinjoy-Inspiron-3521.out

    To stop the daemons, run stop-dfs.sh and stop-yarn.sh, or stop-all.sh (deprecated).

    localhost: stopping namenode
    localhost: stopping datanode
    Stopping secondary namenodes [0.0.0.0]
    0.0.0.0: stopping secondarynamenode

stopping yarn daemons
stopping resourcemanager
resourcemanager did not stop gracefully after 5 seconds: killing with kill -9
localhost: stopping nodemanager
localhost: nodemanager did not stop gracefully after 5 seconds: killing with kill -9
no proxyserver to stop

ERROR: If on start-dfs.sh, or at any stage, the following error occurs

```
deploy@olympus:~$ start-dfs.sh
Starting namenodes on [olympus]
olympus: Error: JAVA_HOME is not set and could not be found.
```

Open hadoop-env.sh in /usr/local/hadoop/etc/hadoop, and specify hadoop to use the JAVA_HOME in the path provided. Add the following line to hadoop-env.sh and save.

export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64

9. Verify the processes with jps: $ jps

Which should show:

6262 SecondaryNameNode
5721 NameNode
8060 RunJar
5970 DataNode
25681 Jps
6414 ResourceManager
6666 NodeManager

And hadoop is all set up! Now run a simple word count program on hadoop to verify if the installation is correct or not. For example, use the streamer api in a Python code to create a mapper and reducer available on the michaelnoll tutorial, and write the following command to run them (considering the filenames are the same as the ones in the tutorial)

$ bin/hadoop jar /usr/local/hadoop/share/hadoop/tools/lib/hadoop-*streaming*.jar -file /home/hduser/mapper.py -mapper /home/hduser/mapper.py -file /home/hduser/reducer.py -reducer /home/hduser/reducer.py -input /user/hduser/gutenberg/* -output /user/hduser/gutenberg-output

Then, access the files as per the hdfs commands to read them and verify them.

Hive Installation: (For version 1.2.0)

1. Download the version 1.2.0 of Hive from the Apache Hive website:
   http://apache.mirrors.lucidnetworks.net/hive/stable/apache-hive-1.2.0-bin.tar.gz
   (This is just one of the mirrors) Try any of them.

2. Extract the file, and copy it to the /usr/local folder for installation. Then change the owner to hduser, who will access it.

   ```
   sudo su
   cd /usr/local
   cp /home//Download/apache-hive-1.2.0-bin.tar.gz /usr/local/
   tar -xvzf apache-hive-1.2.0-bin.tar.gz
   mv apache-hive-1.2.0-bin hive
   chown -R hduser:hadoop hive
   ```

3. Update /etc/profile PATH or su hduser: vi ~/.bashrc

   ```
   export HIVE_HOME=/usr/local/hive
   export PATH=$PATH:$HADOOP_HOME/bin:$HIVE_HOME/bin
   ```

   #Some errors may occur, due to being unable to find the java libraries for jLine Terminals (or something like that), in which case, add these to the end of .bashrc

   ```
   export HADOOP_USER_CLASSPATH_FIRST=true
   export JAVA_LIBRARY_PATH=$HADOOP_HOME/lib/native:$JAVA_LIBRARY_PATH
   ```

   This might also be a solution for Hadoop errors like this:

15/06/27 01:39:52 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

   Then, source the file.

4. In addition, you must create /tmp and /user/hive/warehouse (aka hive.metastore.warehouse.dir) and set them chmod g+w in HDFS before you can create a table in Hive.

   ```
   $ $HADOOP_HOME/bin/hadoop fs -mkdir       /tmp
   $ $HADOOP_HOME/bin/hadoop fs -mkdir       /user/hive/warehouse
   $ $HADOOP_HOME/bin/hadoop fs -chmod g+w   /tmp
   $ $HADOOP_HOME/bin/hadoop fs -chmod g+w   /user/hive/warehouse
   ```

5. Remove .template extension from all the files stored in $HIVE_HOME/conf folder. (*Ex : mv hive-env.sh.template hive-env.sh*)

6. Now, to configure hive to point to the metastore. This will install the jar for mysql connector through java, cd /usr/share/java/ to see the mysql-connector-java.jar)

   ```
   $ sudo apt-get install mysql-server-5.5
   ```

$ sudo apt-get install libmysql-java

7. Then, Download the latest version of the mysql connector from the dev.mysql website, and copy it into the /usr/local/hive/lib/ folder.

8. Create the initial database schema using the hive-schema-

```
$ mysql -uroot -proot
mysql>CREATE DATABASE metastore;
mysql>USE metastore;
mysql> SOURCE /usr/local/hive/scripts/metastore/upgrade/mysql/hive-schema-1.2.0.mysql.sql
mysql> CREATE USER 'hive'@'metastorehost' IDENTIFIED BY 'mypassword';
mysql> REVOKE ALL PRIVILEGES, GRANT OPTION FROM 'hive'@'metastorehost';
mysql> GRANT SELECT,INSERT,UPDATE,DELETE,LOCK TABLES,EXECUTE ON metastore.*
TO 'hive'@'metastorehost';
mysql> FLUSH PRIVILEGES;
mysql> quit;
```

9. Inside hive/conf, create a file called hive-site.xml, and write the following properties into it.

```
<configuration>
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:mysql://localhost/metastore</value>
  <description>the URL of the MySQL database</description>
</property>
<property>
  <name>javax.jdo.option.ConnectionDriverName</name>
  <value>com.mysql.jdbc.Driver</value>
</property>
<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>hiveuser</value>
</property>
<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>hivepassword</value>
</property>
<property>
  <name>datanucleus.autoCreateSchema</name>
  <value>false</value>
</property>
<property>
  <name>datanucleus.fixedDatastore</name>
  <value>true</value>
</property>

<property>
      <name>hive.metastore.uris</name>
```

```
    <value>thrift://localhost:9083</value>
</property>

</configuration>
```

10. To start the Hive Metastore, run the following command:

    $ hive --service metastore & or
    $ $HIVE_HOME/bin/hive --service metastore &
    and press Enter. The metastore is started in the background.

Note: In case of the following error, review the hive-site.xml, and see the structure and syntax of the file in the positions given... maybe the tags are missing, or properties are out of the configuration tags.

```
Starting Hive Metastore Server
[Fatal Error] hive-site.xml:7:2: The markup in the document following the root element must be well-formed.
15/06/25 15:22:27 FATAL conf.Configuration: error parsing conf file:/usr/local/hive/conf/hive-site.xml
org.xml.sax.SAXParseException; systemId: file:/usr/local/hive/conf/hive-site.xml; lineNumber: 7; columnNumber: 2; The markup in the document following the root element must be well-formed.
        at org.apache.xerces.parsers.DOMParser.parse(Unknown Source)
        at org.apache.xerces.jaxp.DocumentBuilderImpl.parse(Unknown Source)
        at javax.xml.parsers.DocumentBuilder.parse(DocumentBuilder.java:150)
        at org.apache.hadoop.conf.Configuration.parse(Configuration.java:2352)
        at org.apache.hadoop.conf.Configuration.parse(Configuration.java:2340)
        at org.apache.hadoop.conf.Configuration.loadResource(Configuration.java:2408)
        at org.apache.hadoop.conf.Configuration.loadResources(Configuration.java:2374)
        at org.apache.hadoop.conf.Configuration.getProps(Configuration.java:2281)
        at org.apache.hadoop.conf.Configuration.get(Configuration.java:1108)
        at org.apache.hadoop.hive.conf.HiveConf.getVar(HiveConf.java:2605)
        at org.apache.hadoop.hive.conf.HiveConf.getVar(HiveConf.java:2626)
        at org.apache.hadoop.hive.conf.HiveConf.initialize(HiveConf.java:2696)
        at org.apache.hadoop.hive.conf.HiveConf.<init>(HiveConf.java:2641)
        at org.apache.hadoop.hive.common.LogUtils.initHiveLog4jCommon(LogUtils.java:74)
        at org.apache.hadoop.hive.common.LogUtils.initHiveLog4j(LogUtils.java:58)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5841)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
Exception in thread "main" java.lang.RuntimeException: org.xml.sax.SAXParseException; systemId: file:/usr/local/hive/conf/hive-site.xml; lineNumber: 7; columnNumber: 2; The markup in the document following the root element must be well-formed.
        at org.apache.hadoop.conf.Configuration.loadResource(Configuration.java:2517)
        at org.apache.hadoop.conf.Configuration.loadResources(Configuration.java:2374)
        at org.apache.hadoop.conf.Configuration.getProps(Configuration.java:2281)
        at org.apache.hadoop.conf.Configuration.get(Configuration.java:1108)
```

```
        at org.apache.hadoop.hive.conf.HiveConf.getVar(HiveConf.java:2605)
        at org.apache.hadoop.hive.conf.HiveConf.getVar(HiveConf.java:2626)
        at org.apache.hadoop.hive.conf.HiveConf.initialize(HiveConf.java:2696)
        at org.apache.hadoop.hive.conf.HiveConf.<init>(HiveConf.java:2641)
        at org.apache.hadoop.hive.common.LogUtils.initHiveLog4jCommon(LogUtils.java:74)
        at org.apache.hadoop.hive.common.LogUtils.initHiveLog4j(LogUtils.java:58)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5841)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
Caused by: org.xml.sax.SAXParseException; systemId: file:/usr/local/hive/conf/hive-site.xml;
lineNumber: 7; columnNumber: 2; The markup in the document following the root element must be
well-formed.
        at org.apache.xerces.parsers.DOMParser.parse(Unknown Source)
        at org.apache.xerces.jaxp.DocumentBuilderImpl.parse(Unknown Source)
        at javax.xml.parsers.DocumentBuilder.parse(DocumentBuilder.java:150)
        at org.apache.hadoop.conf.Configuration.parse(Configuration.java:2352)
        at org.apache.hadoop.conf.Configuration.parse(Configuration.java:2340)
        at org.apache.hadoop.conf.Configuration.loadResource(Configuration.java:2408)
        ... 16 more
```

Note: If this is the error you get on running hive metastore, then stop the hive metastore services that are already running using “ps -ef | grep 'hive' and then 'kill -9 <pid>' with the process id's... it means that the ports are being used by other instances of the server.

```
ls: cannot access /usr/local/spark-1.2.1/lib/spark-assembly-*.jar: No such file or directory
Starting Hive Metastore Server
org.apache.thrift.transport.TTransportException: Could not create ServerSocket on address
0.0.0.0/0.0.0.0:9083.
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:109)
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:91)
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:83)
        at
org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerSocketKeepAlive.java:34)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaStore.java:5936)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5877)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
Exception in thread "main" org.apache.thrift.transport.TTransportException: Could not create
ServerSocket on address 0.0.0.0/0.0.0.0:9083.
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:109)
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:91)
```

```
        at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:83)
        at
org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerSocketKeepAlive.java:34)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaStore.java:5936)
        at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5877)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
        at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
```

For example:

```
$ ps -ef | grep -e 'hive'
hduser   11314     1  0 Jun24 ?        00:00:58 /usr/lib/jvm/java-7-openjdk-amd64/bin/java -Xmx256m
-Djava.net.preferIPv4Stack=true -Dhadoop.log.dir=/usr/local/hadoop/logs
-Dhadoop.log.file=hadoop.log -Dhadoop.home.dir=/usr/local/hadoop -Dhadoop.id.str=hduser
-Dhadoop.root.logger=INFO,console
-Djava.library.path=/usr/local/hadoop/lib/native:/usr/local/hadoop/lib/native:/usr/local/hadoop/lib/nativ
e: -Dhadoop.policy.file=hadoop-policy.xml -Djava.net.preferIPv4Stack=true -Xmx512m
-Dhadoop.security.logger=INFO,NullAppender org.apache.hadoop.util.RunJar /usr/local/hive-
1.2.0/lib/hive-service-1.2.0.jar org.apache.hadoop.hive.metastore.HiveMetaStore
hduser   13153 12357 12 15:31 pts/3    00:00:06 /usr/lib/jvm/java-7-openjdk-amd64/bin/java
-Xmx256m -Djava.net.preferIPv4Stack=true -Dhadoop.log.dir=/usr/local/hadoop/logs
-Dhadoop.log.file=hadoop.log -Dhadoop.home.dir=/usr/local/hadoop -Dhadoop.id.str=hduser
-Dhadoop.root.logger=INFO,console
-Djava.library.path=/usr/local/hadoop/lib/native:/usr/local/hadoop/lib/native:
-Dhadoop.policy.file=hadoop-policy.xml -Djava.net.preferIPv4Stack=true -Xmx512m
-Dhadoop.security.logger=INFO,NullAppender org.apache.hadoop.util.RunJar /usr/local/hive/lib/hive-
service-1.2.0.jar org.apache.hadoop.hive.metastore.HiveMetaStore
hduser   13232 12357  0 15:32 pts/3    00:00:00 grep --color=auto -e hive
hduser   30753     1  0 Jun24 pts/0    00:00:23 /usr/lib/jvm/java-7-openjdk-amd64/bin/java -Xmx256m
-Djava.net.preferIPv4Stack=true -Dhadoop.log.dir=/usr/local/hadoop/logs
-Dhadoop.log.file=hadoop.log -Dhadoop.home.dir=/usr/local/hadoop -Dhadoop.id.str=hduser
-Dhadoop.root.logger=INFO,console -Djava.library.path=/usr/local/hadoop/lib/native:
-Dhadoop.policy.file=hadoop-policy.xml -Djava.net.preferIPv4Stack=true -Xmx512m
-Dhadoop.security.logger=INFO,NullAppender org.apache.hadoop.util.RunJar /usr/local/hive-
1.2.0/lib/hive-service-1.2.0.jar org.apache.hadoop.hive.metastore.HiveMetaStore
$ /usr/local/hive/conf$ kill 11314
```

11. Create *vi /home/hduser/user.txt* in your local file system and add below content or download

```
userid,username,city,state,country
1,John,Montgomery,Alabama,US
2,David,Phoenix,Arizona,US
3,Sarah,Sacramento,California,US
```

4,Anoop,Montgomery,Alabama,US
5,Gubs,Villupuram,TamilNadu,India

Note : In hive, table names are all case insensitive

12. Go to hive prompt and we'll create the table users in the Hive MetaStore to map data from user.txt

```
$ hive (Enter)
hive>CREATE TABLE user(id INT, name STRING, City STRING, State STRING, Country STRING)
 ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
 LINES TERMINATED BY '\n' STORED AS TEXTFILE;
hive>show tables;
```

13. The following command maps user.txt data to the users table by loading data from user.txt. (LOCAL => keyword needed to load file from local into hive. Not necessary if your file is in hdfs) You can add OVERWRITE prior to 'INTO TABLE' if you want to overwrite existing user table content.
    You can use : hadoop fs -put <localfilesystempath> <hdfsfiledirectory>

Load data into hive from your local directory. Remove LOCAL keyword if you file is in hdfs.

```
$hive>LOAD DATA LOCAL INPATH '/home/hduser/user.txt' INTO TABLE user;
```

Note : Load Local command copies the file from local location to /user/hive/warehouse/userdb/user/user.txt. When you execute "DROP TABLE user" the file will also be dropped/removed from the hive location /user/hive/warehouse/userdb/user/user.txt.

Query How many people belong to each state?

```
$hive>select state, count(state) from user group by state;
```

12. So, hive is giving the correct answer in results. But, is it working correctly? To check, go to mysql, and enter as root or as the hive user. Select the metastore database, and do: “Select * from TBLS” after creating the hive table. If the table is created, then it will appear in the results of the output. Else, go back to the hive-site.xml, and review it for errors. Add the connector properly to the HIVE_HOME/lib folder. Then, if all else fails, remove the installation of hie, and reinstall it again.

Spark Installation: (For version 1.2.1)

1. Now that Java is installed, we need to install Scala for Spark installation. First, download the version of spark that is required: http://downloads.typesafe.com/scala/2.10.5/scala-2.10.5.tgz?_ga=1.185786607.772777287.1434964699

2. Then, make a separate directory for scala in /usr/local/src/ and unpack the file in that location

   $ sudo mkdir /usr/local/src/scala
   $ sudo tar xvf scala-2.10.4.tgz -C /usr/local/src/scala/

3. Then open ~/.bashrc (inside the hduser profile), and edit to add the file following lines, followed by sourcing the file.

   export SCALA_HOME=/usr/local/src/scala/scala-2.10.4
   export PATH=$SCALA_HOME/bin:$PATH

   Then, restart bashrc: $ . .bashrc
   Then, check the version of scala installed using scala -version.

4. Then, download and install git, and configure it: sudo apt-get install git

5. Finally, download the package of spark 1.2.1 from here: http://apache.mirrors.lucidnetworks.net/spark/spark-1.2.1/spark-1.2.1.tgz

6. Then, unpack the file, and place the folder in /usr/local/. Then add the following line to .bashrc:

   export SPARK_HOME=/usr/local/spark-1.2.1

7. Now, we are required to build the Spark distribution using Hive. Given that hive is working correctly, we will use sbt (Simple Build tool) to build spark. SBT comes preloaded with spark, and is in the SPARK_HOME/sbt folder.

8. Copy the hive-site.xml file in hive/conf folder to spark conf folder and then build it. Any changes should be in both of them. This allows Spark to be built using the configurations of Hive.

9. Also, copy the mysql-connector.jar file into spark-1.2.1/lib_managed/jars/ folder for Spark to use. This will help to avoid the following error (there are additional steps as well). Also, If you see the following error while executing a Spark script using Hive tables, make sure hive metastore service is running.

```
Spark assembly has been built with Hive, including Datanucleus jars on classpath
Traceback (most recent call last):
  File "/home/hduser/sampleHiveSpark.py", line 9, in <module>
    sqlContext.sql("CREATE TABLE IF NOT EXISTS src (key INT, value STRING)")
  File "/usr/local/spark-1.2.1/python/pyspark/sql.py", line 1620, in sql
    return SchemaRDD(self._ssql_ctx.sql(sqlQuery).toJavaSchemaRDD(), self)
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/java_gateway.py", line 538, in
```

```
__call__
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/protocol.py", line 300, in
get_return_value
py4j.protocol.Py4JJavaError: An error occurred while calling o19.sql.
: java.lang.RuntimeException: java.lang.RuntimeException: Unable to instantiate
org.apache.hadoop.hive.metastore.HiveMetaStoreClient
        at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:346)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:235)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:231)
        at scala.Option.orElse(Option.scala:257)
        at org.apache.spark.sql.hive.HiveContext.x$3$lzycompute(HiveContext.scala:231)
        at org.apache.spark.sql.hive.HiveContext.x$3(HiveContext.scala:229)
        at org.apache.spark.sql.hive.HiveContext.hiveconf$lzycompute(HiveContext.scala:229)
        at org.apache.spark.sql.hive.HiveContext.hiveconf(HiveContext.scala:229)
        at org.apache.spark.sql.hive.HiveMetastoreCatalog.<init>(HiveMetastoreCatalog.scala:55)
        at org.apache.spark.sql.hive.HiveContext$$anon$2.<init>(HiveContext.scala:253)
        at org.apache.spark.sql.hive.HiveContext.catalog$lzycompute(HiveContext.scala:253)
        at org.apache.spark.sql.hive.HiveContext.catalog(HiveContext.scala:253)
        at org.apache.spark.sql.hive.HiveContext$$anon$4.<init>(HiveContext.scala:263)
        at org.apache.spark.sql.hive.HiveContext.analyzer$lzycompute(HiveContext.scala:263)
        at org.apache.spark.sql.hive.HiveContext.analyzer(HiveContext.scala:262)
        at
org.apache.spark.sql.SQLContext$QueryExecution.analyzed$lzycompute(SQLContext.scala:411)
        at org.apache.spark.sql.SQLContext$QueryExecution.analyzed(SQLContext.scala:411)
        at org.apache.spark.sql.SchemaRDDLike$class.$init$(SchemaRDDLike.scala:58)
        at org.apache.spark.sql.SchemaRDD.<init>(SchemaRDD.scala:108)
        at org.apache.spark.sql.hive.HiveContext.sql(HiveContext.scala:94)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:231)
        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:379)
        at py4j.Gateway.invoke(Gateway.java:259)
        at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:133)
        at py4j.commands.CallCommand.execute(CallCommand.java:79)
        at py4j.GatewayConnection.run(GatewayConnection.java:207)
        at java.lang.Thread.run(Thread.java:745)
Caused by: java.lang.RuntimeException: Unable to instantiate
org.apache.hadoop.hive.metastore.HiveMetaStoreClient
        at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1412)
        at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:62)
        at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:72
)
        at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:2453)
        at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:2465)
        at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:340)
```

```
        ... 30 more
Caused by: java.lang.reflect.InvocationTargetException
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
        at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1410)
        ... 35 more
Caused by: javax.jdo.JDOFatalInternalException: Error creating transactional connection factory
NestedThrowables:
java.lang.reflect.InvocationTargetException
        at
org.datanucleus.api.jdo.NucleusJDOHelper.getJDOExceptionForNucleusException(NucleusJDOHelper
.java:587)
        at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:788)
        at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.createPersistenceManagerFactory(JDOPersiste
nceManagerFactory.java:333)
        at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.getPersistenceManagerFactory(JDOPersistenc
eManagerFactory.java:202)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at javax.jdo.JDOHelper$16.run(JDOHelper.java:1965)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.jdo.JDOHelper.invoke(JDOHelper.java:1960)
        at
javax.jdo.JDOHelper.invokeGetPersistenceManagerFactoryOnImplementation(JDOHelper.java:1166)
        at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:808)
        at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:701)
        at org.apache.hadoop.hive.metastore.ObjectStore.getPMF(ObjectStore.java:310)
        at org.apache.hadoop.hive.metastore.ObjectStore.getPersistenceManager(ObjectStore.java:339)
        at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:248)
        at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:223)
        at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
        at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
        at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:58)
        at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:67)
        at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:49
7)
        at
```

```
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:475)
        at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.java:
523)
        at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:397)
        at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.<init>(HiveMetaStore.java:356)
        at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:54)
        at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:59)
        at
org.apache.hadoop.hive.metastore.HiveMetaStore.newHMSHandler(HiveMetaStore.java:4944)
        at
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:171)
        ... 40 more
Caused by: java.lang.reflect.InvocationTargetException
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
        at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
        at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:325)
        at
org.datanucleus.store.AbstractStoreManager.registerConnectionFactory(AbstractStoreManager.java:28
2)
        at org.datanucleus.store.AbstractStoreManager.<init>(AbstractStoreManager.java:240)
        at org.datanucleus.store.rdbms.RDBMSStoreManager.<init>(RDBMSStoreManager.java:286)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
        at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
        at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:301)
        at
org.datanucleus.NucleusContext.createStoreManagerForProperties(NucleusContext.java:1187)
        at org.datanucleus.NucleusContext.initialise(NucleusContext.java:356)
        at
```

```
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:775)
        ... 69 more
Caused by: org.datanucleus.exceptions.NucleusException: Attempt to invoke the "BONECP" plugin to
create a ConnectionPool gave an error : The specified datastore driver ("com.mysql.jdbc.Driver") was
not found in the CLASSPATH. Please check your CLASSPATH specification, and the name of the
driver.
        at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java
:259)
        at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.initialiseDataSources(ConnectionFactoryImpl.jav
a:131)
        at org.datanucleus.store.rdbms.ConnectionFactoryImpl.<init>(ConnectionFactoryImpl.java:85)
        ... 87 more
Caused by: org.datanucleus.store.rdbms.connectionpool.DatastoreDriverNotFoundException: The
specified datastore driver ("com.mysql.jdbc.Driver") was not found in the CLASSPATH. Please check
your CLASSPATH specification, and the name of the driver.
        at
org.datanucleus.store.rdbms.connectionpool.AbstractConnectionPoolFactory.loadDriver(AbstractConn
ectionPoolFactory.java:58)
        at
org.datanucleus.store.rdbms.connectionpool.BoneCPConnectionPoolFactory.createConnectionPool(Bo
neCPConnectionPoolFactory.java:54)
        at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java
:238)
        ... 89 more

hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1$ ./bin/spark-submit ~/sampleHiveSpark.py
Spark assembly has been built with Hive, including Datanucleus jars on classpath
Traceback (most recent call last):
  File "/home/hduser/sampleHiveSpark.py", line 9, in <module>
    sqlContext.sql("CREATE TABLE IF NOT EXISTS src (key INT, value STRING)")
  File "/usr/local/spark-1.2.1/python/pyspark/sql.py", line 1620, in sql
    return SchemaRDD(self._ssql_ctx.sql(sqlQuery).toJavaSchemaRDD(), self)
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/java_gateway.py", line 538, in
__call__
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/protocol.py", line 300, in
get_return_value
py4j.protocol.Py4JJavaError: An error occurred while calling o19.sql.
: java.lang.RuntimeException: java.lang.RuntimeException: Unable to instantiate
org.apache.hadoop.hive.metastore.HiveMetaStoreClient
        at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:346)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:235)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:231)
        at scala.Option.orElse(Option.scala:257)
        at org.apache.spark.sql.hive.HiveContext.x$3$lzycompute(HiveContext.scala:231)
        at org.apache.spark.sql.hive.HiveContext.x$3(HiveContext.scala:229)
```

```
	at org.apache.spark.sql.hive.HiveContext.hiveconf$lzycompute(HiveContext.scala:229)
	at org.apache.spark.sql.hive.HiveContext.hiveconf(HiveContext.scala:229)
	at org.apache.spark.sql.hive.HiveMetastoreCatalog.<init>(HiveMetastoreCatalog.scala:55)
	at org.apache.spark.sql.hive.HiveContext$$anon$2.<init>(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext.catalog$lzycompute(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext.catalog(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext$$anon$4.<init>(HiveContext.scala:263)
	at org.apache.spark.sql.hive.HiveContext.analyzer$lzycompute(HiveContext.scala:263)
	at org.apache.spark.sql.hive.HiveContext.analyzer(HiveContext.scala:262)
	at
org.apache.spark.sql.SQLContext$QueryExecution.analyzed$lzycompute(SQLContext.scala:411)
	at org.apache.spark.sql.SQLContext$QueryExecution.analyzed(SQLContext.scala:411)
	at org.apache.spark.sql.SchemaRDDLike$class.$init$(SchemaRDDLike.scala:58)
	at org.apache.spark.sql.SchemaRDD.<init>(SchemaRDD.scala:108)
	at org.apache.spark.sql.hive.HiveContext.sql(HiveContext.scala:94)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:606)
	at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:231)
	at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:379)
	at py4j.Gateway.invoke(Gateway.java:259)
	at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:133)
	at py4j.commands.CallCommand.execute(CallCommand.java:79)
	at py4j.GatewayConnection.run(GatewayConnection.java:207)
	at java.lang.Thread.run(Thread.java:745)
Caused by: java.lang.RuntimeException: Unable to instantiate
org.apache.hadoop.hive.metastore.HiveMetaStoreClient
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1412)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:62)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:72
)
	at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:2453)
	at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:2465)
	at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:340)
	... 30 more
Caused by: java.lang.reflect.InvocationTargetException
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1410)
	... 35 more
Caused by: javax.jdo.JDOFatalInternalException: Error creating transactional connection factory
```

NestedThrowables:
java.lang.reflect.InvocationTargetException
	at
org.datanucleus.api.jdo.NucleusJDOHelper.getJDOExceptionForNucleusException(NucleusJDOHelper
.java:587)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:788)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.createPersistenceManagerFactory(JDOPersiste
nceManagerFactory.java:333)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.getPersistenceManagerFactory(JDOPersistenc
eManagerFactory.java:202)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:606)
	at javax.jdo.JDOHelper$16.run(JDOHelper.java:1965)
	at java.security.AccessController.doPrivileged(Native Method)
	at javax.jdo.JDOHelper.invoke(JDOHelper.java:1960)
	at
javax.jdo.JDOHelper.invokeGetPersistenceManagerFactoryOnImplementation(JDOHelper.java:1166)
	at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:808)
	at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:701)
	at org.apache.hadoop.hive.metastore.ObjectStore.getPMF(ObjectStore.java:310)
	at org.apache.hadoop.hive.metastore.ObjectStore.getPersistenceManager(ObjectStore.java:339)
	at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:248)
	at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:223)
	at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
	at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:58)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:67)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:49
7)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:475)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.java:
523)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:397)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.<init>(HiveMetaStore.java:356)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:54)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:59)

```
	at
org.apache.hadoop.hive.metastore.HiveMetaStore.newHMSHandler(HiveMetaStore.java:4944)
	at
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:171)
	... 40 more
Caused by: java.lang.reflect.InvocationTargetException
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
	at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:325)
	at
org.datanucleus.store.AbstractStoreManager.registerConnectionFactory(AbstractStoreManager.java:28
2)
	at org.datanucleus.store.AbstractStoreManager.<init>(AbstractStoreManager.java:240)
	at org.datanucleus.store.rdbms.RDBMSStoreManager.<init>(RDBMSStoreManager.java:286)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
	at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:301)
	at
org.datanucleus.NucleusContext.createStoreManagerForProperties(NucleusContext.java:1187)
	at org.datanucleus.NucleusContext.initialise(NucleusContext.java:356)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:775)
	... 69 more
Caused by: org.datanucleus.exceptions.NucleusException: Attempt to invoke the "BONECP" plugin to
create a ConnectionPool gave an error : The specified datastore driver ("com.mysql.jdbc.Driver") was
not found in the CLASSPATH. Please check your CLASSPATH specification, and the name of the
driver.
	at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java
:259)
	at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.initialiseDataSources(ConnectionFactoryImpl.jav
```

a:131)
        at org.datanucleus.store.rdbms.ConnectionFactoryImpl.<init>(ConnectionFactoryImpl.java:85)
        ... 87 more
Caused by: org.datanucleus.store.rdbms.connectionpool.DatastoreDriverNotFoundException: The specified datastore driver ("com.mysql.jdbc.Driver") was not found in the CLASSPATH. Please check your CLASSPATH specification, and the name of the driver.
        at
org.datanucleus.store.rdbms.connectionpool.AbstractConnectionPoolFactory.loadDriver(AbstractConnectionPoolFactory.java:58)
        at
org.datanucleus.store.rdbms.connectionpool.BoneCPConnectionPoolFactory.createConnectionPool(BoneCPConnectionPoolFactory.java:54)
        at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java:238)
        ... 89 more

10. Change into the Spark directory (cd /usr/local/spark-1.2.1) and execute the following command to build it with the latest version of Hive (1.2.0) and Hadoop (2.6.0) installed.

    $ sudo sbt/sbt -Phive -Pyarn -Phadoop-2.6 -Dhadoop.version=2.6.0 assembly

    This should take some time (1 hr, or more), depending on the internet connection and the processor speed. A lot of things will be downloaded, and it will show several jar files being downloaded,

    And in any case the build is not successful due to a large number of errors, such as Server Not Found error (Bad internet connection), or 407 Proxy Authentication errors, then before restarting the build, do a cleanup: sbt/sbt clean

    At the end of the process, this should be the output.

    [info] SHA-1: 49a71f3ae33f0ff59e39f94887d8c5315187c6fd
    [info] Packaging /usr/local/spark-1.2.1/examples/target/scala-2.10/spark-examples-1.2.1-hadoop2.6.0.jar ...
    [info] Done packaging.
    [info] Done packaging.
    [success] Total time: 3270 s, completed Jun 26, 2015 12:58:48 AM

11. In hdfs, allow access to the /tmp folder on hdfs: hdfs dfs -chmod -R +777 /tmp
    This would allow access by all to the /tmp folder on hdfs, but in any case, we need to be able to access /tmp for getting to use the hive tables stored on hdfs.

Note: In case the following error appears, there may have been multiple spark-assembly jars built inside Spark. Remove the version that does not correspond to the version of your hadoop (it probably happened because you did not specify the version of hadoop to build spark for).

hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1$ ./bin/pyspark
Python 2.7.3 (default, Dec 18 2014, 19:10:20)

[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
Found multiple Spark assembly jars in /usr/local/spark-1.2.1/assembly/target/scala-2.10:
/usr/local/spark-1.2.1/assembly/target/scala-2.10/spark-assembly-1.2.1-hadoop1.0.4.jar
/usr/local/spark-1.2.1/assembly/target/scala-2.10/spark-assembly-1.2.1-hadoop2.6.0.jar
Please remove all but one jar.

Note: In case the error appears “sbt is corrupt” while building with sbt, get a proper version of the sbt-launch version 0.13.6 (which is required) from here:

http://d29vzk4ow07wi7.cloudfront.net/3fafa9e66dce62a614f5526522e2bbf90f9c9697?response-content-disposition=attachment%3Bfilename%3D%22sbt-launch.jar%22&Policy=eyJTdGF0ZW1lbnQiOiBbeyJSZXNvdXJjZSI6Imh0dHAqOi8vZDI5dnprNG93MDd3aTcuY2xvdWRmcm9udC5uZXQvM2ZhZmE5ZTY2ZGNlNjJhNjE0ZjU1MjY1MjJlMmJiZjkwZjljOTY5Nz9yZXNwb25zZS1jb250ZW50LWRpc3Bvc2l0aW9uPWF0dGFjaG1lbnQlM0JmaWxlbmFtZSUzRCUyMnNidC1sYXVuY2guamFyJTIyIiwiQ29uZGl0aW9uIjp7IkRhdGVMZXNzVGhhbiI6eyJBV1M6RXBvY2hUaW1lIjoxNDM0OTcxMTA2fSwiSXBBZGRyZXNzIjp7IkFXUzpTb3VyY2VJcCI6IjAuMC4wLjAvMCJ9fX1dfQ__&Signature=GWy7YzxYfS5bUi0x8cbcgnsqyq9QCi03pBtB6HPrXf7bnCtrRIliGFudH4lCMvHATKUMpu0yJHNo49xB55y5gIkDwiKpWJfBSgi9~F-f29~FzUlhMNXQQhPLYaHTFrTNWCsMZs8wxCt35BLVLNqZAuKVf6BUkz14a6cJpLedjL0M-kyFmtn5wizFeDRyAF9SeV9WwHHjTbvDZs4dUpnuAJgUwmxNvOP6-fxpeInlKLOls7WLH-krozJWLTpSJihhfoj38CtuKViSlT9toKxBaMvlP~NSGENWS1FJlOeKe1MWvwst6JX8rgRIrOqC1Zb0Miy4aDmdn99nIq6SOjzMFg__&Key-Pair-Id=APKAIFKFWOMXM2UMTSFA

12. Now, it's time to fire up pyspark, or run sample scripts in sprak-submit. The following are the errors that indicate a fault with the installation.

Note: In case of the following error:

hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1/bin$ ./spark-submit ~/sampleHiveSpark.py Spark assembly has been built with Hive, including Datanucleus jars on classpath
Traceback (most recent call last):
  File "/home/hduser/sampleHiveSpark.py", line 9, in <module>
    sqlContext.sql("CREATE TABLE IF NOT EXISTS src (key INT, value STRING)")
  File "/usr/local/spark-1.2.1/python/pyspark/sql.py", line 1620, in sql
    return SchemaRDD(self._ssql_ctx.sql(sqlQuery).toJavaSchemaRDD(), self)
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/java_gateway.py", line 538, in __call__
  File "/usr/local/spark-1.2.1/python/lib/py4j-0.8.2.1-src.zip/py4j/protocol.py", line 300, in get_return_value
py4j.protocol.Py4JJavaError: An error occurred while calling o19.sql.
: java.lang.RuntimeException: java.lang.RuntimeException: Unable to instantiate org.apache.hadoop.hive.metastore.HiveMetaStoreClient
        at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:346)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:235)
        at org.apache.spark.sql.hive.HiveContext$$anonfun$4.apply(HiveContext.scala:231)
        at scala.Option.orElse(Option.scala:257)
        at org.apache.spark.sql.hive.HiveContext.x3lzycompute(HiveContext.scala:231)
        at org.apache.spark.sql.hive.HiveContext.x$3(HiveContext.scala:229)

```
	at org.apache.spark.sql.hive.HiveContext.hiveconf$lzycompute(HiveContext.scala:229)
	at org.apache.spark.sql.hive.HiveContext.hiveconf(HiveContext.scala:229)
	at org.apache.spark.sql.hive.HiveMetastoreCatalog.<init>(HiveMetastoreCatalog.scala:55)
	at org.apache.spark.sql.hive.HiveContext$$anon$2.<init>(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext.catalog$lzycompute(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext.catalog(HiveContext.scala:253)
	at org.apache.spark.sql.hive.HiveContext$$anon$4.<init>(HiveContext.scala:263)
	at org.apache.spark.sql.hive.HiveContext.analyzer$lzycompute(HiveContext.scala:263)
	at org.apache.spark.sql.hive.HiveContext.analyzer(HiveContext.scala:262)
	at
org.apache.spark.sql.SQLContext$QueryExecution.analyzed$lzycompute(SQLContext.scala:411)
	at org.apache.spark.sql.SQLContext$QueryExecution.analyzed(SQLContext.scala:411)
	at org.apache.spark.sql.SchemaRDDLike$class.$init$(SchemaRDDLike.scala:58)
	at org.apache.spark.sql.SchemaRDD.<init>(SchemaRDD.scala:108)
	at org.apache.spark.sql.hive.HiveContext.sql(HiveContext.scala:94)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:606)
	at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:231)
	at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.java:379)
	at py4j.Gateway.invoke(Gateway.java:259)
	at py4j.commands.AbstractCommand.invokeMethod(AbstractCommand.java:133)
	at py4j.commands.CallCommand.execute(CallCommand.java:79)
	at py4j.GatewayConnection.run(GatewayConnection.java:207)
	at java.lang.Thread.run(Thread.java:745)
Caused by: java.lang.RuntimeException: Unable to instantiate
org.apache.hadoop.hive.metastore.HiveMetaStoreClient
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1412)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:62)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:72
)
	at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:2453)
	at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:2465)
	at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:340)
	... 30 more
Caused by: java.lang.reflect.InvocationTargetException
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1410)
	... 35 more
Caused by: javax.jdo.JDOFatalInternalException: Error creating transactional connection factory
```

NestedThrowables:
java.lang.reflect.InvocationTargetException
	at
org.datanucleus.api.jdo.NucleusJDOHelper.getJDOExceptionForNucleusException(NucleusJDOHelper
.java:587)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:788)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.createPersistenceManagerFactory(JDOPersiste
nceManagerFactory.java:333)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.getPersistenceManagerFactory(JDOPersistenc
eManagerFactory.java:202)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:606)
	at javax.jdo.JDOHelper$16.run(JDOHelper.java:1965)
	at java.security.AccessController.doPrivileged(Native Method)
	at javax.jdo.JDOHelper.invoke(JDOHelper.java:1960)
	at
javax.jdo.JDOHelper.invokeGetPersistenceManagerFactoryOnImplementation(JDOHelper.java:1166)
	at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:808)
	at javax.jdo.JDOHelper.getPersistenceManagerFactory(JDOHelper.java:701)
	at org.apache.hadoop.hive.metastore.ObjectStore.getPMF(ObjectStore.java:310)
	at org.apache.hadoop.hive.metastore.ObjectStore.getPersistenceManager(ObjectStore.java:339)
	at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:248)
	at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:223)
	at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
	at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:58)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:67)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:49
7)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:475)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.java:
523)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:397)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.<init>(HiveMetaStore.java:356)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:54)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:59)

```
	at
org.apache.hadoop.hive.metastore.HiveMetaStore.newHMSHandler(HiveMetaStore.java:4944)
	at
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:171)
	... 40 more
Caused by: java.lang.reflect.InvocationTargetException
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
	at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:325)
	at
org.datanucleus.store.AbstractStoreManager.registerConnectionFactory(AbstractStoreManager.java:28
2)
	at org.datanucleus.store.AbstractStoreManager.<init>(AbstractStoreManager.java:240)
	at org.datanucleus.store.rdbms.RDBMSStoreManager.<init>(RDBMSStoreManager.java:286)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
	at
org.datanucleus.plugin.NonManagedPluginRegistry.createExecutableExtension(NonManagedPluginRe
gistry.java:631)
	at org.datanucleus.plugin.PluginManager.createExecutableExtension(PluginManager.java:301)
	at
org.datanucleus.NucleusContext.createStoreManagerForProperties(NucleusContext.java:1187)
	at org.datanucleus.NucleusContext.initialise(NucleusContext.java:356)
	at
org.datanucleus.api.jdo.JDOPersistenceManagerFactory.freezeConfiguration(JDOPersistenceManagerF
actory.java:775)
	... 69 more
Caused by: org.datanucleus.exceptions.NucleusException: Attempt to invoke the "BONECP" plugin to
create a ConnectionPool gave an error : The specified datastore driver ("com.mysql.jdbc.Driver") was
not found in the CLASSPATH. Please check your CLASSPATH specification, and the name of the
driver.
	at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java
:259)
	at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.initialiseDataSources(ConnectionFactoryImpl.jav
```

a:131)
	at org.datanucleus.store.rdbms.ConnectionFactoryImpl.<init>(ConnectionFactoryImpl.java:85)
	... 87 more
Caused by: org.datanucleus.store.rdbms.connectionpool.DatastoreDriverNotFoundException: The specified datastore driver ("com.mysql.jdbc.Driver") was not found in the CLASSPATH. Please check your CLASSPATH specification, and the name of the driver.
	at
org.datanucleus.store.rdbms.connectionpool.AbstractConnectionPoolFactory.loadDriver(AbstractConnectionPoolFactory.java:58)
	at
org.datanucleus.store.rdbms.connectionpool.BoneCPConnectionPoolFactory.createConnectionPool(BoneCPConnectionPoolFactory.java:54)
	at
org.datanucleus.store.rdbms.ConnectionFactoryImpl.generateDataSources(ConnectionFactoryImpl.java:238)
	... 89 more

Solution: Start the hive metastore server (hive –service metastore &, and enter) , and make sure the hive-site.xml file has proper address of the mysql metastore database.

Note: In case of the following error sequence following starting hive metastore -

hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1/bin$ hive --service metastore &
[1] 26061
hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1/bin$ ls: cannot access /usr/local/spark-1.2.1/lib/spark-assembly-*.jar: No such file or directory
Starting Hive Metastore Server
org.apache.thrift.transport.TTransportException: Could not create ServerSocket on address 0.0.0.0/0.0.0.0:9083.
	at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:109)
	at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:91)
	at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:83)
	at
org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerSocketKeepAlive.java:34)
	at org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaStore.java:5936)
	at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5877)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
	at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:606)
	at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
	at org.apache.hadoop.util.RunJar.main(RunJar.java:136)
Exception in thread "main" org.apache.thrift.transport.TTransportException: Could not create ServerSocket on address 0.0.0.0/0.0.0.0:9083.
	at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:109)
	at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:91)

at org.apache.thrift.transport.TServerSocket.<init>(TServerSocket.java:83)
at
org.apache.hadoop.hive.metastore.TServerSocketKeepAlive.<init>(TServerSocketKeepAlive.java:34)
at org.apache.hadoop.hive.metastore.HiveMetaStore.startMetaStore(HiveMetaStore.java:5936)
at org.apache.hadoop.hive.metastore.HiveMetaStore.main(HiveMetaStore.java:5877)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:606)
at org.apache.hadoop.util.RunJar.run(RunJar.java:221)
at org.apache.hadoop.util.RunJar.main(RunJar.java:136)

Run 'ps -ef' to find the instances of Hive Metastore already running. Then close those instances by killing them, and restart hive metastore).

Note: This error can be ignored, as spark will still work properly.

ls: cannot access /usr/local/spark-1.2.1/lib/spark-assembly-*.jar: No such file or directory

Note: The following command can be used to log whether Hive is accessing the Metastore, etc:

hduser@arinjoy-Inspiron-3521:/usr/local/hive$ hive -hiveconf hive.root.logger=DEBUG,console

The output looks something like this:

ls: cannot access /usr/local/spark-1.2.1/lib/spark-assembly-*.jar: No such file or directory
15/06/26 01:38:35 [main]: WARN common.LogUtils: DEPRECATED: Ignoring hive-default.xml found on the CLASSPATH at /usr/local/hive/conf/hive-default.xml
15/06/26 01:38:35 [main]: DEBUG common.LogUtils: Using hive-site.xml found on CLASSPATH at /usr/local/hive/conf/hive-site.xml

Logging initialized using configuration in file:/usr/local/hive/conf/hive-log4j.properties
15/06/26 01:38:36 [main]: INFO SessionState:
Logging initialized using configuration in file:/usr/local/hive/conf/hive-log4j.properties
15/06/26 01:38:36 [main]: DEBUG parse.VariableSubstitution: Substitution is on: hive
15/06/26 01:38:36 [main]: DEBUG lib.MutableMetricsFactory: field
org.apache.hadoop.metrics2.lib.MutableRate
org.apache.hadoop.security.UserGroupInformation$UgiMetrics.loginSuccess with annotation
@org.apache.hadoop.metrics2.annotation.Metric(value=[Rate of successful kerberos logins and latency (milliseconds)], about=, valueName=Time, type=DEFAULT, always=false, sampleName=Ops)
15/06/26 01:38:36 [main]: DEBUG lib.MutableMetricsFactory: field
org.apache.hadoop.metrics2.lib.MutableRate
org.apache.hadoop.security.UserGroupInformation$UgiMetrics.loginFailure with annotation
@org.apache.hadoop.metrics2.annotation.Metric(value=[Rate of failed kerberos logins and latency (milliseconds)], about=, valueName=Time, type=DEFAULT, always=false, sampleName=Ops)
15/06/26 01:38:36 [main]: DEBUG lib.MutableMetricsFactory: field
org.apache.hadoop.metrics2.lib.MutableRate

org.apache.hadoop.security.UserGroupInformation$UgiMetrics.getGroups with annotation @org.apache.hadoop.metrics2.annotation.Metric(value=[GetGroups], about=, valueName=Time, type=DEFAULT, always=false, sampleName=Ops)
15/06/26 01:38:36 [main]: DEBUG impl.MetricsSystemImpl: UgiMetrics, User and group related metrics
15/06/26 01:38:36 [main]: DEBUG util.KerberosName: Kerberos krb5 configuration not found, setting default realm to empty
15/06/26 01:38:36 [main]: DEBUG security.Groups: Creating new Groups object
15/06/26 01:38:36 [main]: DEBUG util.NativeCodeLoader: Trying to load the custom-built native-hadoop library...
15/06/26 01:38:36 [main]: DEBUG util.NativeCodeLoader: Loaded the native-hadoop library
15/06/26 01:38:36 [main]: DEBUG security.JniBasedUnixGroupsMapping: Using JniBasedUnixGroupsMapping for Group resolution
15/06/26 01:38:36 [main]: DEBUG security.JniBasedUnixGroupsMappingWithFallback: Group mapping impl=org.apache.hadoop.security.JniBasedUnixGroupsMapping
15/06/26 01:38:36 [main]: DEBUG security.Groups: Group mapping impl=org.apache.hadoop.security.JniBasedUnixGroupsMappingWithFallback; cacheTimeout=300000; warningDeltaMs=5000
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: hadoop login
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: hadoop login commit
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: using local user:UnixPrincipal: hduser
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: Using user: "UnixPrincipal: hduser" with name hduser
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: User entry: "hduser"
15/06/26 01:38:36 [main]: DEBUG security.UserGroupInformation: UGI loginUser:hduser (auth:SIMPLE)
15/06/26 01:38:36 [main]: INFO hive.metastore: Trying to connect to metastore with URI thrift://127.0.0.1:9083
15/06/26 01:38:36 [main]: WARN hive.metastore: Failed to connect to the MetaStore Server...

Note: If the error states that it is Hive is connecting to derby database, then one way would be to reinstall Hive, and configure it properly to connect it to MySQL server using the previous steps.

Also, remove any other version of Spark that you may have installed before.

Note: In case of the following error:

hduser@arinjoy-Inspiron-3521:/usr/local/spark-1.2.1$ sudo ./bin/spark-submit ~/sampleHiveSpark.py
Loading Spark jar with 'jar' failed.
This is likely because Spark was compiled with Java 7 and run
with Java 6. (see SPARK-1703). Please use Java 7 to run Spark
or build Spark with Java 6.

Get the version of mysql connector in Hive, which should probably be the latest one.

hduser@arinjoy-Inspiron-3521:/usr/local/hive/lib$ sudo cp mysql-connector-java-5.1.16.jar ../../spark-1.2.1/lib_managed/jars/

Sqoop to transfer database table from mysql to hive:

hduser@arinjoy-Inspiron-3521:/usr/local/sqoop/bin$ ./sqoop import --connect jdbc:mysql://localhost:3306/IITBxDataAnalytics --username root --password " --table UserSessionOldLog -m 1 --hive-import --hive-table default.UserLog --warehouse-dir /user/hive/warehouse

Warning: /usr/local/sqoop/../hbase does not exist! HBase imports will fail.
Please set $HBASE_HOME to the root of your HBase installation.
Warning: /usr/local/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/local/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /usr/local/sqoop/../zookeeper does not exist! Accumulo imports will fail.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
15/06/26 16:20:20 INFO sqoop.Sqoop: Running Sqoop version: 1.4.5
15/06/26 16:20:20 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
15/06/26 16:20:20 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can override
15/06/26 16:20:20 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
15/06/26 16:20:20 WARN tool.BaseSqoopTool: It seems that you're doing hive import directly into default
15/06/26 16:20:20 WARN tool.BaseSqoopTool: hive warehouse directory which is not supported. Sqoop is
15/06/26 16:20:20 WARN tool.BaseSqoopTool: firstly importing data into separate directory and then
15/06/26 16:20:20 WARN tool.BaseSqoopTool: inserting data into hive. Please consider removing
15/06/26 16:20:20 WARN tool.BaseSqoopTool: --target-dir or --warehouse-dir into /user/hive/warehouse in
15/06/26 16:20:20 WARN tool.BaseSqoopTool: case that you will detect any issues.
15/06/26 16:20:20 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
15/06/26 16:20:20 INFO tool.CodeGenTool: Beginning code generation
15/06/26 16:20:21 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `UserSessionOldLog` AS t LIMIT 1
15/06/26 16:20:21 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `UserSessionOldLog` AS t LIMIT 1
15/06/26 16:20:21 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop
Note: /tmp/sqoop-hduser/compile/e359e45d830b65f492367ecd99a756dc/UserSessionOldLog.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
15/06/26 16:20:26 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-hduser/compile/e359e45d830b65f492367ecd99a756dc/UserSessionOldLog.jar
15/06/26 16:20:26 WARN manager.MySQLManager: It looks like you are importing from mysql.
15/06/26 16:20:26 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
15/06/26 16:20:26 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
15/06/26 16:20:26 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
15/06/26 16:20:26 INFO mapreduce.ImportJobBase: Beginning import of UserSessionOldLog

15/06/26 16:20:27 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
15/06/26 16:20:28 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
15/06/26 16:20:28 INFO client.RMProxy: Connecting to ResourceManager at /0.0.0.0:8032
15/06/26 16:20:29 INFO mapreduce.JobSubmissionFiles: Permissions on staging directory /tmp/hadoop-yarn/staging/hduser/.staging are incorrect: rwxrwxrwx. Fixing permissions to correct value rwx------
15/06/26 16:20:31 INFO db.DBInputFormat: Using read commited transaction isolation
15/06/26 16:20:31 INFO mapreduce.JobSubmitter: number of splits:1
15/06/26 16:20:32 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1435299858158_0001
15/06/26 16:20:33 INFO impl.YarnClientImpl: Submitted application application_1435299858158_0001
15/06/26 16:20:33 INFO mapreduce.Job: The url to track the job: http://arinjoy-Inspiron-3521:8088/proxy/application_1435299858158_0001/
15/06/26 16:20:33 INFO mapreduce.Job: Running job: job_1435299858158_0001
15/06/26 16:20:45 INFO mapreduce.Job: Job job_1435299858158_0001 running in uber mode : false
15/06/26 16:20:45 INFO mapreduce.Job: map 0% reduce 0%
15/06/26 16:22:55 INFO mapreduce.Job: map 100% reduce 0%
15/06/26 16:22:57 INFO mapreduce.Job: Job job_1435299858158_0001 completed successfully
15/06/26 16:22:57 INFO mapreduce.Job: Counters: 30
	File System Counters
		FILE: Number of bytes read=0
		FILE: Number of bytes written=115239
		FILE: Number of read operations=0
		FILE: Number of large read operations=0
		FILE: Number of write operations=0
		HDFS: Number of bytes read=87
		HDFS: Number of bytes written=1568072086
		HDFS: Number of read operations=4
		HDFS: Number of large read operations=0
		HDFS: Number of write operations=2
	Job Counters
		Launched map tasks=1
		Other local map tasks=1
		Total time spent by all maps in occupied slots (ms)=126591
		Total time spent by all reduces in occupied slots (ms)=0
		Total time spent by all map tasks (ms)=126591
		Total vcore-seconds taken by all map tasks=126591
		Total megabyte-seconds taken by all map tasks=129629184
	Map-Reduce Framework
		Map input records=3830879
		Map output records=3830879
		Input split bytes=87
		Spilled Records=0
		Failed Shuffles=0
		Merged Map outputs=0
		GC time elapsed (ms)=1427

```
                CPU time spent (ms)=93700
                Physical memory (bytes) snapshot=203685888
                Virtual memory (bytes) snapshot=1161834496
                Total committed heap usage (bytes)=110624768
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=1568072086
15/06/26 16:22:57 INFO mapreduce.ImportJobBase: Transferred 1.4604 GB in 148.8832 seconds
(10.0443 MB/sec)
15/06/26 16:22:57 INFO mapreduce.ImportJobBase: Retrieved 3830879 records.
15/06/26 16:22:57 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM
`UserSessionOldLog` AS t LIMIT 1
15/06/26 16:22:57 WARN hive.TableDefWriter: Column createDateTime had to be cast to a less
precise type in Hive
15/06/26 16:22:57 WARN hive.TableDefWriter: Column problemSubmissionTime had to be cast to a
less precise type in Hive
15/06/26 16:22:57 WARN hive.TableDefWriter: Column lastModDateTime had to be cast to a less
precise type in Hive
15/06/26 16:22:57 INFO hive.HiveImport: Loading uploaded data into Hive
15/06/26 16:22:58 INFO hive.HiveImport: ls: cannot access /usr/local/spark-1.2.1/lib/spark-assembly-
*.jar: No such file or directory
15/06/26 16:23:07 INFO hive.HiveImport:
15/06/26 16:23:07 INFO hive.HiveImport: Logging initialized using configuration in
file:/usr/local/hive/conf/hive-log4j.properties
15/06/26 16:23:16 INFO hive.HiveImport: OK
15/06/26 16:23:16 INFO hive.HiveImport: Time taken: 4.103 seconds
15/06/26 16:23:17 INFO hive.HiveImport: Loading data to table default.userlog
15/06/26 16:23:18 INFO hive.HiveImport: Table default.userlog stats: [numFiles=1,
totalSize=1568072086]
15/06/26 16:23:18 INFO hive.HiveImport: OK
15/06/26 16:23:18 INFO hive.HiveImport: Time taken: 1.664 seconds
15/06/26 16:23:19 INFO hive.HiveImport: Hive import complete.
15/06/26 16:23:19 INFO hive.HiveImport: Export directory is not empty, keeping it.
```

Note: This following error -

```
15/06/26 16:14:11 INFO sqoop.Sqoop: Running Sqoop version: 1.4.5
15/06/26 16:14:11 WARN tool.BaseSqoopTool: Setting your password on the command-line is
insecure. Consider using -P instead.
15/06/26 16:14:11 INFO tool.BaseSqoopTool: Using Hive-specific delimiters for output. You can
override
15/06/26 16:14:11 INFO tool.BaseSqoopTool: delimiters with --fields-terminated-by, etc.
15/06/26 16:14:11 ERROR tool.BaseSqoopTool: Got error creating database manager:
java.io.IOException: No manager for connect string: jdbc:mysql//localhost:3306/IITBxDataAnalytics
        at org.apache.sqoop.ConnFactory.getManager(ConnFactory.java:191)
        at org.apache.sqoop.tool.BaseSqoopTool.init(BaseSqoopTool.java:247)
        at org.apache.sqoop.tool.ImportTool.init(ImportTool.java:89)
```

```
        at org.apache.sqoop.tool.ImportTool.run(ImportTool.java:589)
        at org.apache.sqoop.Sqoop.run(Sqoop.java:143)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)
        at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:179)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:218)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:227)
        at org.apache.sqoop.Sqoop.main(Sqoop.java:236)
```

Means the connect string given jdbc:... has an error in the format or spellings.. check it.

Note: In case of this error, please check if hadoop hdfs and yarn is running or not. The best bet would be to execute stop-all.sh and then start-all.sh, or stop-dfs.sh and stop-yarn.sh and start-dfs.sh and stop-yarn.sh in succession, and retrying the command.

```
15/06/28 00:41:33 ERROR tool.ImportTool: Encountered IOException running import job:
java.net.ConnectException: Call From arinjoy-Inspiron-3521/127.0.1.1 to localhost:54310 failed on
connection exception: java.net.ConnectException: Connection refused; For more details see:
http://wiki.apache.org/hadoop/ConnectionRefused
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:57)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:
45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:526)
        at org.apache.hadoop.net.NetUtils.wrapWithMessage(NetUtils.java:791)
        at org.apache.hadoop.net.NetUtils.wrapException(NetUtils.java:731)
        at org.apache.hadoop.ipc.Client.call(Client.java:1472)
        at org.apache.hadoop.ipc.Client.call(Client.java:1399)
        at org.apache.hadoop.ipc.ProtobufRpcEngine$Invoker.invoke(ProtobufRpcEngine.java:232)
        at com.sun.proxy.$Proxy9.getFileInfo(Unknown Source)
        at
org.apache.hadoop.hdfs.protocolPB.ClientNamenodeProtocolTranslatorPB.getFileInfo(ClientNamenod
eProtocolTranslatorPB.java:752)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:57)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:606)
        at
org.apache.hadoop.io.retry.RetryInvocationHandler.invokeMethod(RetryInvocationHandler.java:187)
        at org.apache.hadoop.io.retry.RetryInvocationHandler.invoke(RetryInvocationHandler.java:102)
        at com.sun.proxy.$Proxy10.getFileInfo(Unknown Source)
        at org.apache.hadoop.hdfs.DFSClient.getFileInfo(DFSClient.java:1988)
        at org.apache.hadoop.hdfs.DistributedFileSystem$18.doCall(DistributedFileSystem.java:1118)
        at org.apache.hadoop.hdfs.DistributedFileSystem$18.doCall(DistributedFileSystem.java:1114)
```

```
        at org.apache.hadoop.fs.FileSystemLinkResolver.resolve(FileSystemLinkResolver.java:81)
        at
org.apache.hadoop.hdfs.DistributedFileSystem.getFileStatus(DistributedFileSystem.java:1114)
        at org.apache.hadoop.fs.FileSystem.exists(FileSystem.java:1400)
        at
org.apache.hadoop.mapreduce.lib.output.FileOutputFormat.checkOutputSpecs(FileOutputFormat.java:
145)
        at org.apache.hadoop.mapreduce.JobSubmitter.checkSpecs(JobSubmitter.java:562)
        at org.apache.hadoop.mapreduce.JobSubmitter.submitJobInternal(JobSubmitter.java:432)
        at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1296)
        at org.apache.hadoop.mapreduce.Job$10.run(Job.java:1293)
        at java.security.AccessController.doPrivileged(Native Method)
        at javax.security.auth.Subject.doAs(Subject.java:415)
        at org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1628)
        at org.apache.hadoop.mapreduce.Job.submit(Job.java:1293)
        at org.apache.hadoop.mapreduce.Job.waitForCompletion(Job.java:1314)
        at org.apache.sqoop.mapreduce.ImportJobBase.doSubmitJob(ImportJobBase.java:186)
        at org.apache.sqoop.mapreduce.ImportJobBase.runJob(ImportJobBase.java:159)
        at org.apache.sqoop.mapreduce.ImportJobBase.runImport(ImportJobBase.java:247)
        at org.apache.sqoop.manager.SqlManager.importTable(SqlManager.java:665)
        at org.apache.sqoop.manager.MySQLManager.importTable(MySQLManager.java:118)
        at org.apache.sqoop.tool.ImportTool.importTable(ImportTool.java:497)
        at org.apache.sqoop.tool.ImportTool.run(ImportTool.java:601)
        at org.apache.sqoop.Sqoop.run(Sqoop.java:143)
        at org.apache.hadoop.util.ToolRunner.run(ToolRunner.java:70)
        at org.apache.sqoop.Sqoop.runSqoop(Sqoop.java:179)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:218)
        at org.apache.sqoop.Sqoop.runTool(Sqoop.java:227)
        at org.apache.sqoop.Sqoop.main(Sqoop.java:236)
Caused by: java.net.ConnectException: Connection refused
        at sun.nio.ch.SocketChannelImpl.checkConnect(Native Method)
        at sun.nio.ch.SocketChannelImpl.finishConnect(SocketChannelImpl.java:739)
        at org.apache.hadoop.net.SocketIOWithTimeout.connect(SocketIOWithTimeout.java:206)
        at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:530)
        at org.apache.hadoop.net.NetUtils.connect(NetUtils.java:494)
        at org.apache.hadoop.ipc.Client$Connection.setupConnection(Client.java:607)
        at org.apache.hadoop.ipc.Client$Connection.setupIOstreams(Client.java:705)
        at org.apache.hadoop.ipc.Client$Connection.access$2800(Client.java:368)
        at org.apache.hadoop.ipc.Client.getConnection(Client.java:1521)
        at org.apache.hadoop.ipc.Client.call(Client.java:1438)
        ... 40 more
```