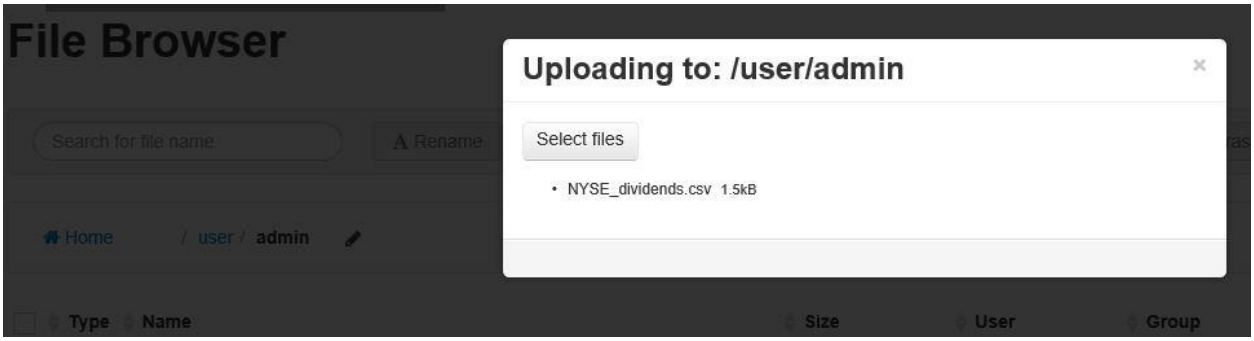
HBASE PROJECT REPORT

Inserting Bulk Data into HBase through PIG Script

Loading Data



HBase Table Creation

Create a HBAse table ‘NYSE’ with Column Family as ‘records’.

```
15/01/25 17:29:27 WARN conf.Configuration: hadoop.native.lib is deprecat
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 0.94.15-cdh4.7.1, rUnknown, Tue Nov 18 08:50:03 PST 2014

hbase(main):001:0> create 'NYSE', 'records'
0 row(s) in 2.2360 seconds

=> Hbase::Table - NYSE
hbase(main):002:0>
```

PIG Programming

Login to PIG Shell

```
login as: root
root@119.9.74.140's password:
Last login: Sun Jan 25 17:29:06 2015 from 59.97.88.37
[root@hadoop-cdh ~]# pig
2015-01-25 17:32:28,090 [main] INFO  org.apache.pig.Main - Apache Pig version 0.
11.0-cdh4.7.1 (rexported) compiled Nov 18 2014, 09:08:23
2015-01-25 17:32:28,091 [main] INFO  org.apache.pig.Main - Logging error message
s to: /root/pig_1422207148084.log
2015-01-25 17:32:28,130 [main] INFO  org.apache.pig.impl.util.Utils - Default bo
otup file /root/.pigbootup not found
2015-01-25 17:32:28,522 [main] WARN  org.apache.hadoop.conf.Configuration - fs.d
efault.name is deprecated. Instead, use fs.defaultFS
2015-01-25 17:32:28,522 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.HExecutionEngine - Connecting to hadoop file system at: hdfs://hadoop-cdh:802
0
2015-01-25 17:32:29,512 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.HExecutionEngine - Connecting to map-reduce job tracker at: hadoop-cdh:8021
2015-01-25 17:32:29,516 [main] WARN  org.apache.hadoop.conf.Configuration - fs.d
efault.name is deprecated. Instead, use fs.defaultFS
grunt>
```

Load Data to PIG

Grunt> A = load '/user/admin/NYSE_dividends.csv' using PigStorage(',') AS (id:int, exchange:chararray, stock_symbol:chararray, dividends:float);

```
grunt> A = load '/user/admin/NYSE_dividends.csv' using PigStorage(',') AS (id:int, exchange:chararray, stock_symbol:chararray, dividends:float);
```

Though not always recommended, but that will be a good practice, if you check the data stored in variable every time, using command DUMP <variable name>;

Constructing MAP

Use TOMAP function to construct Map which will be stored in Column Family “records”

Grunt> B = foreach A generate id, TOMAP('exchange', exchange, 'stock_symbol', stock_symbol, 'dividends', dividends);

```
grunt>
grunt> B = foreach A generate id, TOMAP('exchange', exchange, 'stock_symbol', stock_symbol, 'date', date, 'dividends', dividends);
grunt>
```

Store data in HBase

Grunt> store B into 'hbase://NYSE' using org.apache.pig.backend.hadoop.hbase.HBaseStorage('records');

```
grunt> store B into 'hbase://NYSE' using org.apache.pig.backend.hadoop.hbase.HBaseStorage('records');
```

If everything goes fine, you will get "Success" Method like below:-

```
HadoopVersion   PigVersion      UserId  StartedAt       FinishedAt      Features
2.0.0-cdh4.7.1  0.11.0-cdh4.7.1 root    2015-01-25 17:46:40     2015-01-25 17:47:05     UNKNOWN

Success!

Job Stats (time in seconds):
JobId   Maps    Reduces MaxMapTime      MinMapTIme      AvgMapTime      MedianMapTime   MaxReduceTime   MinReduceTime   AvgReduceTime   MedianReducetime        Alias
eature  Outputs
job_201501251000_0037   1       0       6       6       6       6       0       0       0       0       A,B     MAP_ONLY        hbase://cricket,

Input(s):
Successfully read 85 records (1920 bytes) from: "/user/admin/NYSE_dividends.csv"

Output(s):
Successfully stored 85 records in: "hbase://cricket"

Counters:
Total records written : 85
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_201501251000_0037


2015-01-25 17:47:05,711 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Encountered Warning ACCESSING_NON_EXISTENT_FIELD
5 time(s).
2015-01-25 17:47:05,711 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
grunt>
```

Validation

Go to Hbase Console and scan the table

```
hbase(main):001:0> create 'NYSE', 'records'
0 row(s) in 2.2360 seconds

=> Hbase::Table - NYSE
hbase(main):002:0> scan 'NYSE'
```

.

Self-Study

Once you are done with the above project, you are required to complete self study, by manually creating the table and enter the data as mentioned in the video recording of the session.

You are supposed to realize the below architecture:-