Hadoop Based Applications

Apache Pig

- ***Overview of apps, high level languages, services***
- ***Databases/Stores***
- ***Querying***
- ***Machine Learning***
- ***Graph Processing***

Databases/Stores

- ***Avro:*** data structures within context of Hadoop MapReduce jobs.
- ***Hbase:*** distributed non-relational database
- ***Cassandra:*** distributed data management system

Querying

- ***Pig :*** Platform for analyzing large data sets in HDFS
- ***Hive :*** Query and manage large datasets
- ***Impala :*** High-performance, low-latency SQL querying of data in Hadoop file formats
- ***Spark :*** General processing engine for streaming, SQL, machine learning and graph processing.

Machine Learning, Graph Processing

- ***Giraph:*** Iterative graph processing using Hadoop framework
- ***Mahout:*** Framework for machine learning applications using Hadoop, Spark
- ***Spark:*** General processing engine for streaming, SQL, machine learning and graph processing.

Apache Pig

- *Pig components – PigLatin, and infrastructure layer*
- *Typical Pig use cases*
- *Run Pig with Hadoop integration.*

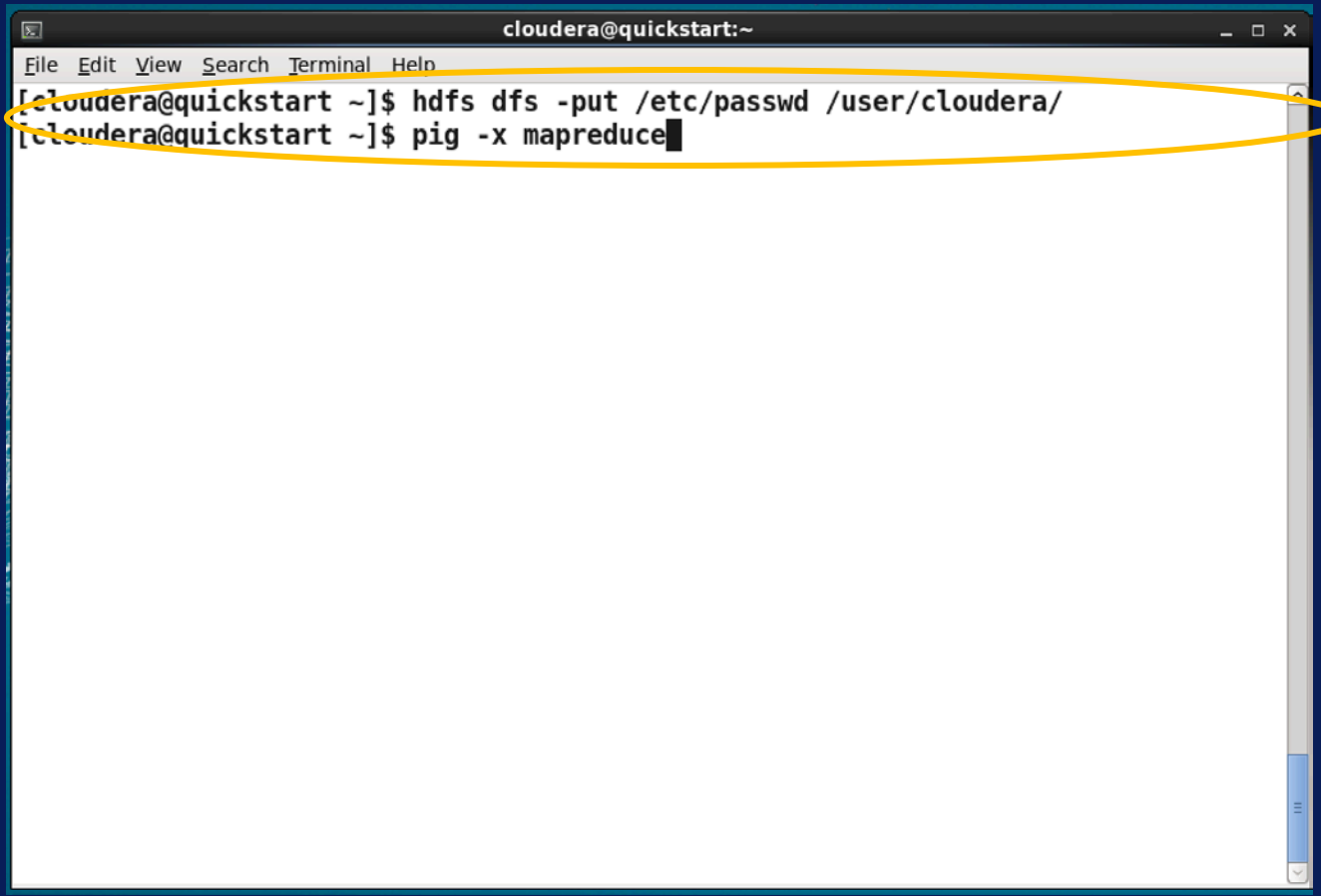
Apache Pig

- ***Platform for data processing***
- ***Pig Latin:*** ***High level language***
- ***Pig execution environment:*** ***Local, MapReduce, Tez***
- ***In built operators and functions***
- ***Extensible***

Pig Usage Areas

- ***Extract, Transform, Load (ETL) operations***
- ***Manipulating, analyzing “raw” data***
- ***Widely used, extensive list at:***
  https://cwiki.apache.org/confluence/display/PIG/PoweredBy

Pig Example

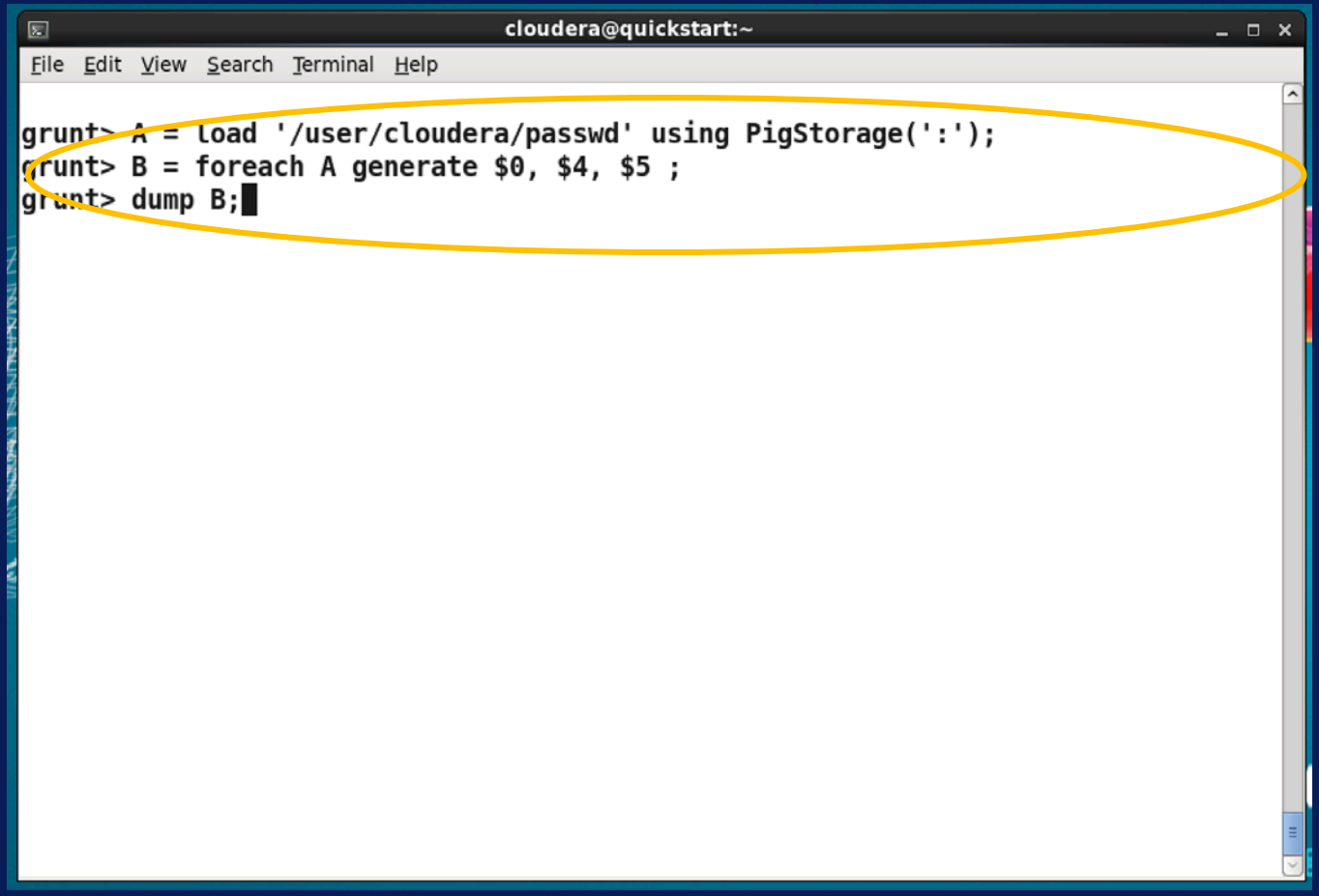


- ***Load passwd file and work with data.***
- ***Step 1:***
  hdfs dfs -put /etc/passwd /user/cloudera

(Note: this is a single line)

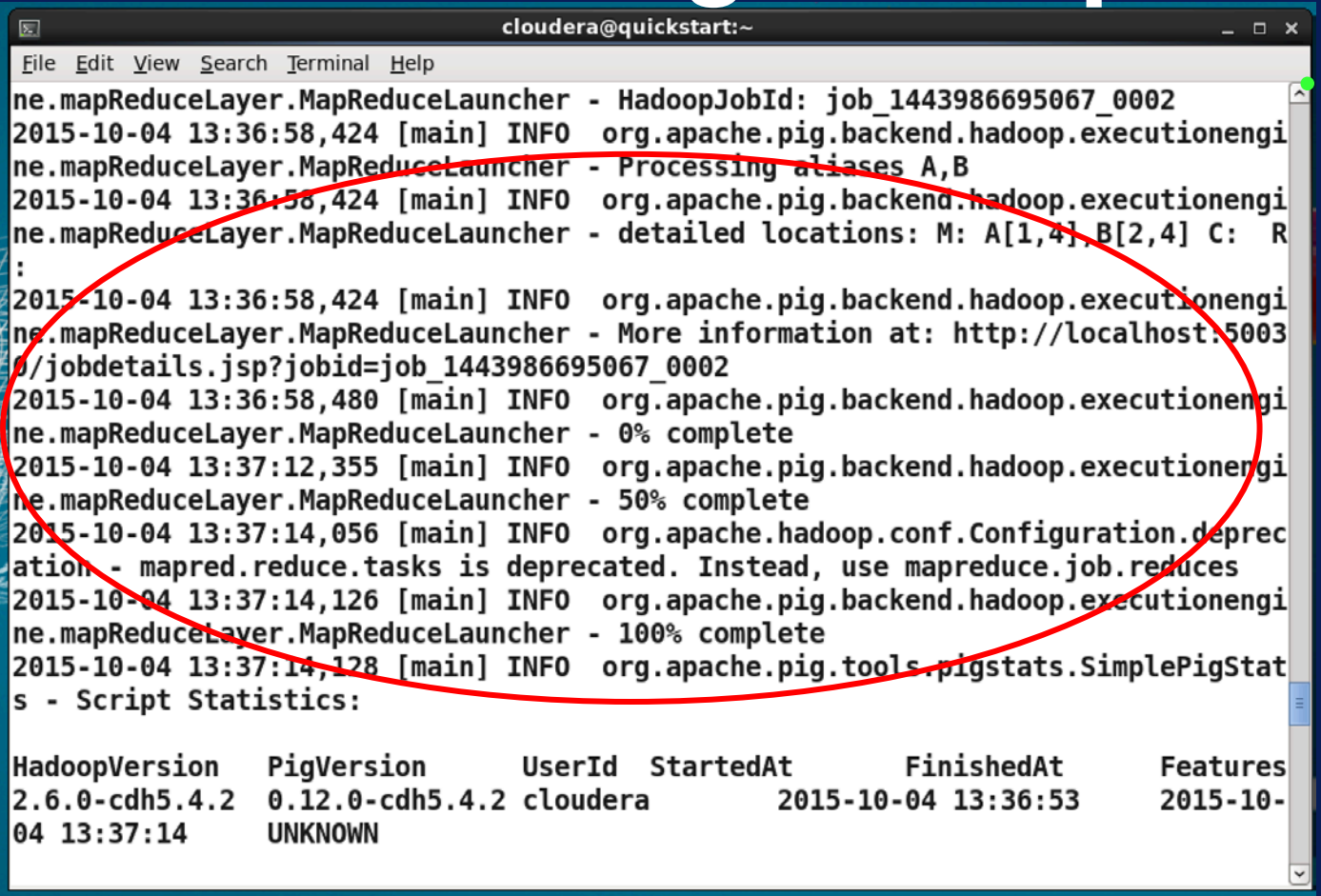
- ***Step 2:***
  pig -x mapreduce

Pig Example



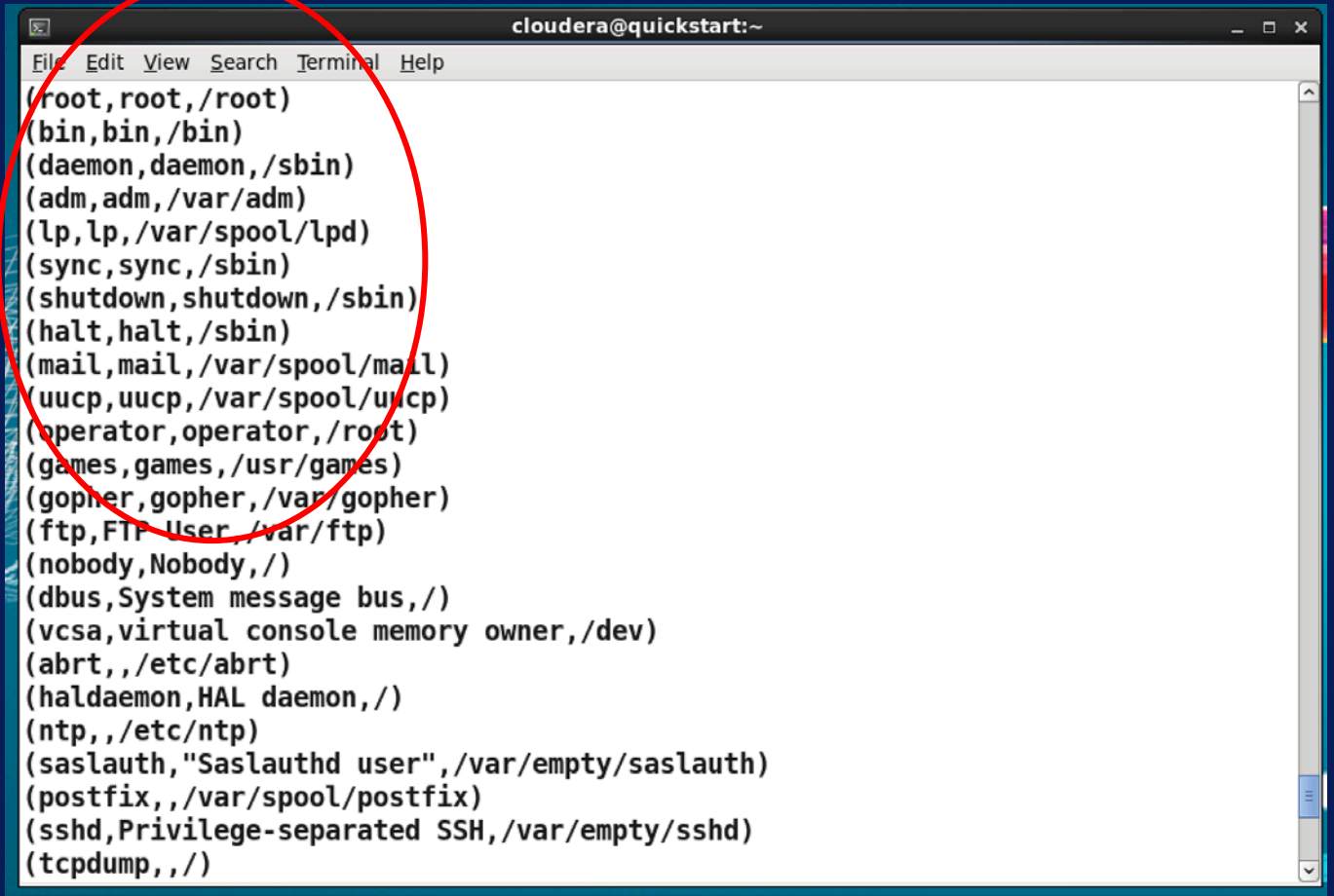
- ***Puts you in "grunt" shell.***
- ***clear***
- ***Load the file:***
  load '/user/cloudera/passwd' using PigStorage(':');
- ***Pick subset of values:***
  B = foreach A generate $0, $4, $5;
  dump B;

Pig Example



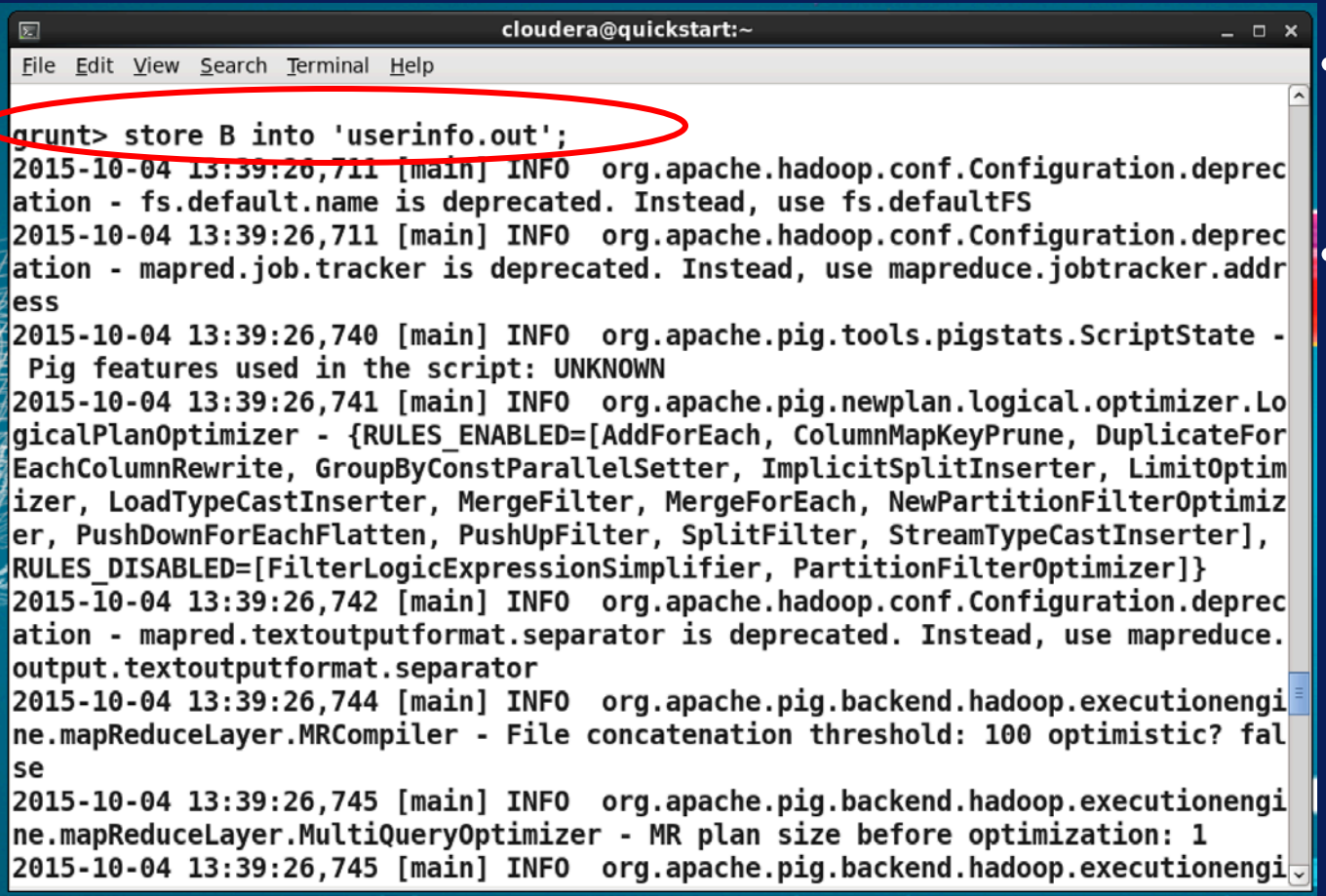
Backend Hadoop job info.

Pig Example



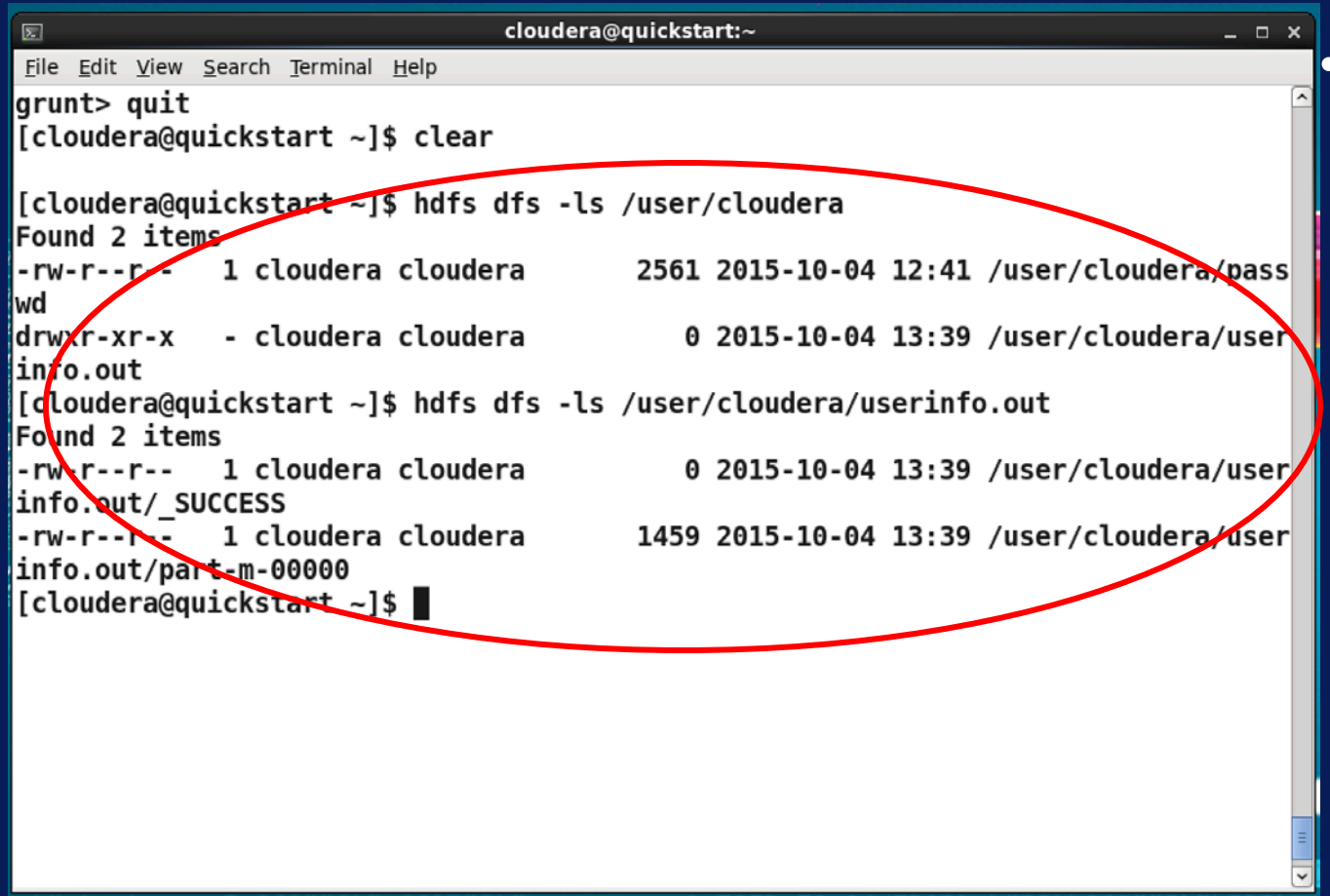
- ***Outputs username, full name, and home directory path.***

Pig Example



- ***Can store this processed data in HDFS***
- ***Command:***
  store B into 'userinfo.out';

Pig Example



- ***Verify the new data is in HDFS.***

Summary

- ***Used interactive shell for Pig example***
- ***Can also run using scripts***
- ***Also as embedded programs in a host language (Java for example).***

Apache Hive

- *Query and manage data using HiveQL*
- *Run interactively using beeline.*
- *Other run mechanisms*

Apache Hive

- ***Data warehouse software***
- ***HiveQL:*** ***SQL like language to structure, and query data***
- ***Execution environment:*** ***MapReduce, Tez, Spark***
- ***Data in HDFS, HBase***
- ***Custom mappers/reducers***

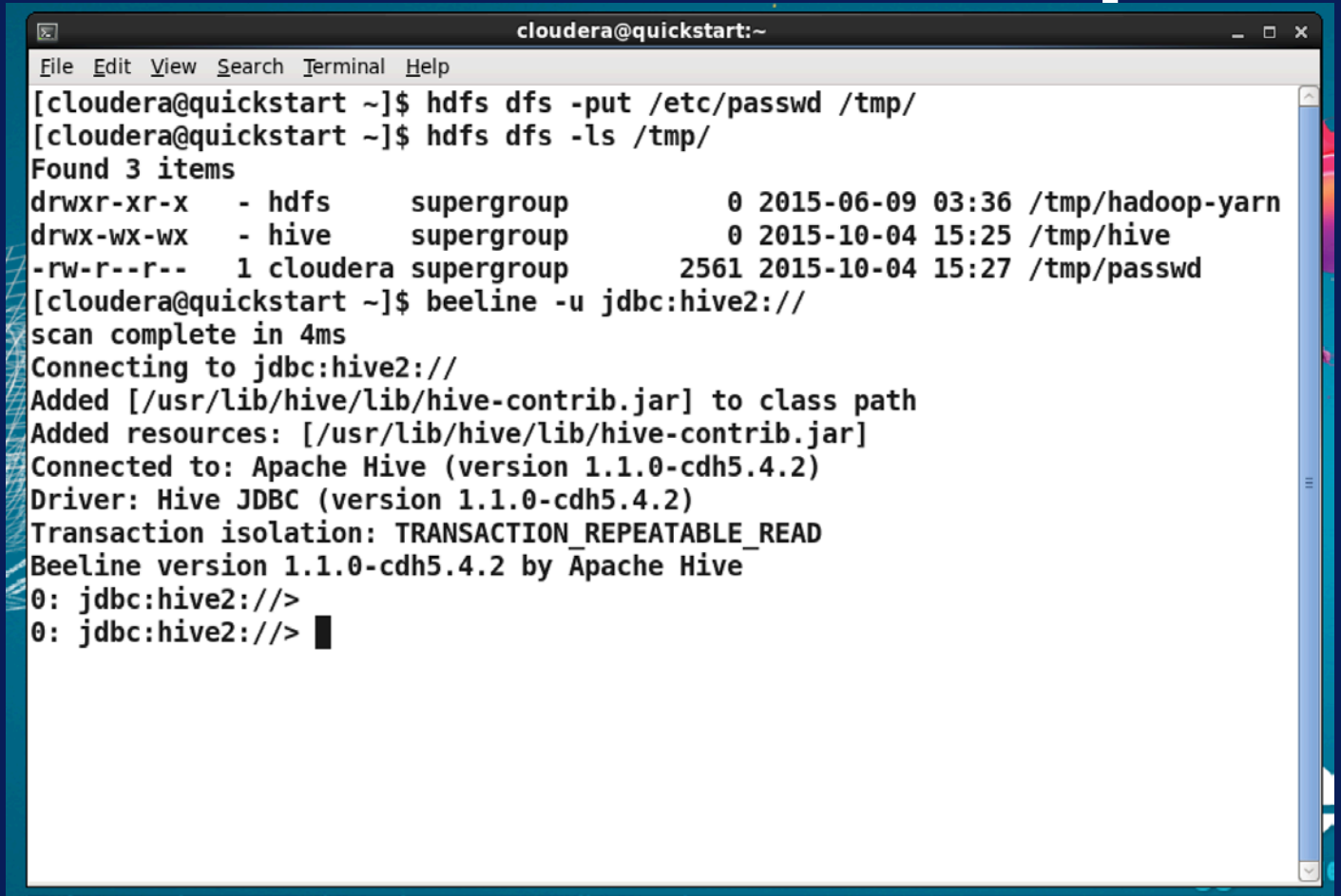
Hive Usage Areas

- ***Data mining, analytics***
- ***Machine learning***
- ***Ad hoc analysis***
- ***Widely used, extensive list at:***
  https://cwiki.apache.org/confluence/display/Hive/PoweredBy

Hive Example

- *Revisit /etc/passwd file example from Pig lesson video.*
- *Start by loading file into HDFS:*
  hdfs dfs -put /etc/passwd /tmp/
- *Run beeline to access interactively:*
  beeline -u jdbc:hive2://

Hive Example

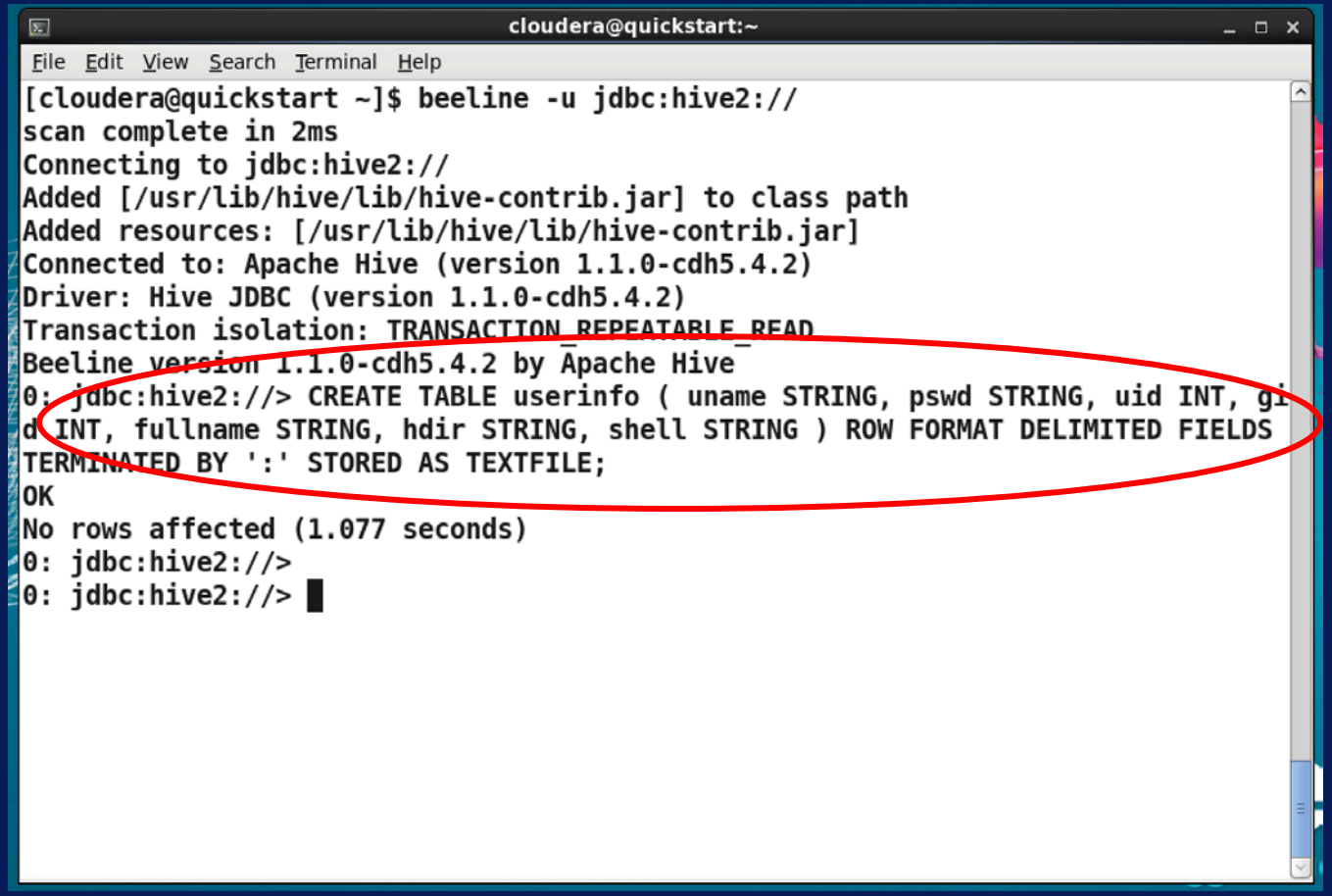


- *Copy passwd file to HDFS*
- *Running interactively usiing beeline.*

Hive Example: Command list

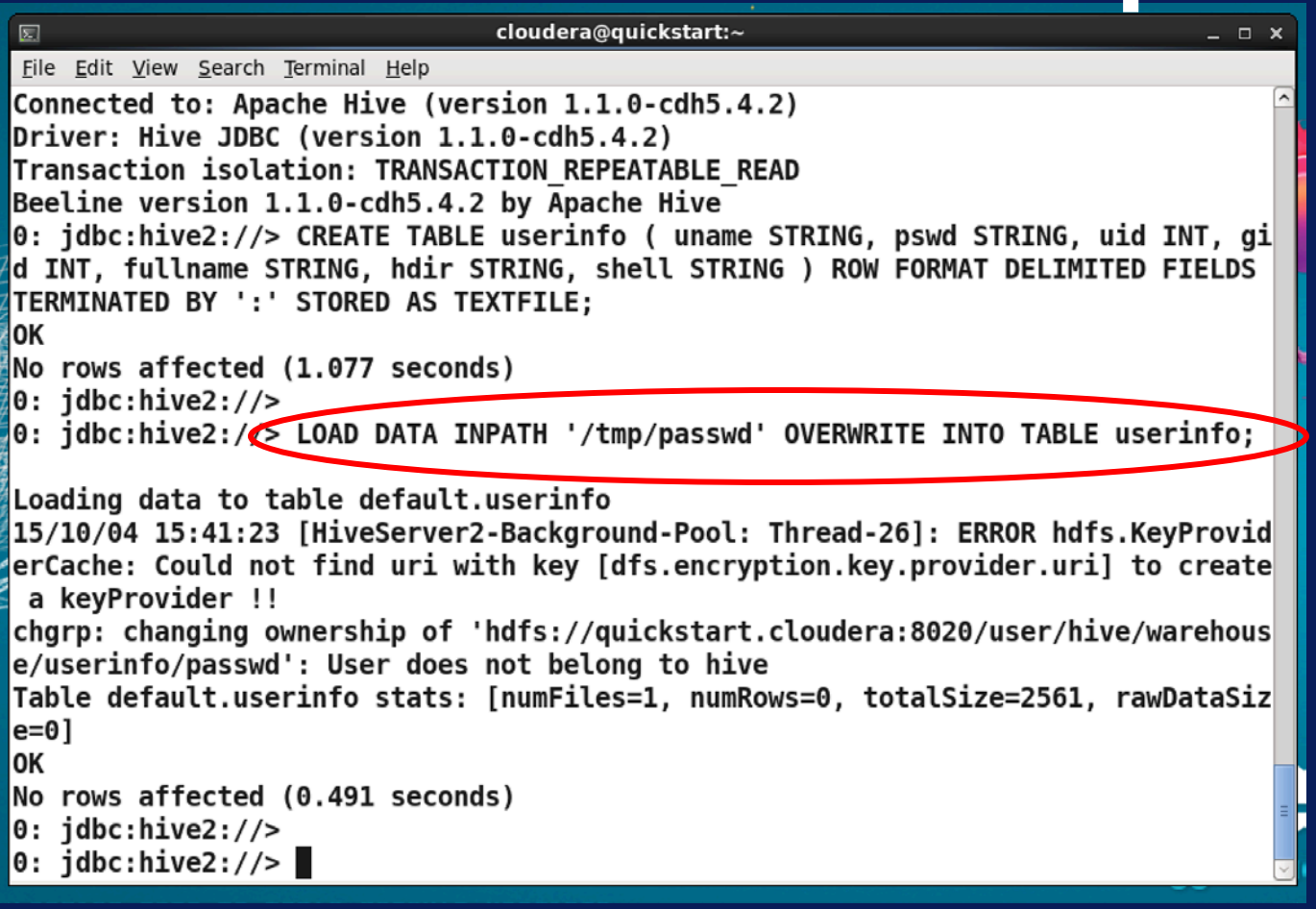
```
CREATE TABLE userinfo ( uname STRING, pswd STRING, uid INT, gid INT, fullname STRING, hdir STRING, shell STRING ) ROW FORMAT DELIMITED FIELDS TERMINATED BY ':' STORED AS TEXTFILE;

LOAD DATA INPATH '/tmp/passwd' OVERWRITE INTO TABLE userinfo;

SELECT uname, fullname, hdir FROM userinfo ORDER BY uname ;
```

Hive Example



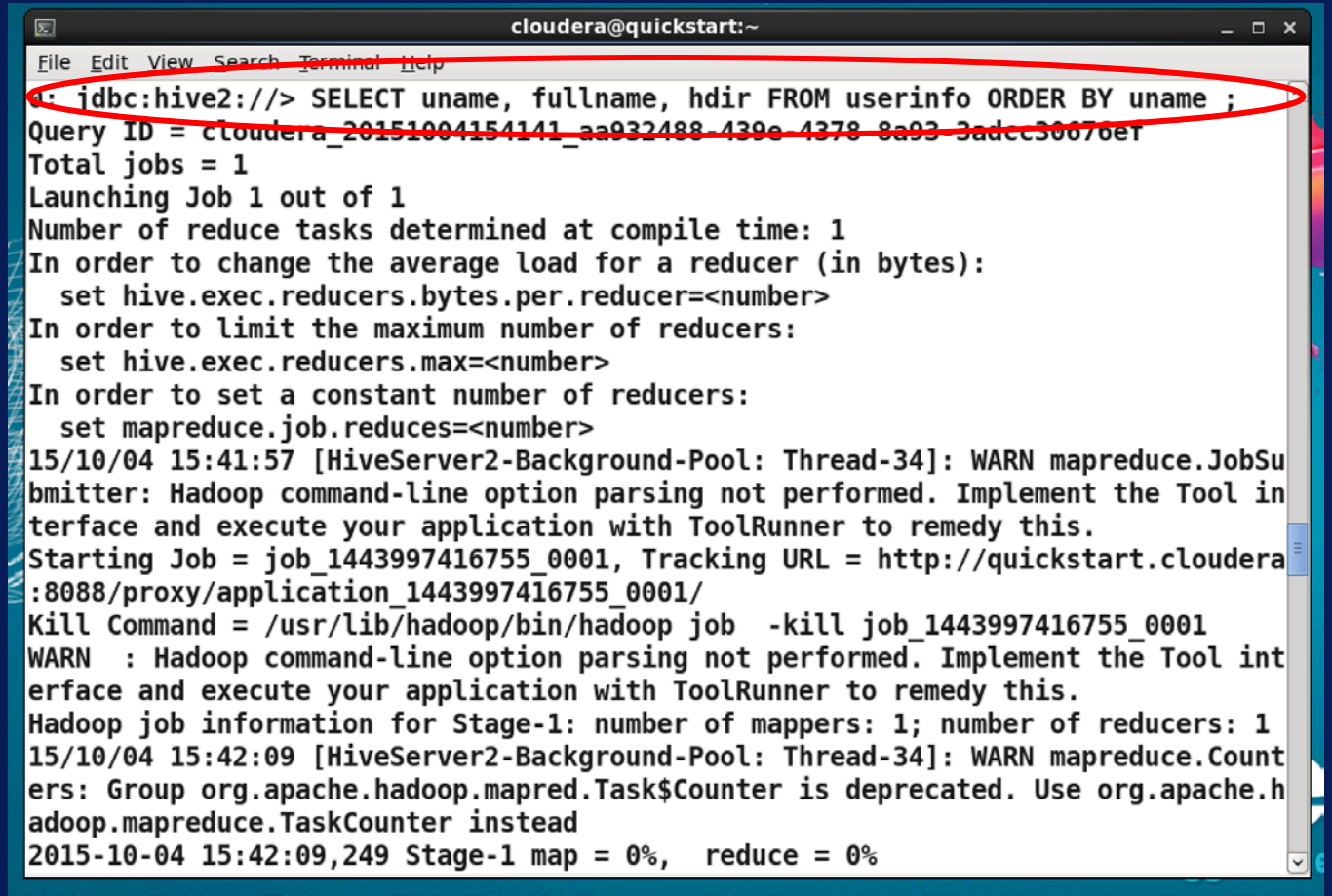
- ***Run the Create table command***

Hive Example



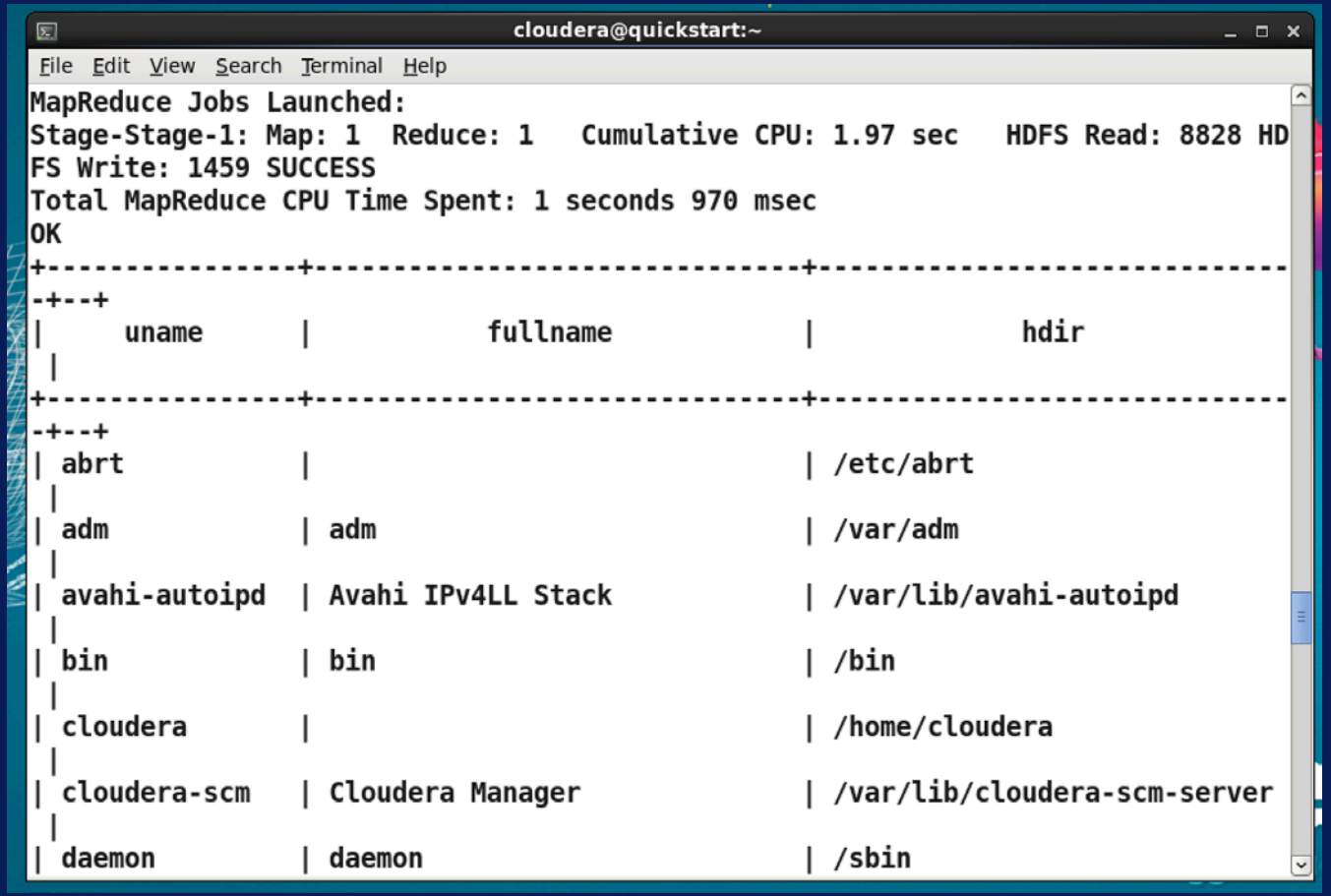
- ***Load passwd file from HDFS***

Hive Example



- *Select info - this launches the Hadoop job and outputs once its complete.*

Hive Example



- ***Completed MapReduce jobs; output shows username, fullname, and home directory.***

Summary

- ***Used beeline for interactive Hive example***
- ***Can also use***
  - *Hive command line interface (CLI)*
  - *Hcatalog*
  - *WebHcat.*

Apache HBase

- *Hbase features*
- *Run interactively using HBase shell.*
- *List other access mechanisms*

Apache HBase

- ***Scalable data store***
- ***Non-relational distributed database***
- ***Runs on top of HDFS***
- ***Compression***
- ***In-memory operations: MemStore, BlockCache***

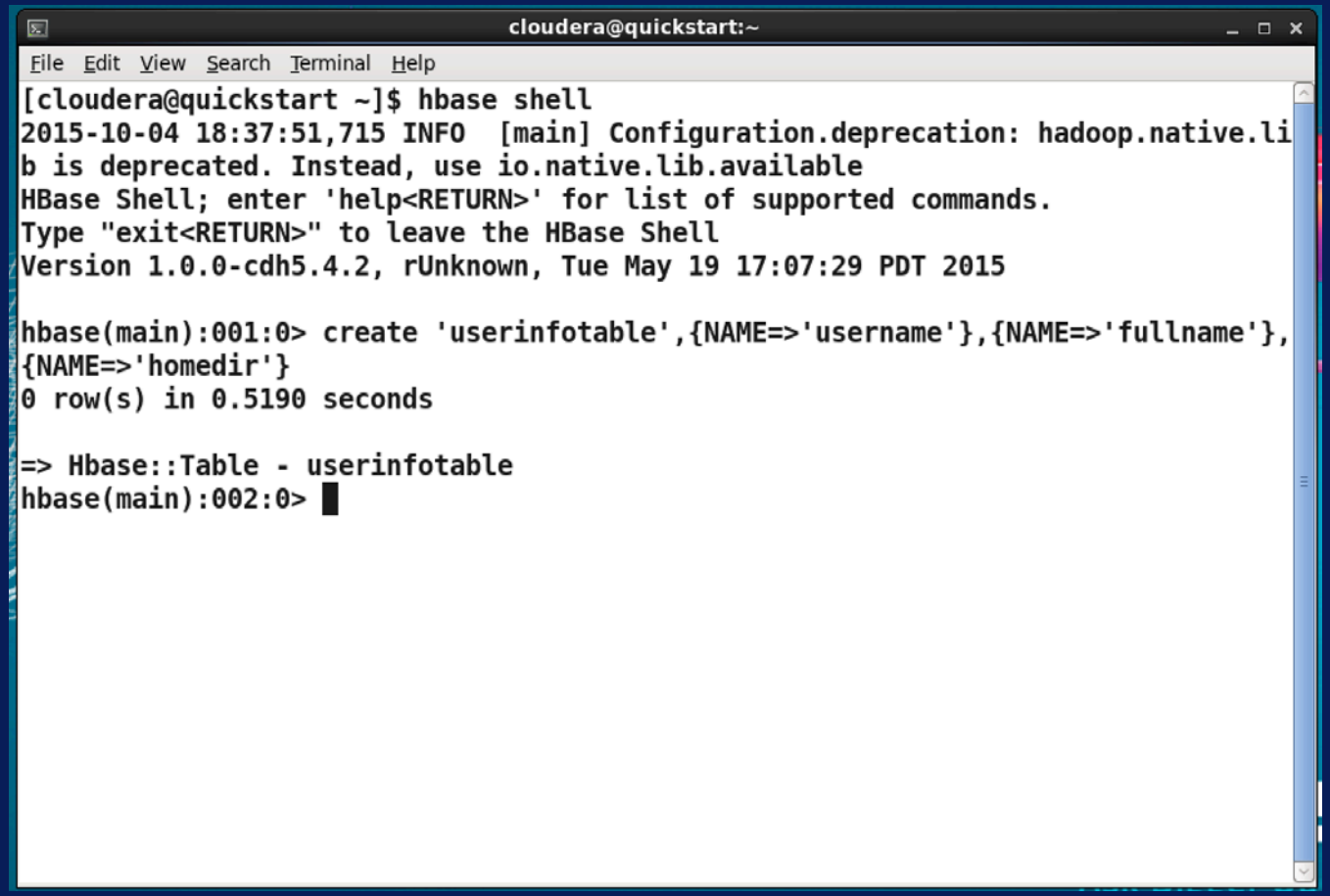
HBase Features

- ***Consistency***
- ***High Availability***
- ***Automatic Sharding***

Hbase Features

- ***Replication***
- ***Security***
- ***SQL like access (Hive, Spark, Impala)***

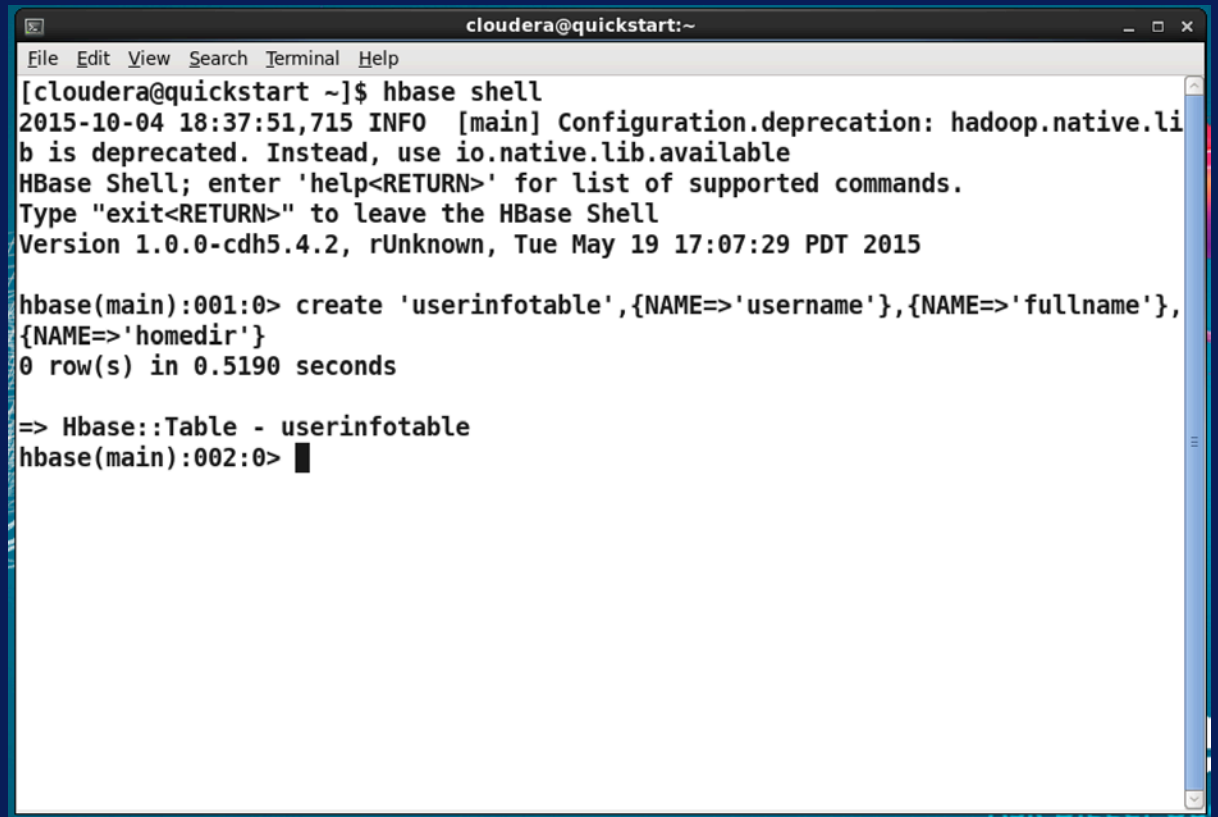
HBase Example



- ***Start HBase shell:***
  hbase shell

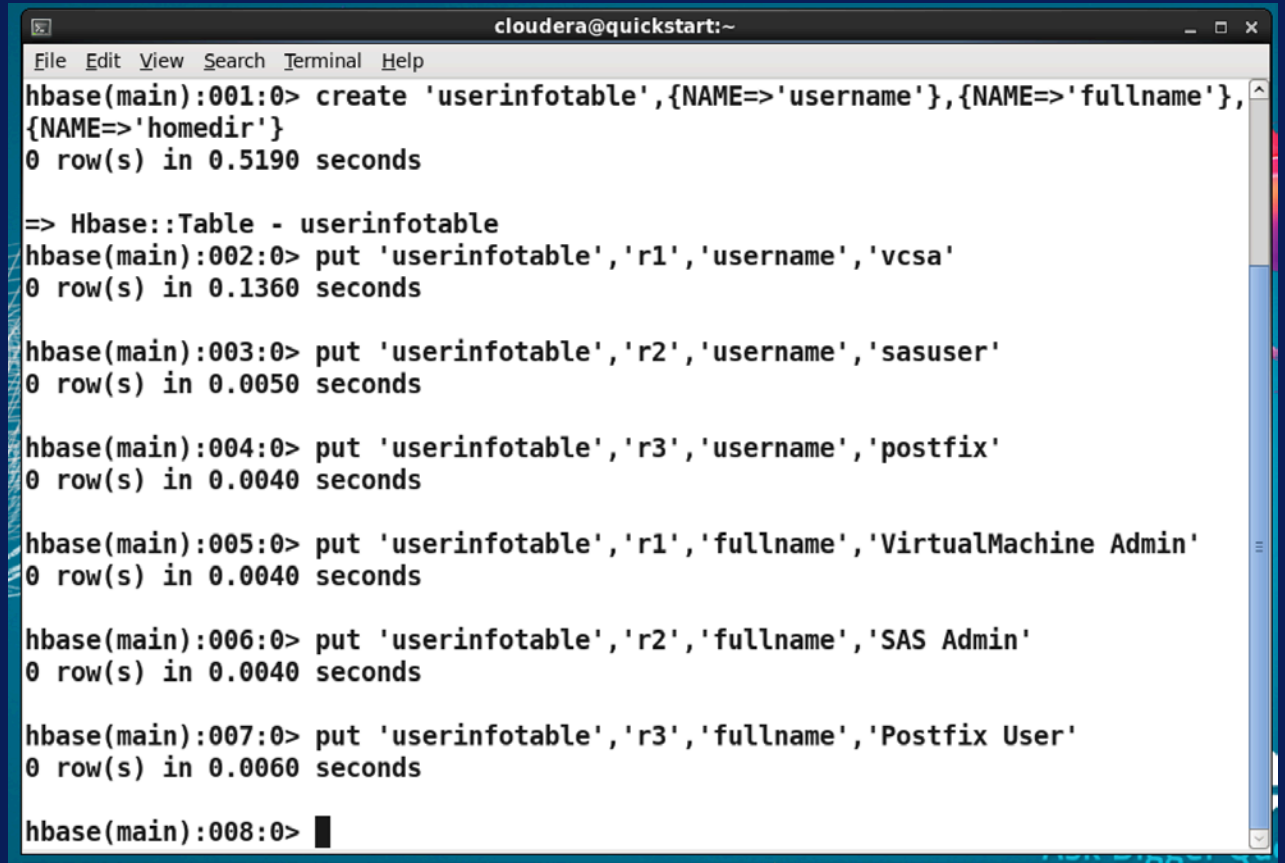
HBase Example

- *Create Table: create ‘usertableinfo’,{NAME=>’username’},{NAME=>’fullname’},{NAME=>’hom*



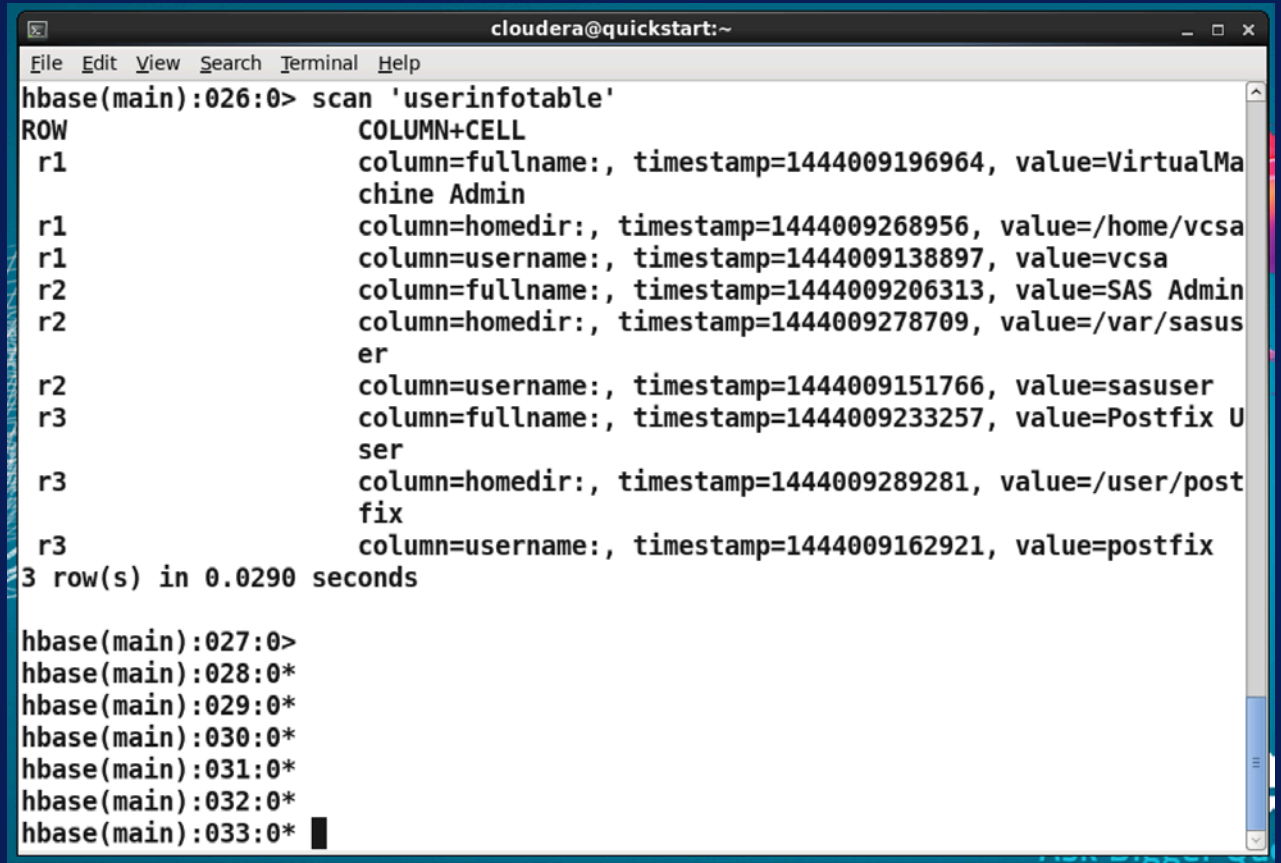
HBase Example

- *Add data: put ‘userinfotable’,’r1’,’username’,’vcsa’*



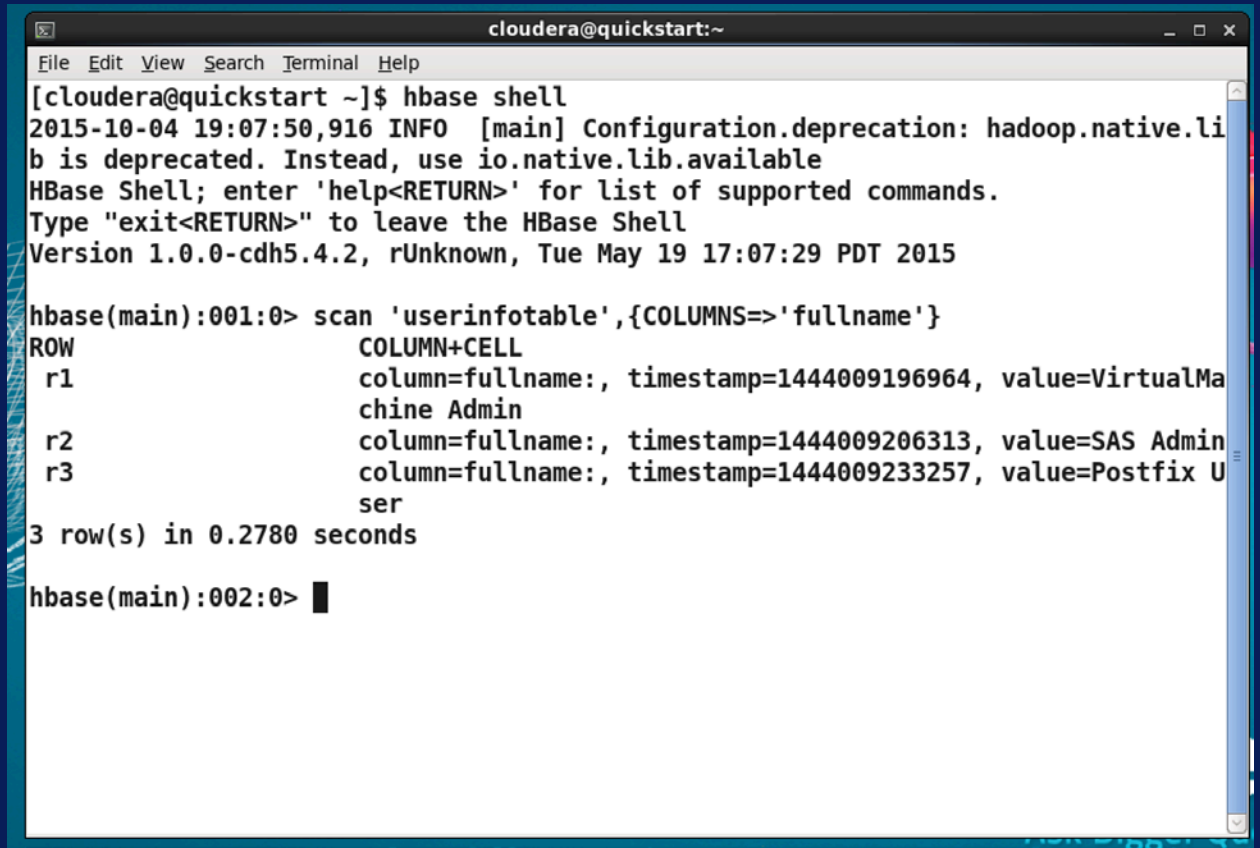
HBase Example

- *Scan table after data entry: scan ‘userinfotable’*



HBase Example

- ***Select info from all rows corresponding to column ‘fullname’.***



Summary

- ***We used: Apache HBase Shell***
- ***Other options:***
  - *HBase, MapReduce*
  - *HBase API*
  - *HBase External API*