CMU MSP 36602: Hive

H. Seltman, Mar 18, 2019

1) **Hive** is an alternative to Pig that also separates the user from the details of MapReduce. It is classified as a "data warehouse infrastructure built on top of Hadoop".
   a) Hive's interface is like SQL, but unlike a true RDMBS, it uses "schema on read" to convert raw data to a specific useful form when the query is run rather that in advance. The raw capabilities of Pig and Hive are very similar, but the way programs are written are quite different. Hive requires only SQL knowledge and not programming knowledge.
   b) Hive is as an efficient **ETL** (Extract, Transform, Load) tool. It is good for **OLAP** (OnLine Analytical Processing). It does not support row level updating, so it is not good for **OLTP** (OnLine Transactional Processing).
   c) Hive is much more scalable than a RDBMS.
   d) A core component of hive is the "metastore", which is a database containing the schemas for the data tables currently accessible in hive.
2) ***Links***:
   a) Official Website: http://hive.apache.org/
   b) The Free Hive Book: http://www.semantikoz.com/blog/the-free-apache-hive-book/
   c) Cheatsheet: https://2xbbhjxc6wk3v21p62t8n4d4-wpengine.netdna-ssl.com/wp-content/uploads/2016/05/Hortonworks.CheatSheet.SQLtoHive.pdf
3) ***Hive SQL language details***
   a) Case insensitive
   b) Lines end with semicolon
   c) Comments start with -- on a separate line (see below)
4) ***Other Hive details***
   a) Works in interactive or batch mode
   b) It ***stores*** datasets in the Hadoop file system
5) ***Example 1***: race car drivers

```
$ wget https://github.com/hortonworks/data-
tutorials/raw/master/tutorials/hdp/interactive-query-for-hadoop-with-
apache-hive-on-apache-tez/assets/driver_data.zip
$ unzip driver_data.zip
$ tail –n+2 drivers.csv > headlessDrivers.csv
$ tail –n+2 timesheet.csv > headlessTimesheet.csv
$ hdfs dfs –mkdir hiveEx
$ hdfs dfs –put headless* hiveEx/.
$ hdfs dfs –ls hiveEx

$ hive
hive> !ls;

hive> dfs –ls hive;
-rw-r--r--   1 student supergroup       1995 2017-04-19 08:21
   hiveEx/headlessDrivers.csv

hive> create table drivers (driverId int, name string, ssn bigint,
      location string, certified string, wageplan string)
      row format delimited fields terminated by ',';
```

```
hive> load data inpath 'hiveEx/headlessDrivers.csv' overwrite into
      table drivers;
Loading data to table default.drive
Table default.drive stats: [numFiles=1, numRows=0, totalSize=1995,
                            rawDataSize=0]

hive> dfs –ls hive;

hive> -- Comments must be on a separate line!!

    > -- Note (above) that the input file was "consumed" on load.
    > show tables;
drivers

hive> select * from drivers limit 4;
  10     George Vetticaden 621011971   244-4532 Nulla Rd.      N     miles
  11     Jamie Engesser    262112338   366-4125 Ac Street      N     miles
  12     Paul Coddin 198041975   Ap #622-957 Risus. Street     Y     hours
  13     Joe Niemiec 139907145   2071 Hendrerit. Ave     Y     hours

hive> describe drivers;
driverid             int
name                 string
ssn                  bigint
location             string
certified            string
wageplan             string

hive> select name, ssn from drivers where instr(lower(name), "to") > 0;
Tom McCuch     363303105
Ryan Templeton 290304287
Dave Patton    977706052

hive> select count(*) from drivers;
Query ID = student_20170419082834_1f7a33cf-b78c-49bb-bf97-14c10b428035
...
FS Write: 3 SUCCESS
Total MapReduce CPU Time Spent: 1 seconds 880 msec
34

hive> create table timesheet (driverId int, week int, hours_logged int,
      miles_logged int)
      row format delimited fields terminated by ',';

hive> -- 'local' allows loading data from Linux side (without consuming)
    > load data local inpath 'headlessTimesheet.csv' overwrite into
        table timesheet;

hive> show tables;
drivers
timesheet

hive> describe timesheet;
driverid             int
week                 int
hours_logged         int
miles_logged         int
```

```
select * from timesheet limit 5;
10 1     70    3300
10 2     70    3300
10 3     60    2800
10 4     70    3100
10 5     70    3200

hive> select driverId, sum(hours_logged), sum(miles_logged) from timesheet
         group by driverId;
Query ID = student_20170418192553_e130cf07-5b5b-4a58-9980-a630daea5e35
Total jobs = 1
...
DFS Write: 510 SUCCESS
Total MapReduce CPU Time Spent: 1 seconds 910 msec

10 3232  147150
11 3642  179300
...
41 2723  138407
42 2697  136673

hive> create table forExport as
      select d.driverId, d.name, t.total_hours, t.total_miles
        from drivers d
      join (select driverId, sum(hours_logged) total_hours,
                   sum(miles_logged) total_miles FROM timesheet
            group by driverId ) t
      on (d.driverId = t.driverId);
Query ID = student_20170418193121_4d112f82-bc28-49e3-ae7b-1a46972705f8
Total jobs = 2
...
035 SUCCESS
Total MapReduce CPU Time Spent: 2 seconds 820 msec

hive> select * from forExport;
10 George Vetticaden 3232  147150
11 Jamie Engesser    3642  179300
...
42 Randy Gelhausen   2697  136673
43 Dave Patton 2750  136993

hive> insert overwrite local directory 'results'
        row format delimited fields terminated by ','
        select * from forExport;
Query ID = student_20170418194309_74cd908d-0b5a-4950-88ee-472efb07c64e
Total jobs = 1
...
61 SUCCESS
Total MapReduce CPU Time Spent: 760 msec

hive> show tables;
drivers
forexport
timesheet

hive> drop table forexport;
hive> show tables;
```

```
drivers
timesheet

hive> exit;
$ ls results
000000_0
$ mv results/000000_0 results.csv
$ rm –r results  # very dangerous – be careful
$ head results.csv
```

6) ***Example 2***: As a hive script

drivers.sql contains:

```
drop table drivers;
drop table timesheet;

create table drivers (driverId int, name string, ssn bigint,
      location string, certified string, wageplan string)
      row format delimited fields terminated by ',';

load data local inpath 'headlessDrivers.csv' overwrite into
      table drivers;

create table timesheet (driverId int, week int, hours_logged int,
      miles_logged int)
      row format delimited fields terminated by ',';

load data local inpath 'headlessTimesheet.csv' overwrite into
      table timesheet;

create table myExport as
      select d.driverId, d.name, t.total_hours, t.total_miles
        from drivers d
      join (select driverId, sum(hours_logged) total_hours,
                  sum(miles_logged) total_miles FROM timesheet
            group by driverId ) t
      on (d.driverId = t.driverId);

insert overwrite local directory 'here' row format
   delimited fields terminated by ',' select * from forExport;
```

At the Unix prompt, run:

```
hive -f drivers.sql
```