Purdue University

Fall 2016

CS 541: Database Systems

Hadoop Cluster: Running SQL Queries on Hive and Spark

/**********
* Overview
**********/

For the final class project, you will explore ideas for optimizing SQL-style queries in Hive and Spark. This handout is intended to introduce the APIs for executing basic queries with these tools.

/**********
* Dataset, Queries, and Setup
**********/

For the final project, you will run queries over a well-known synthetic dataset called TPC-H. This dataset is frequently used to benchmark traditional Relational Database Management Systems. It consists of 8 tables.

http://www.tpc.org/tpch/

For a description of the dataset schema, follow the "TPC-H Specifications and Tools" link from the page above, open the pdf for TPC-H 2.17.1, and begin reading at page 14.

CAUTION: We will provide steps below to generate a base dataset. Disk space will be a concern during the final project. The cluster data nodes have 5,600 GB of disk space, and 5,600 GB / 50 students / HDFS replication factor 3 ~ 35 GB per student. For your first dataset, we recommend generating only the 1GB dataset to maximize the space remaining for your project.

Follow these steps to generate a TPC-H dataset and load it into HDFS.

```
# Get a copy of the TPC-H source.
# Normally, you would download from their website,
# but they require email registration.
# There is a copy of the source on the master node
```

```
# at /home/shared_ostack.
$ mkdir tpch
$ cd tpch
$ cp /home/shared_ostack/tpch_2_17_0.zip .
$ unzip tpch_2_17_0.zip
$ cd tpch_2_17_0/dbgen
$ cp makefile.suite makefile.suite.bak
$ vim makefile.suite
# Modify the following lines:
CC = gcc
DATABASE = ORACLE
MACHINE = LINUX
WORKLOAD = TPCH
# Save and quit.
# Compile.
$ make -f makefile.suite
# Run the generator. (Generates 1GB dataset by default.)
$ ./dbgen
# Collect and view the generated tables.
$ mkdir tables
$ mv *.tbl tables
$ ls -lh tables

# Load the data into HDFS.
# Normally we would simply put all of the table files in
# a single HDFS directory, but instead we must create one
# directory for each file because of a quirk with
# Hive's table naming system.
$ BASE=/user/[username]_ostack/tpch_00
$ hdfs dfs -mkdir $BASE
$ cd tables
$ for tableName in part supplier partsupp customer orders lineitem
nation region; do hdfs dfs -mkdir $BASE/$tableName; hdfs dfs -put
./$tableName.tbl $BASE/$tableName; echo $tableName; done
$ cd ..

# At this point, if you are still in the tpch_2_17_0/dbgen directory,
you can view a list of the standard TPC-H queries.
$ ls queries

# We have provided scripts that demonstrate how to run TPC-H Query 1.
# Get a copy of example query scripts.
$ cp -r /home/shared_ostack/scripts .
```

```
$ cd scripts

# Edit all four script files so that the variables contain your
username.

# Register the table files in Hive.
$ hive -f register_tables.hive

# Run TPC-H Query 1 in Hive.
$ hive -f q1.hive

# "Register" tables in Spark.
# Note that this script only registers tables temporarily.
# It is provided mostly for reference.
$ spark-shell -i register_tables.scala

# Run TPC-H Query 1 in Spark.
$ spark-shell -i q1.scala

# Examine output.
$ hdfs dfs -ls /user/[username]_ostack/tpch_out/
$ hdfs dfs -cat /user/[username]_ostack/tpach_out/spark_out/tmp1/*
# Etc.
```

If you cat the Spark SQL Query 1 output, you will notice that it is missing fields 5 and 6. This is because those columns require functionality that is not implemented in Spark SQL version 1.6.0.

The setup scripts provided above are submitted directly to the CLI for Hive and Spark, respectively. Running commands in the CLI interactively is an excellent way to explore the two systems and prototype your solutions. For example, here are some commands you might want to try in the Hive CLI:

```
$ hive
hive> show databases;
hive> use db_[username]_ostack;
hive> show tables;
hive> select * from customer limit 1;

/**********
* Tool Documentation
**********/
```

We are currently running Hive 1.1.0 and Spark 1.6.0 on the Openstack cluster. You can find documentation for these tools here:

https://cwiki.apache.org/confluence/display/Hive/Home
https://spark.apache.org/docs/1.6.0/

/**********
* Deliverables
**********/

No deliverables. This handout is provided exclusively to get you up and running with the Hive and Spark-SQL APIs.