* Checking SQL access on hadoop cluster - use below command

mysql -u retail\_dba -p

when prompted give below password

cloudera

* check for databases available using below command,

show databases;

* select a database using below command,

use <database-name>(as obtained from above command)>

* once logged in, you can run various sql commands checking for database details. Database has ~6 tables as shown below,

mysql> show tables;

+---------------------+

| Tables\_in\_retail\_db |

+---------------------+

| categories |

| customers |

| departments |

| order\_items |

| orders |

| products |

+---------------------+

* another user is root that will give you access to several other tables

mysql -u root -p

when prompted give below password

cloudera

once logged in using ‘root’, you can run various sql commands checking for database details.

mysql> show databases;

+--------------------+

| Database |

+--------------------+

| information\_schema |

| cm |

| firehose |

| hue |

| metastore |

| mysql |

| nav |

| navms |

| oozie |

| retail\_db |

| rman |

| sentry |

+--------------------+

Next step is to test libraries such as sqoop/hive/spark/hdfs are available in cloudera or not using below; in order

to test all of these simply type commands with names sqoop/hive/spark/hdfs and press enter, this will take you to the respective libraries.

* Hostname -f will give you current hostname

hostname ip: 192.168.19.129

Sqoop:-

sqoop list-databases

--connect "jdbc:mysql://quickstart.cloudera:3306"

--username retail\_dba

--password cloudera

sqoop list-tables

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db"

--username retail\_dba

--password cloudera

sqoop eval

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db"

--username retail\_dba

--password cloudera

--query "select count(1) from order\_items"

* command to check namenode aand datanode

ps -fu hdfs, this will list output as shown below

UID PID PPID C STIME TTY TIME CMD

hdfs 2343 1 0 Nov05 ? 00:03:27 /usr/java/jdk1.7.0\_67-cloudera/bin/java -Dproc\_datanode -Xmx1000m -Dhadoop.log.dir=/var/log/hadoop-hdfs -Dhadoop.log.fil

hdfs 2421 1 0 Nov05 ? 00:01:41 /usr/java/jdk1.7.0\_67-cloudera/bin/java -Dproc\_journalnode -Xmx1000m -Dhadoop.log.dir=/var/log/hadoop-hdfs -Dhadoop.log.

hdfs 2502 1 0 Nov05 ? 00:05:47 /usr/java/jdk1.7.0\_67-cloudera/bin/java -Dproc\_namenode -Xmx1000m -Dhadoop.log.dir=/var/log/hadoop-hdfs -Dhadoop.log.fil

hdfs 2603 1 0 Nov05 ? 00:02:05 /usr/java/jdk1.7.0\_67-cloudera/bin/java -Dproc\_secondarynamenode -Xmx1000m -Dhadoop.log.dir=/var/log/hadoop-hdfs -Dhadoo

[cloudera@quickstart ~]$

command to check resource manager and node manager

ps -fu yarn

to access more commands, you can access github account named as ‘dgadiraju’

###

**SQOOP**

Connect & list database:

[cloudera@quickstart ~]$ sqoop list-databases --connect "jdbc:mysql://quickstart.cloudera:3306" --username root --password cloudera

16/11/06 09:00:27 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0

16/11/06 09:00:27 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.

16/11/06 09:00:28 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.

information\_schema

cm

firehose

hue

metastore

mysql

nav

navms

oozie

retail\_db

rman

sentry

List tables in database:

[cloudera@quickstart ~]$ sqoop list-tables --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" --username retail\_dba --password cloudera

16/11/06 08:58:34 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0

16/11/06 08:58:34 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.

16/11/06 08:58:34 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.

categories

customers

departments

order\_items

orders

products

[cloudera@quickstart ~]$

[cloudera@quickstart ~]$

Run SQL query using “eval” command:

[cloudera@quickstart ~]$ sqoop eval --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" --username retail\_dba --password cloudera --query "select \* from categories"

16/11/06 08:58:58 INFO sqoop.Sqoop: Running Sqoop version: 1.4.6-cdh5.8.0

16/11/06 08:58:59 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.

16/11/06 08:58:59 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.

----------------------------------------------------

| category\_id | category\_department\_id | category\_name |

----------------------------------------------------

| 1 | 2 | Football |

| 2 | 2 | Soccer |

| 3 | 2 | Baseball & Softball |

| 4 | 2 | Basketball |

| 5 | 2 | Lacrosse |

| 6 | 2 | Tennis & Racquet |

| 7 | 2 | Hockey |

| 8 | 2 | More Sports |

| 9 | 3 | Cardio Equipment |

| 10 | 3 | Strength Training |

| 11 | 3 | Fitness Accessories |

| 12 | 3 | Boxing & MMA |

| 13 | 3 | Electronics |

| 14 | 3 | Yoga & Pilates |

| 15 | 3 | Training by Sport |

| 16 | 3 | As Seen on TV! |

| 17 | 4 | Cleats |

| 18 | 4 | Men's Footwear |

| 19 | 4 | Women's Footwear |

Sqoop import:

Sqoop help – command to list out all different arguments that can be used with Sqoop

sqoop import-all-tables \

-m 1 \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--as-textfile \

--warehouse-dir=/user/cloudera/sqoop\_import/

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--as-avrodatafile \

--target-dir=/user/cloudera/departments

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--as-sequencefile \

--target-dir=/user/cloudera/departments

sqoop import-all-tables \

-m 6 \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--as-avrodatafile \

--warehouse-dir=/user/hive/warehouse/retail\_stage.db

Create table in hive using imported file from mysql

-- A file with extension avsc will be created under the directory from which sqoop import is executed

-- Copy avsc file to HDFS location

-- Create hive table with LOCATION to /user/cloudera/departments and TBLPROPERTIES pointing to avsc file

hadoop fs -put sqoop\_import\_departments.avsc /user/cloudera

CREATE EXTERNAL TABLE departments

ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.avro.AvroSerDe'

STORED AS INPUTFORMAT 'org.apache.hadoop.hive.ql.io.avro.AvroContainerInputFormat'

OUTPUTFORMAT 'org.apache.hadoop.hive.ql.io.avro.AvroContainerOutputFormat'

LOCATION 'hdfs:///user/cloudera/dept'

TBLPROPERTIES ('avro.schema.url'='hdfs://quickstart.cloudera/user/cloudera/departments.avsc');

sqoop import-all-tables \

--num-mappers 1 \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--hive-import \

--hive-overwrite \

--create-hive-table \

--compress \

--compression-codec org.apache.hadoop.io.compress.SnappyCodec \

--outdir java\_files

Create hive database using below command;

>hive -e "CREATE DATABASE IF NOT EXISTS retail\_stage"

Run below sqoop import-all-tables command to load data into existing hive database

sqoop import-all-tables

--num-mappers 1

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db"

--username=retail\_dba

--password=cloudera

--hive-import

--hive-overwrite

--create-hive-table

--outdir java\_files

--hive-database retail\_stage

Validate by running this query hive -e "USE retail\_stage; SHOW TABLES; SELECT \* FROM departments;"

Below command is used to load data into a hive table directly via Sqoop, table should already exists,

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--hive-home /user/hive/warehouse \

--hive-import \

--hive-overwrite \

--hive-table sqoop\_import.departments \

--outdir java\_files

Use Sqoop to load data into a table and create the hive table at the same time, Create hive table example

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--hive-home /user/hive/warehouse \

--hive-import \

--hive-table sqoop\_import.departments\_test \

--create-hive-table \

--outdir java\_files

-- Basic import

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/cloudera/departments

-- Boundary Query and columns

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/cloudera/departments \

-m 2 \

--boundary-query "select min(department\_id), max(department\_id) from departments where department\_id < 9000 limit 1" \

--columns department\_id,department\_name

-- query and split-by

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--query="select \* from orders join order\_items on orders.order\_id = order\_items.order\_item\_order\_id where \$CONDITIONS" \

--target-dir /user/cloudera/order\_join \

--split-by order\_id \

--num-mappers 4

-- Copying into existing table or directory (append)

-- Customizing number of threads (num-mappers)

-- Changing delimiter

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/hive/warehouse/departments \

--append \

--num-mappers 1 \

--outdir java\_files

Getting delta (--where)

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/hive/warehouse/departments \

--append \

-m 1 \

--split-by department\_id \

--where "department\_id > 7" \

--outdir java\_files

Incremental load

sqoop import \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/hive/warehouse/departments \

--append \

--check-column "department\_id" \

--incremental append \

--last-value 7 \

--outdir java\_files

Job creation:

sqoop job --create sqoop\_job \

-- import \

--num-mappers 1 \

--connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username=retail\_dba \

--password=cloudera \

--table departments \

--target-dir /user/hive/warehouse/departments \

--append \

--fields-terminated-by '|' \

--lines-terminated-by '\n' \

--check-column "department\_id" \

--incremental append \

--last-value 7 \

--outdir java\_files

sqoop job --list

sqoop job --show sqoop\_job

sqoop job --exec sqoop\_job

-Insert

sqoop eval --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--query "insert into departments values (10000, 'Inserting for merge')"

sqoop eval --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--query "insert into departments values (8000, 'Bigger record')"

sqoop eval --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--query "insert into departments values (9000, 'Even Bigger record')"

HADOOP\_MAPRED\_HOME is /usr/lib/hadoop-mapreduce

**Export**

--Connect to mysql and create database for reporting database

--user:root, password:cloudera

mysql -u root -p

create database retail\_rpt\_db;

grant all on retail\_rpt\_db.\* to retail\_dba;

flush privileges;

use retail\_rpt\_db;

create table departments as select \* from retail\_db.departments where 1=2;

exit;

--For certification change database name retail\_rpt\_db to retail\_db

sqoop export --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--table order\_items\_export \

--export-dir /user/cloudera/order\_items \

--num-mappers 1 \

--batch \

--outdir java\_files

sqoop export --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--table departments \

--export-dir /user/cloudera/departments \

--batch \

--outdir java\_files \

-m 1 \

--update-key department\_id \

--update-mode allowinsert

sqoop export --connect "jdbc:mysql://quickstart.cloudera:3306/retail\_db" \

--username retail\_dba \

--password cloudera \

--table departments\_test \

--export-dir /user/hive/warehouse/departments \

--input-fields-terminated-by '\001' \

--input-lines-terminated-by '\n' \

--num-mappers 2 \

--batch \

--outdir java\_files \

--input-null-string nvl \

--input-null-non-string -1

--Merge

sqoop merge --merge-key department\_id \

--new-data /user/cloudera/departments/dept \

--onto /user/cloudera/department \

--target-dir /user/cloudera/sqoop\_merge/departments \

--class-name departments \

--jar-file /tmp/sqoop-cloudera/compile/fc807ef99e35d1a3fd1f0f6c66df648f/departments.jar

hadoop fs -cat /user/cloudera/sqoop\_merge/departments/part\*

--Delete old directory

hadoop fs -rm -R /user/cloudera/departments/dept

--Move/rename stage directory to original directory

hadoop fs -mv /user/cloudera/sqoop\_merge/departments /user/cloudera/departments/dept

--Validate that original directory have merged data

hadoop fs -cat /user/cloudera/departments/dept/part\*