Big Data: How can I import data from MySQL to Hadoop with Apache Sqoop?

How can I import data from MySQL to Hadoop with Apache Sqoop?

This is best explained with an example.

Apache Sqoop is a data transfer tool used to move data between Hadoop and structured datastores. We will show how to "ingest" MySQL data into Hadoop with Sqoop2, with a little help from Connector/J.

A pre-existing Apache Hadoop 2.2.0 instance (with name "hdfs1") is used in this example.

1. Download Sqoop2 tarball and unpack it

```
# cd /cm/shared/apps/hadoop/Apache/
# tar xvf /tmp/sqoop-1.99.4-bin-hadoop200.tar.gz
```

2. Install Connector/J for MySQL and copy it to Sqoop2 directory

```
# yum install mysql-connector-java

# cp /usr/share/java/mysql-connector-java-5.1.17.jar /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server/webapps/sqoop/WEB-INF/lib
```

3. Edit Sqoop configuration file sqoop.properties

```
# cd /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server/conf
# vi sqoop.properties
```

Modify the following line using the configuration directory for the Hadoop instance (e.g. 'hdfs1')

```
org.apache.sqoop.submission.engine.mapreduce.configuration.directory=/etc/hadoop/hdfs1/
```

4. Edit Sqoop configuration file catalina.properties

```
# cd /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server/conf
# vi catalina.properties
```

The value of `common.loader` should be customized using the `$HADOOP_PREFIX` for the Hadoop instance, which in the example is `/cm/shared/apps/hadoop/hdfs1`

```
common.loader=${catalina.base}/lib,${catalina.base}/lib/*.jar,${catalina.home}/lib,${catalina.home}/lib/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/common/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/common/lib/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/hdfs/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/hdfs/lib/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/mapreduce/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/mapreduce/lib/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/tools/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/tools/lib/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/yarn/*.jar,/cm/shared/apps/hadoop/hdfs1/share/hadoop/yarn/lib/*.jar
```

5. Verify Sqoop configuration

```
# cd /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/bin
# sh ./sqoop2-tool verify
```

6. Grant privileges to MySQL user 'testuser'

```
# mysql -u root -p cmdaemon
> GRANT ALL PRIVILEGES ON cmdaemon.* TO 'testuser'@'%' IDENTIFIED BY 'testpass';
```

7. Start Sqoop server

```
# cd /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/bin
# sh ./sqoop2-server start
Sqoop home directory: /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200
Setting SQOOP_HTTP_PORT:     12000
Setting SQOOP_ADMIN_PORT:     12001
Using   CATALINA_OPTS:
Adding to CATALINA_OPTS:    -Dsqoop.http.port=12000 -Dsqoop.admin.port=12001
Using CATALINA_BASE:
/cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server
Using CATALINA_HOME:
/cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server
Using CATALINA_TMPDIR:
/cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server/temp
Using JRE_HOME:        /usr/lib/jvm/jre-1.7.0-openjdk.x86_64/
Using CLASSPATH:
/cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/server/bin/bootstrap.jar
```

8. Create links to HDFS and MySQL Sqoop shell

In this example, the NameNode service is running on node001, port 8020

```
# cd /cm/shared/apps/hadoop/Apache/sqoop-1.99.4-bin-hadoop200/bin
# sh ./sqoop2-shell
sqoop:000> show connector
+----+------------------------+---------+------------------------------------------------------+----------------------+
| Id |          Name          | Version |                      Class                           | Supported Directions |
+----+------------------------+---------+------------------------------------------------------+----------------------+
| 1  | hdfs-connector         | 1.99.4  | org.apache.sqoop.connector.hdfs.HdfsConnector        |
```

```
FROM/TO              |
| 2  | generic-jdbc-connector | 1.99.4  | org.apache.sqoop.connector.jdbc.GenericJdbcConnector
| FROM/TO              |
+----+------------------------+---------+-----------------------------------------------------+----------------------+
sqoop:000> create link -c 1
Creating link for connector with id 1
Please fill following values to create new link object
Name: hdfs

Link configuration

HDFS URI: hdfs://node001:8020
New link was successfully created with validation status OK and persistent id 1
sqoop:000> show link
+----+------+-----------+---------+
| Id | Name | Connector | Enabled |
+----+------+-----------+---------+
| 1  | hdfs | 1         | true    |
+----+------+-----------+---------+
sqoop:000> create link -c 2
Creating link for connector with id 2
Please fill following values to create new link object
Name: mysql

Link configuration

JDBC Driver Class: com.mysql.jdbc.Driver
JDBC Connection String: jdbc:mysql://hadoopdev/cmdaemon
Username: testuser
Password: ********
JDBC Connection Properties:
There are currently 0 values in the map:
entry# protocol=tcp
There are currently 1 values in the map:
protocol = tcp
entry#
New link was successfully created with validation status OK and persistent id 2
sqoop:000> show link
+----+-------+-----------+---------+
| Id | Name  | Connector | Enabled |
+----+-------+-----------+---------+
| 1  | hdfs  | 1         | true    |
| 2  | mysql | 2         | true    |
+----+-------+-----------+---------+
```

Big Data: How can I import data from MySQL to Hadoop with Apache Sqoop?

9. Create Sqoop job to "ingest" data from MySQL to HDFS

```
sqoop:000> create job --from 2 --to 1
Creating job for links with from id 2 and to id 1
Please fill following values to create new job object
Name: testjob

From database configuration

Schema name: cmdaemon
Table name: HadoopHDFSs
Table SQL statement:
Table column names:
Partition column name:
Null value allowed for the partition column:
Boundary query:

ToJob configuration

Output format:
  0 : TEXT_FILE
  1 : SEQUENCE_FILE
Choose: 0
Compression format:
  0 : NONE
  1 : DEFAULT
  2 : DEFLATE
  3 : GZIP
  4 : BZIP2
  5 : LZO
  6 : LZ4
  7 : SNAPPY
  8 : CUSTOM
Choose: 0
Custom compression format:
Output directory: /user/root/

Throttling resources

Extractors:
Loaders:
New job was successfully created with validation status OK  and persistent id 1
sqoop:000> show job
+----+---------+----------------+--------------+---------+
| Id |  Name   | From Connector | To Connector | Enabled |
+----+---------+----------------+--------------+---------+
```

```
| 1  | testjob | 2              | 1           | true    |
+----+---------+----------------+-------------+---------+
```

10. Start Sqoop job and check its status

```
sqoop:000> start job -j 1
Submission details
Job ID: 1
Server URL: http://localhost:12000/sqoop/
Created by: root
Creation date: 2014-12-09 16:03:50 CET
Lastly updated by: root
External ID: job_1418134094039_0004
    http://node003:8088/proxy/application_1418134094039_0004/
2014-12-09 16:03:50 CET: BOOTING  - Progress is not available
sqoop:000> status job -j 1
```

11. Check "ingest" result

```
# module load hadoop/hdfs1
# hdfs dfs -cat /user/root/*
103079215108,NULL,'hdfs1',0,'2.6.0','Apache','/etc/hadoop/hdfs1','/cm/shared/apps/hadoop/hdfs1','/etc/hadoop/zookeeper',NULL,'/etc/hadoop/hbase',NULL,'installed from: /cm/local/apps/hadoop/hadoop-2.6.0.tar.gz',NULL,10,'dataNode',2,1,'/tmp/hadoop/hdfs1/',false,'077',false,67108864,3,512,'/var/log/hadoop/hdfs1','/var/lib/hadoop/hdfs1/',NULL,false,0,false,false,false,NULL,false,NULL,false,false,0,1418124724,NULL,NULL,NULL,NULL,NULL
```

Unique solution ID: #1242
Author: Michele Lamarca
Last update: 2016-11-23 17:21