DADS 6002 / CI 7301
Big Data Analytics
Data Ingestion Lab

Importing from MySQL to HDFS

- Let first create a simple database under MySQL.

  # mysql –uroot –pcloudera

  mysql > create database energydata;

  mysql > use energydata;

  mysql > create table avgprice_by_state (

  year INT NOT NULL,

Importing from MySQL to HDFS

```
        state VARCHAR(5) NOT NULL,
        sector  VARCHAR(255),
        residential  DECIMAL(10,2),
        industrial   DECIMAL(10,2),
        transportation      DECIMAL(10,2),
        other           DECIMAL(10, 2),
        total           DECIMAL(10,2) );
mysql > quit;
```

Importing from MySQL to HDFS

Download a zip file from MS Teams to the shared folder and copy it to the working directory

(Source https://github.com/bbengfort/hadoop-fundamentals/raw/master/data/avgprice_kwh_state.zip)

unzip avgprice_kwh_state.zip

Importing from MySQL to HDFS

```
# mysql -h localhost -uroot -pcloudera
   --local-infile=1
    mysql > use energydata;
    mysql > load data local infile
    '/home/cloudera/avgprice_kwh_state.csv' into table
    avgprice_by_state fields terminated by ',' lines
    terminated by '\n' ignore 1 lines;
    mysql > quit;
```

Importing from MySQL to HDFS

```
# sqoop import --connect
      jdbc:mysql://localhost:3306/energydata
      --username root  --password cloudera --table
      avgprice_by_state --target-dir
      /user/cloudera/energydata  -m  1

# hadoop fs -cat /user/cloudera/energydata/part-m-
      00000
```

Importing from MySQL to Hive

```
# sqoop import --connect jdbc:mysql:
//localhost:3306/energydata --username root --
password cloudera --table avgprice_by_state --
hive-table avgprice --hive-import -m 1

#  hive

hive > select * from avgprice;
```

Importing from MySQL to Hbase

- Let first create a table to be imported under MySQL

  # mysql -uroot -pcloudera

  mysql > create database country_db;

  mysql > use country_db;

  mysql > create table country_tbl

  (id int not null, country varchar(50), primary key (id));

Importing from MySQL to Hbase

mysql > insert into country_tbl values(1, ‘USA’);

mysql > insert into country_tbl values(2, ‘CANADA’);

mysql > insert into country_tbl values(3, ‘JAPAN’);

mysql > insert into country_tbl values(4, ‘ENGLAND’);

mysql > insert into country_tbl values(5, ‘THAILAND’);

mysql > select * from country_tbl;

mysql > quit;

Importing from MySQL to Hbase

```
# sqoop import --connect
   jdbc:mysql://localhost:3306/country_db  --
   username root --password cloudera --table
   country_tbl --hbase-table country  --column-
   family country-cf --hbase-row-key id --hbase-
   create-table -m 1

# hbase shell
   hbase > scan ‘country’
```

Ingesting Product Impression Data

- Use Flume to consume the streaming user-interaction data generated by a hypothetical online store.
- Simulate an ecommerce impression log that records user interactions in the following JSON format:

```
{
    “sku” : “T9921-5”
   “timestamp” : 1453167527737
   “cid” : “51761”
   “action” : “add_cart”
   “ip” : “226.43.51.25”
}
```

Ingesting Product Impression Data

- To create the necessary directories and HDFS, download and run the script as a user with sudo privileges
- Download a flume setup shell file (flume_setup.sh) from MS Teams to the shared folder and copy it the working directory.

(Source : https://raw.githubusercontent.com/bbengfort/hadoop-fundamentals/master/flume/setup.sh)

Ingesting Product Impression Data

- Use nano editor to view the flume_setup.sh as follow :

```
#!/bin/bash
hadoop fs -mkdir -p /user/cloudera/impressions/
hadoop fs -chmod 777 /user/cloudera/impressions/
mkdir /tmp/impressions
chmod  777 /tmp/impressions
mkdir  /tmp/flume
chmod 777 /tmp/flume
```

- Execute the setup file by

```
# sudo su –
# cd /home/cloudera
# sh flume_setup.sh
```

Ingesting Product Impression Data

- Download a python program named impression_tracker.py from MS Teams into the shared folder and copy it to the working directory. We will execute the program to create the log file named impressions.log at /tmp/impressions.

  (Source : https://raw.githubusercontent.com/bbengfort/hadoop-fundamentals/master/flume/impression_tracker.py)

- Add execution privilege to the python program file then execute it as follow :

  # chmod +x impression_tracker.py

  # ./impression_tracker.py

Download Examples of Configuration files

- Download the configuration files, one for client agent and one for collector agent from the MS Teams to the shared folder and then copy them to the working directory.

  (Source : https://raw.githubusercontent.com/bbengfort/hadoop-fundamentals/master/flume/client.conf

  https://raw.githubusercontent.com/bbengfort/hadoop-fundamentals/master/flume/collector.conf)

Configure Source Agent

```
# define spooling directory source :
client.sources=r1
client.sources.r1.channels=ch1
client.sources.r1.type=spooldir
client.sources.r1.spoolDir=/tmp/impressions
# define a file channel:
client.channels=ch1
client.channels.ch1.type=FILE
```

Configure Source Agent

```
# define an Avro sink:
client.sinks=k1
client.sinks.k1.type=avro
client.sinks.k1.hostname=localhost
clinet.sinks.k1.port=4141
client.sinks.k1.channel=ch1
```

Configure Collector Agent

```
# define an Avro source:
collector.sources=r1
collector.sources.r1.type=avro
collector.sources.r1.bind=0.0.0.0
collector.sources.r1.port=4141
collector.sources.r1.channels=ch1
```

Configure Collector Agent

```
# define a file channel using multiple disks for reliability
collector.channels=ch1
collector.channels.ch1.type=FILE
collector.channels.ch1.checkpointDir=/tmp/flume/checkpoint
collector.channels.ch1.dataDir=/tmp/flume/data

# define HDFS sinks to persist events as text
collector.sinks=k1
collector.sinks.k1.type=hdfs
collector.sinks.k1.channel=ch1
```

Configure Collector Agent

```
# HDFS sink configuration
collector.sinks.k1.hdfs.path=/user/cloudera/impressions
collector.sinks.k1.hdfs.filePrefix=impressions
collector.sinks.k1.hdfs.fileSuffix=.log
collector.sinks.k1.hdfs.fileType=DataStream
collector.sinks.k1.hdfs.writeFormat=text
collector.sinks.k1.hdfs.batchSize=1000
```

Running Flume

- Open a new terminal then start a flume service.

sudo su –

service flume-ng-agent start

- Run a collector agent.

flume-ng agent --name collector --conf . --conf-file ./collector.conf

- Open a new terminal then run a client agent.

sudo su –

flume-ng agent --name client --conf . --conf-file ./client.conf

- After finish importing, check for the imported files in the target directory /user/cloudera/impressions

Running Flume

hadoop fs -ls /user/cloudera/impressions

hadoop fs -cat /user/cloudera/impressions/impressions….log

to display one of the imported files

Kafka

- Install Kafka

sudo su –

cd /home/cloudera

mkdir kafka

cd kafka

Download Kafka from MS Teams or from the web by

wget https://archive.apache.org/dist/kafka/0.9.0.1/kafka_2.10-0.9.0.1.tgz

tar xzf kafka_2.10-0.9.0.1.tgz

Kafka

- Open a new terminal and run Kafka Server (Broker)

```
#  sudo su –
#  cd /home/cloudera/kafka/kafka_2.10-0.9.0.1
#  bin/kafka-server-start.sh
   config/server.properties &
```

Kafka

- Open a new terminal and run a Kafka Producer

sudo su –

cd /home/cloudera/kafka/kafka_2.10-0.9.0.1

bin/kafka-console-producer.sh --topic test --broker-list localhost:9092

- Type a few lines of data to send to the Consumer.

This is a test.

Bye, Kafka.

ctrl-D to finish sending

Kafka

- Open a new terminal and run a Kafka Consumer

sudo su –

cd /home/cloudera/kafka/kafka_2.10-0.9.0.1

bin/kafka-console-consumer.sh --topic test --zookeeper localhost:2181 --from-beginning

- The consumer prints the received data from the Producer

This is a test.

Bye, Kafka.

ctrl-c to exit