YCBS-257 - Data at Scale – Khaled Tannir

Final Exam

30% Sunday Mar 10 2019 – 11:59 PM

STREAM DATA INGESTION AND ANALYSIS CASE STUDY

SCENARIO

As a professional Data at Scale consultants, you have been asked to perform 2 mandates:

1. Prepare an assessment of a key project at **Big Company Group** (BCG) with your recommendations to answers some strategic questions for that project and proposed action to implement your solution;

and

2. Help the developers team in creating data ingestion and analysis scripts using Hadoop commons tools such as Pig, Hive, Impala, Sqoop and Flume to improve data ingestion and analysis practices.

REQUIREMENT

- This is an individual assignment and should reflect your understanding of the Hadoop ecosystem and the data at scale problems we are trying to solve.
- You are invited to not share this document in any format and for any reason.

THE CASE STUDY

- The Case Study is based on a real project environment that highlights a number of challenges often found in Hadoop project development. Many of these were addressed as specific topics within the course.

  As such you are encouraged, for each course topic, to identify any related information within the Case Study and to use the course lectures to gain insights for dealing with these challenges.

- **Big Company Group** needs to evaluate the data stream collected in near real-time (< 1 sec) in order to use this data in machine learning, predictive analysis, customer behavior, fraud detection and so on…
- **Big Company Group** develops IoT (Internet of Things) sensors. For a particular experience, there are **5,000** sensors equally split between two cities. There are multiple sensors each with their own data elements, but they follow a common data format.
  - Each sensor transmits on average **50 attribute** records every second.
  - Each sensor's data attribute row is on average **100 bytes** wide.

PROJECT REQUIREMENTS

- Ingest the sensor data into the HDFS store for evaluation and processing.
- Corporate data standards require all input sensor data to be persisted for **12 months**.
- All QA test results data (estimated to 10% of the ingested data per year) to be stored for a period of **5 years**.

PLATFORM PLANNING (10%)

The most common practice to size a Hadoop cluster is sizing the cluster based on the amount of storage required. The more data into the system, the more will be the machines required. Each time you add a new node to the cluster, you get more computing resources in addition to the new storage capacity.

To determine the memory size of the NameNode server, we need to **add the memory** needed by NameNode to manage the HDFS cluster metadata (in memory) and **the memory needed** for the Operating System.
The IT department defined the hardware configuration for each node in the cluster: **1 CPU 8 core**, **32 GB** memory and **20 TB** Hard Disk Drive. The data **node storage capacity** was calculated as **13 TB**

- Assuming a replication factor of **3** how many nodes in the cluster?
- Assuming a HDFS bloc size is **128 MB** and each block need **680 Bytes** for its metadata. What is the recommended NameNode memory size?
- What are the Systems architecture options for pulling the data from the source location in preparation for ingesting into HDFS?

PLATFORM INSTALLATION (10%)

- Describe the relative merits for installing and running the cluster in the cloud vs on-premises.
- How would you verify that the cluster can support the volumes in the project requirements?

DATA INGESTION AND DATA MODELING (80%)

- Given the above project requirements, which components would you use to ingest the data?

The data will be given as many relatively small files. The input data will be **xxxx.csv**-file for each sensor (e.g: **`1763.csv`**). Each of these files contains one row per measurement.

Example: The content of **`1789.csv`** looks similar to this:

```
ITE00100554,17890101,TMAX,-63,,,E,
ITE00100554,17890101,TMIN,-90,,,E,
GM000010962,17890101,PRCP,4,,,E,
EZE00100082,17890101,TMAX,-103,,,E,
EZE00100082,17890101,TMIN,-184,,,E,
ITE00100554,17890102,TMAX,-16,,,E,
ITE00100554,17890102,TMIN,-66,,,E,
GM000010962,17890102,PRCP,15,,,E,
EZE00100082,17890102,TMAX,-98,,,E,
EZE00100082,17890102,TMIN,-170,,,E,
```

Description of the columns:

- the sensor id
- the date in format yyyymmdd
- type of measurement
- temperature in tens of degrees (e.g. -90 = -9.0 deg. C., -184 = -18.4 deg. C.)
- other columns

TASK 01
Import a File into HDFS

1. Create a new directory in HDFS named `/user/bcg/sensors`
2. Put three files (`xxxx.csv, xxxx.csv, xxxx.csv`) from the `/home/datasets/sensors` directory on the local machine into `/user/bcg/sensors` directory in HDFS

TASK 02

Cleanse Data using Pig

Write a Pig script that satisfies all of the following criteria:

- Load all of the data in `/user/bcg/sensors`
- Remove all rows in the sensors data where the `type of measurement` column equals the string `"TMIN"`.
- The output should only contain the `SensorID, Date, TypeM'` and `Temp`
- Store the result as comma-separated records in a new directory in HDFS named `/user/bcg/sensors_clean`

TASK 03

Define a Hive Table

Define a Hive table named **`sensors`** that matches the data stored in your `/user/bcg/sensors_clean` HDFS directory.

The Hive table should satisfy all of the following criteria:

- A Hive non-managed table with the location set to `/user/bcg/sensors_clean`
- The schema matches the columns

```
1. SensorID   string,
2. Date       string,
3. TypeM      string
4. Temp       int
```

TASK 04

Hive Partitioned Tables

Define a new Hive-managed table named **`sensors_partitioned`** that satisfies the following criteria:

- The table has the same schema as the `sensors` table
- The table is partitioned on the **`Year`** and **`Month`** columns
- Populate the table with data from the **`2008_bcg_weather.csv`** file into the appropriate partition of `weather_partitioned`

TASK 05

Define and Populate an ORCFile Table

Define a Hive table named **`sensors_orc`** that satisfies all of the following criteria:

- A Hive-managed table
- The schema matches the columns in `sensors` table
- The table is stored as **`ORCFile`** format
- The table is populated with the records from the csv file 2019_bcg_weather.csv stored on HDFS

TASK 06

Sqoop Export

Assume the data is located into **`/user/bcg/sensors/2019_bcg_weather.csv`** on HDFS

1. Open a new MySQL session using the following command:

```
mysql -u root -p  (password : cloudera)
```

2. Create a new database named **`sensors`**

```
create database sensors;
```

3. Switch to the sensors database

```
use sensors;
```

4. Create a new table named **`weather`** with the following schema:

```
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| stationID     | varchar(100) | YES  |     | NULL    |       |
| year          | varchar(15)  | YES  |     | NULL    |       |
| typeM         | varchar(10)  | YES  |     | NULL    |       |
| temperature   | int(11)      | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
```

5. Use Sqoop to export the **sensors** directory from HDFS to the **weather** table on MySQL.

   i. For the credentials use the following:
   username : **`root`**
   password : `cloudera`

TASK 07

Flume Ingestion

1. Create a Flume agent configuration file that satisfies the following criteria:
   a) Having the Source defined as **`spooldir`** type which allows ingest data by placing files to be ingested into a "spooling" directory on disk.
   b) The output data is written on HDFS into **`/user/bcg/sensors_in/`** directory as a Hive partition. The partition key columns are `Year, Month, Day`.
2. Write the command line to run the Flume agent

What to submit:

One Word document file <your name> - WINTER2019Final.docx

✓ Comments can be submitted in French or in English in the text file