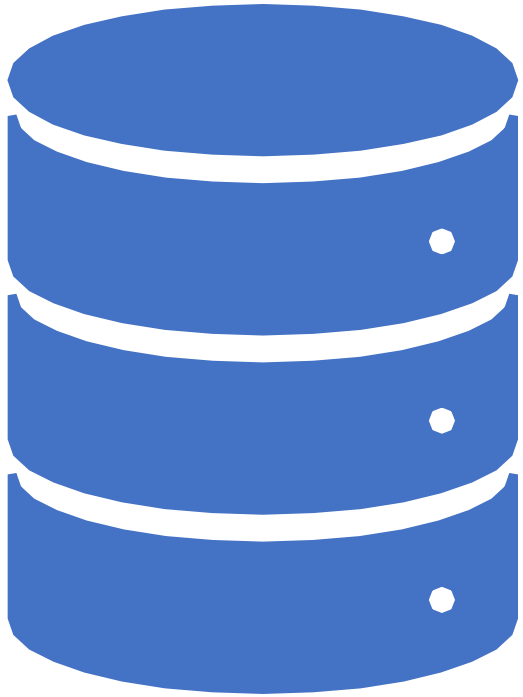

PA220: Database systems for data analytics

Home Assignment 4 & 5

Vlastislav Dohnal

HA 4: Migrate DW to Hive

- Take your Query 4 and migrate the necessary tables to Hive
  - During the process you will execute queries to load data into Hive table, so
  - Report the number of mappers, reducers used as well as execution time.
    - `SELECT DISTINCT program_ver FROM ext_data_sept_oct;`

```
…
INFO  : Stage-Stage-1: Map: 3  Reduce: 2   Cumulative CPU: 17.27 sec   HDFS Read: 458679201 HDFS Write: 446 HDFS EC Read: 0 SUCCESS
INFO  : Total MapReduce CPU Time Spent: 17 seconds 270 msec
INFO  : Completed executing command(queryId=hive_20210121120334_eb686cbc-d712-4c6d-8702-a4ab0622a731); Time taken: 25.525 seconds
```

- Query 4 is:

> Report on the reliability of devices – number of app restarts without device restart (aka app crashes).
> List top-10 for the combination of app version and device (if any)

HA 5: Optimize the fact table for Query 4

- Study possibilities of storage format and table partitioning to organize the Facts table.
  - Choose a way to organize the Fact table in Hive and comment on why.
- For Query 4, report then the number of mappers, reducers used as well as execution time.
- List the query plan by Hive and compare it with the plan by Pg

Submission of Assignment 4 & 5

- Hand in to the IS vault:
  - A txt file named `hive.txt`
  - It will contain
    - commands to migrate / instantiate DW in Hive,
      - and their execution stats (# mappers and reducers, execution time)
    - suggested organization of the Facts table – Hive command and reasoning,
      - execution stats of Query 4
    - compare query plans by Hive and Pg.
- Grading
  - HA 4 – total 10 pts
  - HA 5 – organization 5 pts; plan comparison 5 pts

How to Access Hive

- MetaCentrum - https://wiki.metacentrum.cz/wiki/Hadoop
  - Register first (unless you already have an account)
    - https://metavo.metacentrum.cz/cs/application/index.html
    - Hadoop cluster access must be requsted then: https://www.metacentrum.cz/en/hadoop/
  - SW available:
    - Hadoop 3.0.0 - distributed storage and processing of very large data sets
    - HBase 2.1.0 - distributed, scalable, big data store
    - Hive 2.1.1 - data warehouse software facilitates
    - Hue 4.2.0 - Analytics Workbench for self-service (GUI)
    - Pig 0.17.0 - platform for analyzing large data sets
    - Spark 2.4.0 - fast and general engine for large-scale data processing

Login to Hadoop Frontend in MetaCentrum

- SSH to hador.ics.muni.cz, preferably from Aisa
- Add the lines to your local ssh config: ~/.ssh/config (or C:\Users\<login>\.ssh\config)

```
## MetaCentrum ##########
Host hador
  HostName hador.ics.muni.cz
  User <your_login_in_MetaCentrum>
  Port 22
```

- Log in to Hadoop Cluster frontend:

```
$ ssh hador
```

Login to Hive

Execute in shell on hador

```
$ DBNAME="pa220_`id -un`"
$ JDBC_URL="jdbc:hive2://hador-c1.ics.muni.cz:10000/$DBNAME;principal=hive/hador-c1.ics.muni.cz@ICS.MUNI.CZ"

$ beeline -u $JDBC_URL -e "CREATE DATABASE $DBNAME"

$ beeline -u $JDBC_URL
```

```
0: jdbc:hive2://hador-c1.ics.muni.cz:10000/pa> show tables;
INFO  : Compiling command(queryId=hive_20210121132209_3aea8536-1395-413e-b186-f969f38f20b4): show tables
…
INFO  : OK
+--------------------+
|      tab_name      |
+--------------------+
| dim_app            |
| dim_device         |
| ext_data_sept_oct  |
| facts              |
| pokes              |
+--------------------+
5 rows selected (0.433 seconds)
```

Execute this line just once to create your own database.
All your tables will be stored in HDFS:
hdfs dfs –ls /user/hive/warehouse/$DBNAME.db/

Pg Data Migration

- Export xdohnal.conn_log and xdohnal.service_log
  - OR xdohnal.pa220ha1dataseptoct
    - this table contains the joined data from the solution to HA1
    - Its CSV is available in HDFS: `/user/dohnal/pa220ha1-data-sept-oct-2020.csv`
      `hdfs dfs -ls /user/dohnal/pa220ha1-data-sept-oct-2020.csv`
      you may copy it to your HDFS home:
      `hdfs dfs -cp /user/dohnal/pa220ha1-data-sept-oct-2020.csv .`
  - Export by PgAdmin 4 web app:
    - Right click on the table, choose Backup…
    - Fill in the file to store the table's contents; Format: Plain; Dump Options: Only Data (YES)
    - Edit the resulting file to delete all non-data lines, so it is a CSV file and can be loaded by Hive

```
--
-- PostgreSQL database dump
--
...
COPY xdohnal. pa220ha1dataseptoctitem (item_id, item_name) FROM stdin;
A56    867721024631452 Lenovo PB1-750M 3579    2020-09-01 05:32:00    2020-09-01 05:33:00    0.33    0.33    4...
...
\.
-- Completed on 2021-01-21 13:31:39
--
-- PostgreSQL database dump complete
--
```

Load CSV into Hive

- Create an external table in your HDFS home:

```
CREATE EXTERNAL TABLE ext_data_sept_oct (
    program_ver string,
    pda_imei string,
    device string,
    car_key bigint,
    sl_time timestamp,
    cl_time timestamp,
    app_run_time decimal(6,2),
    pda_run_time decimal(10,2),
    tracking_mode string,
    battery_level string,
    sim_imsi string,
    gsmnet_id string,
    method string
 )
 ROW FORMAT DELIMITED FIELDS TERMINATED BY '\011'
 STORED AS TEXTFILE
 LOCATION '/user/<login>/pa220_data_sept_oct';
```

The <login> placeholder should be replaced with your MetaCentrum login.

- Load the data (Hive will move the file):

```
LOAD DATA INPATH '/user/<login>/pa220ha1-data-sept-oct-2020.csv' INTO TABLE ext_data_sept_oct;
-- Verify that the data is available - the query should return 3912168
select count(*) from ext_data_sept_oct;
```