5th March 2018

PIG (Cont...)

1. JSON Data Processing through PIG

Data.json <key : value>

{"Website" : "Amazon", "location" : "California", "rating" : 1,
"amount" : 500000}

- - -
- - -

JSONScript.pig (same path pig script)

```
initialData = LOAD '/home/gopalkrishna/Batch89/PIGPRAC/
              Data.json' USING
              JsonLoader('website : chararray, location :
              chararray, rating : int, amount : int');

filterData = FOREACH initialData GENERATE Website, location,
             amount;
selectedData = FILTER initialData BY rating > 15;
sortedData = ORDER filterData BY location;
STORE filterData INTO 'JSONFIL-DATA89';
STORE selectedData INTO 'JSONSEL-DATA89';
STORE sortedData INTO 'JSONSORT-DATA89';
```

NOTE :- Not to refer any jar file support.
Keys must be same as defined in JSON Keys, no change in case.

$ ——— /PIGPRAC $ pig -x local JSONScript.pig ↵

2. SPECIAL JOINS IN PIG

Inorder to improvise the performance of JOIN operations, below are the different joins available

a) Replicated join
b) merge join
c) skewed join

Note :- Merge and Replicated joins are included in the Mapper phase.

skewed joins are included in Reducer phase.

a. Replicated join

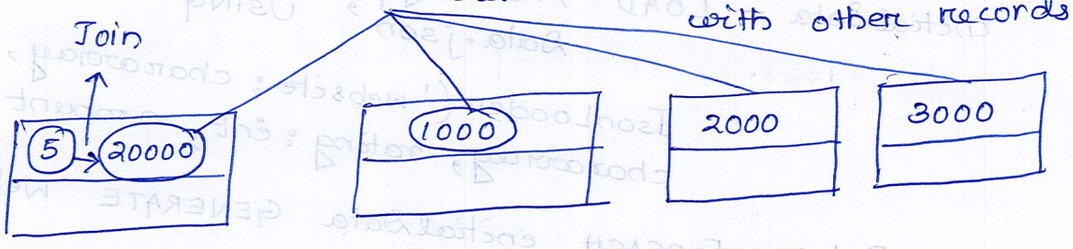
Let's say:

example: part-m-00000

5 records (Id: int) — 85000 records (Id: int)

We can perform the join, but the join operation suffers from performance problems.

example:-



For such case we use REPLICATED JOIN.

cipAdress	Country
5 records	

cipAdress	Country	Capital	Website	Code
63456 records				

3. Embedded Mode of PIG

Steps for using PIG UDFs:-

a) Develop a class by extending from the base class of EvalFunc<String>.

The package need to be imported is as below:

org.apache.pig.EvalFunc

b) To write our own customizable logic we must override below method:-

public String exec (Tuple input) throws IOException

c) We need to add required PIG jar file to the build path and create our own logic with business logic, which needs to be exported to the Hadoop environment where PIG is running.

d. Refer the same jar file in the main pig file using 'register' keyword.

→ register statement should be in first line of pig script.

** END **

6th March 2018

PIG UDFs (cont...)

A. Use case 1 :— cisco Data log file ciscodata.log

Example :→ (tab separated columns) 10101 | Network Related Issue | sanjose | 2011-11-10 10:10:10

10201 | Router Related | Rose City | 2012-12-09 09:10:27

ciscoUdfscript.pig

```
REGISTER /home/gopalkrishna/Batch88/PIGPRAC/PIG-CONCAT.jar;

A = LOAD 'ciscodata.log' USING PigStorage('|') AS
    (logid : int, logerror : chararray, logdate : chararray, logloc : chararray);

B = FOREACH A GENERATE logid, com.pig.concatusecase.
    CONCAT (logerror), logloc ;

STORE B INTO 'ciscoUdfOUT89';
```

→ $ pig -x local ciscoUdfScript.pig ↵

B. Use case 2 :— EpochTime

19245689 → 1970-10-12 : 09
sec — YYYY MM DD HH

EpochScript.pig

```
REGISTER /home/gopalkrishna/Batch88/PIGPRAC/PIG-EPOCH.jar;
eneData = LOAD 'epochTimeData.log' AS
          (timeData : chararray);
convertData = FOREACH eneData GENERATE timeData,
          com.pig.epochusecase.EPOCH(timeData);
```

```
STORE convert_Data INTO 'EPOCHOUT89';
```

```
$ pig -x local EpochScript.pig ↵
```

```
** END **
```