

# PIG TUTORIAL

---

Hands-on Session

by Suchitra Jayaprakash  
suchitra@cmi.ac.in

# Apache PIG

- PIG is an abstraction over MapReduce.
- It uses high level language - Pig Latin.
- PIG Latin is easy to learn if programmer knows procedural language. Programmer can do MapReduce task without having to write complex Java code.
- Pig has many built-in functions & data types for doing operations like joins, filters, ordering, grouping. Thus reducing code length.
- Apache PIG has a internal component called PIG engine.
- PIG engine converts Pig Latin scripts to Map Reduce Task.
- Apache Pig was developed as a research project at Yahoo in 2006. First release of Apache Pig came out in 2008.

# Apache PIG – Run Mode

- Pig can be run in various modes:
  - Local mode : Run PIG in local host and file system
  - MapReduce mode : Run PIG on Hadoop cluster and HDFS. It is the default mode.
  - Interactive mode : Run Pig using Grunt shell.
  - Batch mode : Run Pig using PIG script.

	Interactive Mode	Batch Mode
Local Mode	<pre>\$ pig -x local grunt&gt; &lt;pig Latin statement&gt;</pre>	<pre>\$ pig -x local ScriptFile.pig</pre>
MapReduce Mode	<pre>\$ pig grunt&gt; &lt;pig Latin statement&gt;</pre>	<pre>\$ pig ScriptFile.pig</pre>
	<pre>\$ pig -x mapreduce grunt&gt; &lt;pig Latin statement&gt;</pre>	<pre>\$ pig -x mapreduce ScriptFile.pig</pre>

# PIG Latin Statement

- Pig Latin statement structure is:
- **LOAD** statement to read data from the file system.
- **Transformation** statements to process the data.
- **DUMP** statement to view results or **STORE** statement to save the results.
- **Load Operator**  
Relation\_name = LOAD 'Input file path' USING function as schema;

# PIG Latin Statement

## Transformation statements :

- [FILTER](#) operator to remove unwanted rows of data.
- [FOREACH](#) , [GENERATE](#) operator to perform data transformations on columns of data.
- [GROUP](#) operator to group data in a single relation.
- [COGROUP](#), [inner JOIN](#), and [outer JOIN](#) operators to group or join data in two or more relations.
- [ORDER](#) operator to sort data.
- [UNION](#) operator to merge the contents of two or more relations.
- [SPLIT](#) operator to partition the contents of a relation into multiple relations.

# Exercise

- Find average sepal length for each flower class in IRIS DATASET

- Reference :

<https://pig.apache.org/docs/r0.17.0/basic.html>

<https://pig.apache.org/docs/r0.17.0/func.html>

<https://pig.apache.org/docs/r0.17.0/udf.html>

# Run PIG

- **Start Cloudera server**

```
docker run --hostname=quickstart.cloudera --privileged=true -t -i --  
publish-all=true -p 8888:8888 -p 8080:80 -p 50070:50070 -p 8088:8088  
-p 50075:50075 -p 8032:8032 -p 8042:8042 -p 19888:19888  
cloudera/quickstart /usr/bin/docker-quickstart
```

- **Copy Text file to docker container**

```
docker cp E:/iris.txt <containerid>:/tmp/iris.txt
```

- **Copy Text file to HDFS**

```
hadoop fs -mkdir DATA
```

```
hadoop fs -copyFromLocal /tmp/iris.txt DATA/iris.txt
```

```
#view output
```

```
hdfs dfs -cat DATA/iris.txt
```

# Run PIG

- Start PIG
- type pig and press enter to get PIG command prompt

```
flower = LOAD 'DATA/iris.txt' USING PigStorage(',') as ( sepal_length:int,  
sepal_width:int, petal_length:int, petal_width:int, flower_class:chararray);
```

```
DUMP flower;
```

```
B = GROUP flower BY flower_class;  
DUMP B;
```

```
Result = FOREACH B GENERATE flower.flower_class,  
AVG(flower.sepal_length);
```

```
DUMP Result;
```



# OUTPUT

```
Success!

Job Stats (time in seconds):
JobId      Maps      Reduces  MaxMapTime      MinMapTime      AvgMapTime      MedianMa
pTime      MaxReduceTime  MinReduceTime  AvgReduceTime  MedianReductime  A
lias      Feature  Outputs
job_1581917396709_0001  1      0      47      47      47      47      n/a      n
/a      n/a      n/a      flower  MAP_ONLY      hdfs://quickstart.cloudera:8020/
tmp/tmp1319724132/tmp-899714232,

Input(s):
Successfully read 151 records (4925 bytes) from: "hdfs://quickstart.cloudera:8020/user/root/DATA/iris.txt"

Output(s):
Successfully stored 151 records (4088 bytes) in: "hdfs://quickstart.cloudera:8020/tmp/tmp1319724132/tmp-899714232"

Counters:
Total records written : 151
Total bytes written : 4088
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_1581917396709_0001

2020-02-17 05:50:51,141 [main] WARN  org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MapReduceLauncher - Encountered Warning ACCESSING_NON_EXISTENT
_FIELD 4 time(s).
2020-02-17 05:50:51,144 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MapReduceLauncher - Success!
2020-02-17 05:50:51,161 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - fs.default.name is deprecated. Instead, use fs.defaultFS
2020-02-17 05:50:51,164 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.addr
ess
2020-02-17 05:50:51,169 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Ke
y [pig.schematuple] was not set... will not generate code.
2020-02-17 05:50:51,235 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileI
nputFormat - Total input paths to process is 1
```

# OUTPUT

```
(6,3,5,2,Iris-virginica)
(6,3,5,1,Iris-virginica)
(6,3,4,1,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,3,5,2,Iris-virginica)
(5,2,5,1,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,2,5,1,Iris-virginica)
(6,3,5,2,Iris-virginica)
(6,3,5,2,Iris-virginica)
(5,3,5,1,Iris-virginica)
(,,,,)
grunt> B = GROUP flower BY flower_class;
2020-02-17 05:53:12,148 [main] INFO org.apache.hadoop.conf.Configuration.deprec
ation - fs.default.name is deprecated. Instead, use fs.defaultFS
2020-02-17 05:53:12,150 [main] INFO org.apache.hadoop.conf.Configuration.deprec
ation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.addr
ess
grunt> Result = FOREACH B GENERATE flower.flower_class, AVG<flower.sepal_length>
;
grunt> DUMP Result;
2020-02-17 05:53:42,813 [main] INFO org.apache.pig.tools.pigstats.ScriptState -
Pig features used in the script: GROUP_BY
2020-02-17 05:53:42,826 [main] INFO org.apache.pig.newplan.logical.optimizer.Lo
gicalPlanOptimizer - {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, DuplicateFor
EachColumnRewrite, GroupByConstParallelSetter, ImplicitSplitInserter, LimitOptim
izer, LoadTypeCastInserter, MergeFilter, MergeForEach, NewPartitionFilterOptimiz
er, PushDownForEachFlatten, PushUpFilter, SplitFilter, StreamTypeCastInserter],
RULES_DISABLED=[FilterLogicExpressionSimplifier, PartitionFilterOptimizer]}
2020-02-17 05:53:42,909 [main] INFO org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MRCompiler - File concatenation threshold: 100 optimistic? fal
se
2020-02-17 05:53:42,959 [main] INFO org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MultiQueryOptimizer - MR plan size before optimization: 1
2020-02-17 05:53:42,962 [main] INFO org.apache.pig.backend.hadoop.executionengi
ne.mapReduceLayer.MultiQueryOptimizer - MR plan size after optimization: 1
2020-02-17 05:53:43,080 [main] INFO org.apache.hadoop.yarn.client.RMPProxy - Con
necting to ResourceManager at /0.0.0.0:8032
2020-02-17 05:53:43,097 [main] INFO org.apache.pig.tools.pigstats.ScriptState -
Pig script settings are added to the job
2020-02-17 05:53:43,190 [main] INFO org.apache.pig.backend.hadoop.executionengi
```

# OUTPUT

[illegible]

# USER DEFINED FUNCTION

- Find Volume Weighted Average Price of a stock per year.

Date	Adj Close	Close	High	Low	Open	Volume
31-12-2007	92.64	92.64	94.37	92.45	93.81	5755200
02-01-2008	96.25	96.25	97.43	94.7	95.35	13858700
03-01-2008	95.21	95.21	97.25	94.52	96.06	9122500
04-01-2008	88.79	88.79	93.4	88.5	93.26	10270000

$$\text{VWAP} = \frac{\text{sum}(\text{Price} * \text{Volume})}{\text{sum}(\text{Volume})}$$

It is a measure of the volume weighted average price at which a stock is traded over the trading horizon.

# USER DEFINED FUNCTION

- **Find Volume Weighted Average Price of a stock**

Copy content to docker:

```
docker cp C:/pig/stock.csv <containerid>:/tmp/stock.csv
```

```
docker cp C:/pig/pig_UDF.py <containerid>:/tmp/pig_UDF.py
```

```
docker cp C:/pig/piggybank-0.15.0.jar <containerid>:/tmp/piggybank-0.15.0.jar
```

```
hadoop fs -mkdir DATA
```

```
hadoop fs -copyFromLocal /tmp/stock.csv DATA/stock.csv
```

# USER DEFINED FUNCTION

Execute in PIG Interface:

```
REGISTER '/tmp/pig_UDF.py' using jython as myudfs;  
REGISTER '/tmp/piggybank-0.15.0.jar';
```

```
records = LOAD 'DATA/stock.csv' USING  
org.apache.pig.piggybank.storage.CSVExcelStorage(',', 'NO_MULTILINE', 'UNIX',  
'SKIP_INPUT_HEADER') AS  
(Date:datetime,AdjClose:int,Close:int,High:int,Low:int,Open:int,Volume:int);
```

```
record= foreach records GENERATE Volume, Close, GetYear(Date) as Year;
```

```
yearData = GROUP record BY Year;
```

```
vol_wt_avg_price = FOREACH yearData GENERATE  
myudfs.calVWAP(record.Year,record.Volume,record.Close);
```

```
STORE vol_wt_avg_price INTO '/user/root/PRICE';
```

```
hadoop fs -cat /user/root/PRICE/part-r-00000
```

# OUTPUT

```
ne.mapReduceLayer.MapReduceLauncher - Success!
2020-02-19 00:46:09,260 [main] INFO org.apache.hadoop.conf.Configuration.deprec
ation - fs.default.name is deprecated. Instead, use fs.defaultFS
2020-02-19 00:46:09,266 [main] INFO org.apache.hadoop.conf.Configuration.deprec
ation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.addr
ess
2020-02-19 00:46:09,279 [main] INFO org.apache.pig.data.SchemaTupleBackend - Ke
y [pig.schematuple] was not set... will not generate code.
2020-02-19 00:46:09,465 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileI
nputFormat - Total input paths to process : 1
2020-02-19 00:46:09,472 [main] INFO org.apache.pig.backend.hadoop.executionengi
ne.util.MapRedUtil - Total input paths to process : 1
(<2007,92>)
(<2008,67>)
(<2009,87>)
(<2010,135>)
(<2011,197>)
(<2012,214>)
(<2013,295>)
(<2014,329>)
(<2015,485>)
(<2016,680>)
(<2017,969>)
(<2018,1624>)
grunt>
```

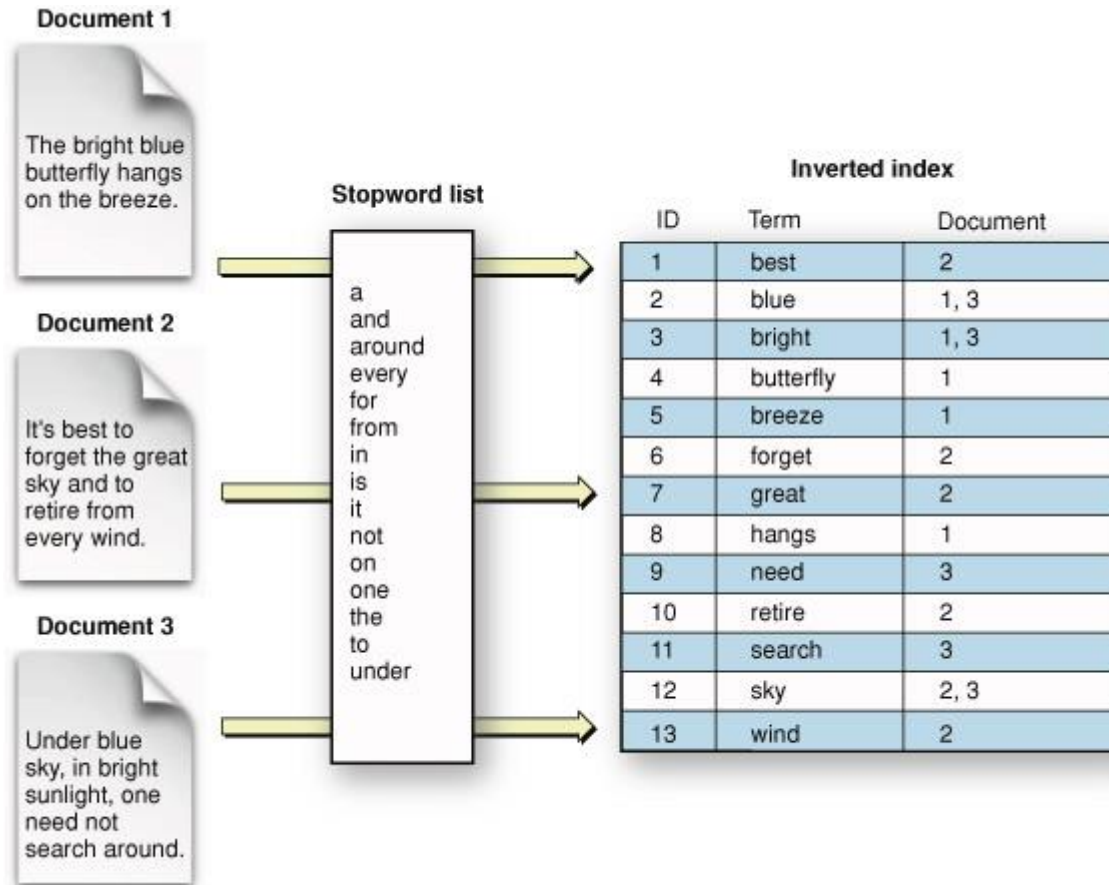
# MAPREDUCE

---

Few Examples



# Inverted Index with MapReduce



# Implementation

- `WORD_RE = re.compile(r"[a-zA-Z]{2,}\b")`
- `class MRInvertedIndex(MRJob):`
  - `def mapper(self, _, line):`
    - `## getting input file name`
    - `filepath = os.environ['map_input_file']`
    - `filename=filepath.split('/')[-1]`
    - `for word in WORD_RE.findall(line):`
      - `yield (word.lower(), filename)`
  - `def reducer(self, word, filenames):`
    - `yield (word, ",".join(list(set(filenames))))`

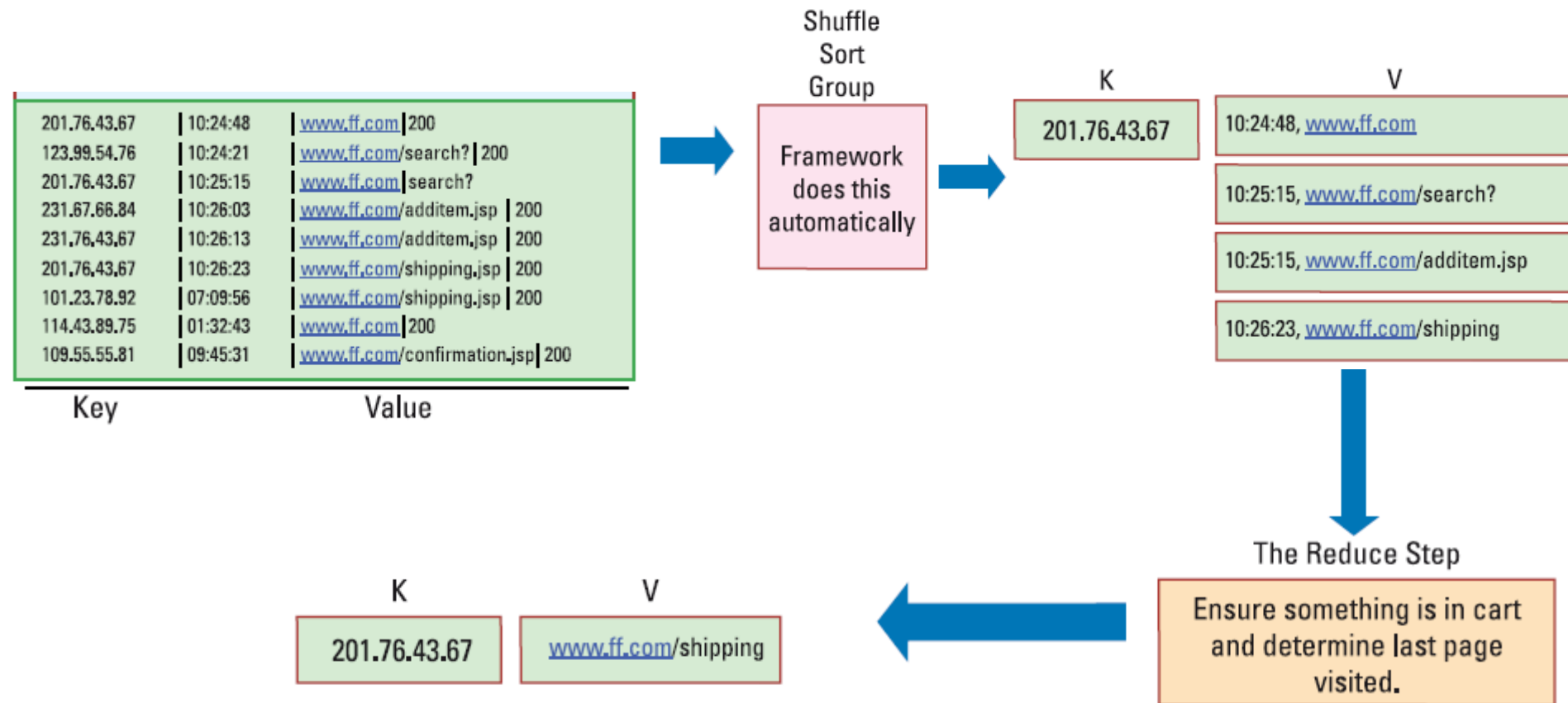
# Capture useful insights from Log data

- An ecommerce web site collects click stream data as log file.

201.76.43.67	10:24:48	<a href="http://www.ff.com">www.ff.com</a>   200
123.99.54.76	10:24:21	<a href="http://www.ff.com/search?">www.ff.com/search?</a>   200
201.76.43.67	10:25:15	<a href="http://www.ff.com">www.ff.com</a>   search?
231.67.66.84	10:26:03	<a href="http://www.ff.com/additem.jsp">www.ff.com/additem.jsp</a>   200
231.76.43.67	10:26:13	<a href="http://www.ff.com/additem.jsp">www.ff.com/additem.jsp</a>   200
201.76.43.67	10:26:23	<a href="http://www.ff.com/shipping.jsp">www.ff.com/shipping.jsp</a>   200
101.23.78.92	07:09:56	<a href="http://www.ff.com/shipping.jsp">www.ff.com/shipping.jsp</a>   200
114.43.89.75	01:32:43	<a href="http://www.ff.com">www.ff.com</a>   200
109.55.55.81	09:45:31	<a href="http://www.ff.com/confirmation.jsp">www.ff.com/confirmation.jsp</a>   200

- Identify key factors behind abandoned shopping carts

# MapReduce Implementation



THANK YOU