



APACHE PIG



# Why Pig?

- Writing mappers and reducers by hand takes a long time.
- Pig introduces *Pig Latin*, a scripting language that lets you use SQL-like syntax to define your map and reduce steps.
- Highly extensible with user-defined functions (UDF's)





MapReduce



YARN

HDFS

# Running Pig

- Grunt
- Script
- Ambari / Hue



# An example

- Find the oldest 5-star movies



```
ratings = LOAD '/user/maria_dev/ml-100k/u.data' AS  
  (userID:int, movieID:int, rating:int, ratingTime:int);
```

This creates a *relation* named “ratings” with a given *schema*.

```
(660,229,2,891406212)  
(421,498,4,892241344)  
(495,1091,4,888637503)  
(806,421,4,882388897)  
(676,538,4,892685437)  
(721,262,3,877137285)
```



# Use PigStorage if you need a different delimiter.

```
metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING
            PigStorage('|')AS (movieID:int, movieTitle:chararray,
                               releaseDate:chararray, videoRelease:chararray,
                               imdbLink:chararray) ;

DUMP metadata;
```

```
(1,Toy Story (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?Toy%20Story%20(1995))
(2,GoldenEye (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?GoldenEye%20(1995))
(3,Four Rooms (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?Four%20Rooms%20(1995))
(4,Get Shorty (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?Get%20Shorty%20(1995))
(5,Copycat (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?Copycat%20(1995))
```

# Creating a relation from another relation; FOREACH / GENERATE

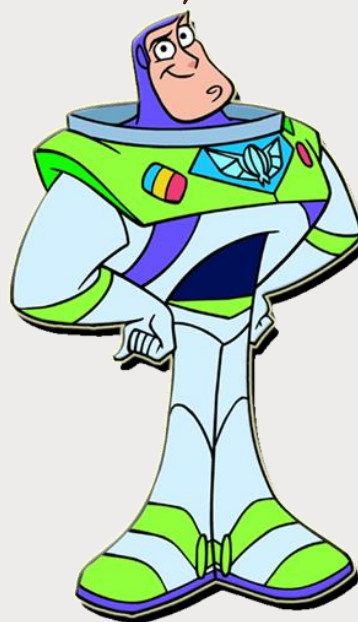
```
metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING PigStorage('|')
           AS (movieID:int, movieTitle:chararray, releaseDate:chararray,
              videoRelease:chararray, imdbLink:chararray);
```

```
nameLookup = FOREACH metadata GENERATE movieID, movieTitle,
           ToUnixTime(ToDate(releaseDate, 'dd-MMM-yyyy')) AS releaseTime;
```

```
(1,Toy Story (1995),01-Jan-1995,,http://us.imdb.com/M/title-exact?Toy%20Story%20(1995))
```



```
(1,Toy Story (1995),788918400)
```





# Group By

```
ratingsByMovie = GROUP ratings BY movieID;  
DUMP ratingsByMovie;
```

```
(1, {(807,1,4,892528231),(554,1,3,876231938),(49,1,2,888068651), ... }  
(2, {(429,2,3,882387599),(551,2,2,892784780),(774,2,1,888557383), ... }
```

ratingsByMovie: {group: int, ratings: {(userID: int, movieID: int, rating: int, ratingTime: int)}}

```
avgRatings = FOREACH ratingsByMovie GENERATE group AS movieID,  
      AVG(ratings.rating) AS avgRating;  
  
DUMP avgRatings;
```

```
(1,3.8783185840707963)  
(2,3.2061068702290076)  
(3,3.0333333333333333)  
(4,3.550239234449761)  
(5,3.302325581395349)
```

```
DESCRIBE ratings;  
DESCRIBE ratingsByMovie;  
DESCRIBE avgRatings;
```

```
ratings: {userID: int, movieID: int, rating: int, ratingTime: int}
```

```
ratingsByMovie: {group: int, ratings: {(userID: int, movieID: int, rating: int, ratingTime: int)}}
```

```
avgRatings: {movieID: int, avgRating: double}
```

# FILTER

```
fiveStarMovies = FILTER avgRatings BY avgRating > 4.0;
```

```
(12,4.385767790262173)  
(22,4.151515151515151)  
(23,4.1208791208791204)  
(45,4.05)
```

# JOIN

```
DESCRIBE fiveStarMovies;
```

```
DESCRIBE nameLookup;
```

```
fiveStarsWithData = JOIN fiveStarMovies BY movieID, nameLookup BY movieID;
```

```
DESCRIBE fiveStarsWithData;
```

```
DUMP fiveStarsWithData;
```

```
fiveStarMovies: {movieID: int,avgRating: double}
```

```
nameLookup: {movieID: int,movieTitle: chararray,releaseTime: long}
```

```
fiveStarsWithData: {fiveStarMovies::movieID: int,fiveStarMovies::avgRating: double,  
                    nameLookup::movieID: int,nameLookup::movieTitle: chararray,nameLookup::releaseTime: long}
```

```
(12,4.385767790262173,12,Usual Suspects, The (1995),808358400)
```

```
(22,4.151515151515151,22,Braveheart (1995),824428800)
```

```
(23,4.1208791208791204,23,Taxi Driver (1976),824428800)
```

# ORDER BY

```
oldestFiveStarMovies = ORDER fiveStarsWithData BY  
    nameLookup::releaseTime;
```

```
DUMP oldestFiveStarMovies;
```

```
(493,4.15,493,Thin Man, The (1934),-1136073600)  
(604,4.012345679012346,604,It Happened One Night (1934),-1136073600)  
(615,4.0508474576271185,615,39 Steps, The (1935),-1104537600)  
(1203,4.0476190476190474,1203,Top Hat (1935),-1104537600)
```



# Putting it all together

```
ratings = LOAD '/user/maria_dev/ml-100k/u.data' AS (userID:int, movieID:int, rating:int, ratingTime:int);

metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING PigStorage('|')
  AS (movieID:int, movieTitle:chararray, releaseDate:chararray, videoRelease:chararray, imdbLink:chararray);

nameLookup = FOREACH metadata GENERATE movieID, movieTitle,
  ToUnixTime(ToDate(releaseDate, 'dd-MMM-yyyy')) AS releaseTime;

ratingsByMovie = GROUP ratings BY movieID;

avgRatings = FOREACH ratingsByMovie GENERATE group AS movieID, AVG(ratings.rating) AS avgRating;

fiveStarMovies = FILTER avgRatings BY avgRating > 4.0;

fiveStarsWithData = JOIN fiveStarMovies BY movieID, nameLookup BY movieID;

oldestFiveStarMovies = ORDER fiveStarsWithData BY nameLookup::releaseTime;

DUMP oldestFiveStarMovies;
```



Let's run it



# Pig Latin: Diving Deeper

## Things you can do to a relation

- LOAD STORE DUMP
  - *STORE ratings INTO 'outRatings' USING PigStorage(':');*
- FILTER DISTINCT FOREACH/GENERATE MAPREDUCE STREAM SAMPLE
- JOIN COGROUP GROUP CROSS CUBE
- ORDER RANK LIMIT
- UNION SPLIT

# Diagnostics

- DESCRIBE
- EXPLAIN
- ILLUSTRATE

# UDF's

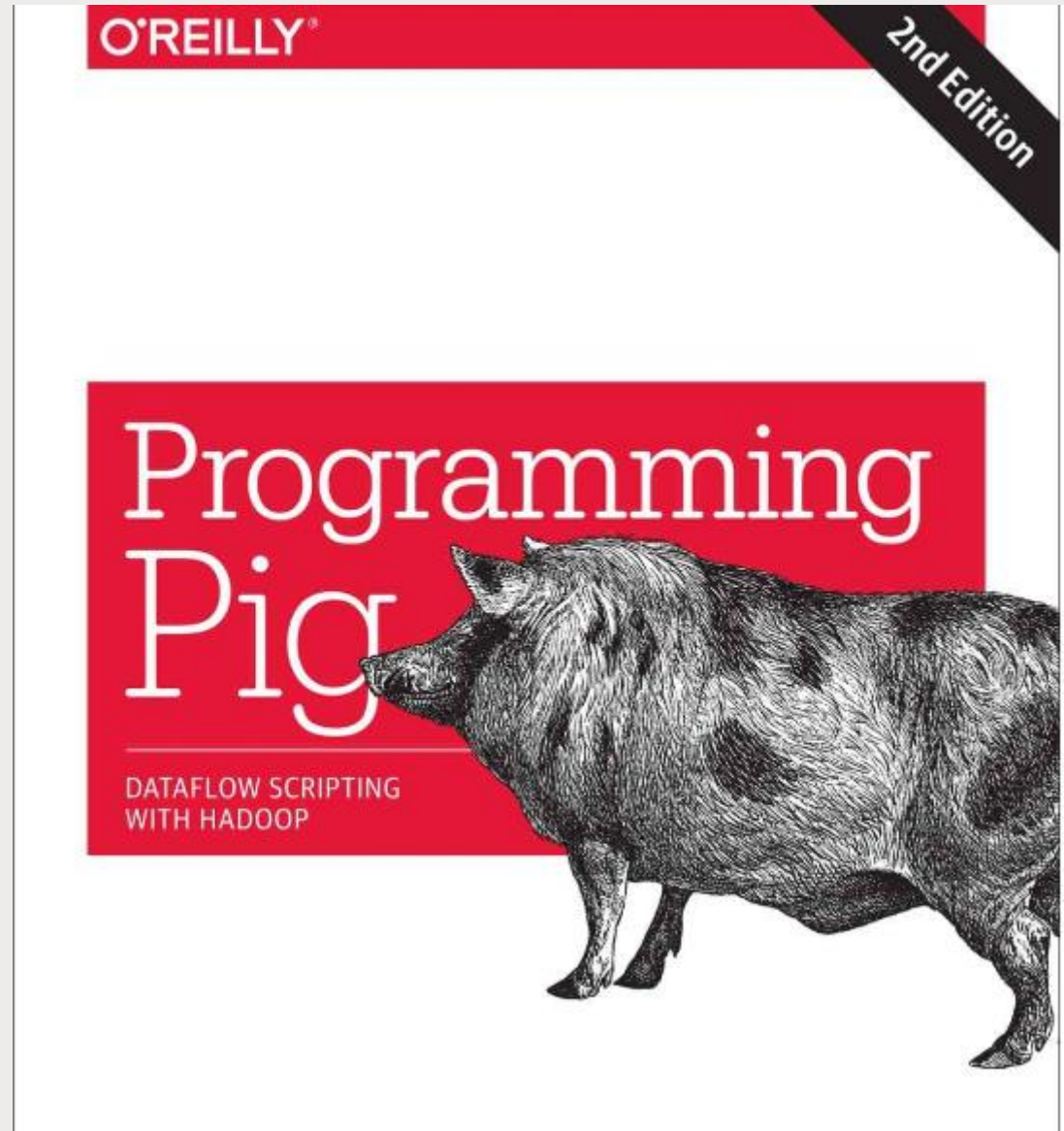
- REGISTER
- DEFINE
- IMPORT

# Some other functions and loaders

- AVG   CONCAT   COUNT   MAX   MIN   SIZE   SUM

- PigStorage
- TextLoader
- JsonLoader
- AvroStorage
- ParquetLoader
- OrcStorage
- HBaseStorage

# Learning more







# PIG CHALLENGE

Find the most popular bad movies



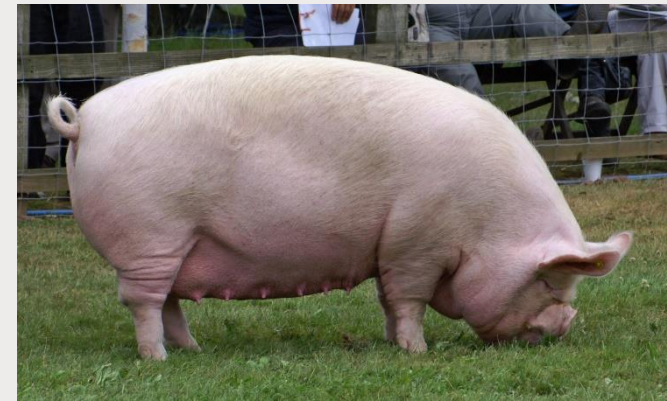
# Defining the problem

- Find all movies with an average rating less than 2.0
- Sort them by the total number of ratings



# Hint

- We used everything you need in our earlier example of finding old movies with ratings greater than 4.0
- Only new thing you need is COUNT(). This lets you count up the number of items in a bag.
  - *So just like you can say AVG(ratings.rating) to get the average rating from a bag of ratings,*
  - *You can say COUNT(ratings.rating) to get the total number of ratings for a given group's bag.*





HDFS



YARN



MapReduce2



Tez



Hive



HBase



Pig



Sqoop



Oozie



ZooKeeper



Falcon



Storm



Flume



Ambari Infra



Ambari Metrics



Atlas



Kafka



Knox



Ranger



Spark



Spark2



Zeppelin

Notebook



Slider

Summary

Configs

Service Actions

Group

Default (1)



Manage Config Groups

Filter...



V1

admin

23 days ago

HDP-2.5



V1



admin authored on Tue, Oct 25, 2016 03:26

Discard

Save

Advanced pig-env

Advanced pig-log4j

Advanced pig-properties



## Service checks completed.



Storage test



HDFS test



WebHCat test



User Home Directory test

YARN Queue Manager

Files View

Hive View

Pig View

Storm View

Tez View



Scripts



UDFs



History

Scripts

## New Script



Name

Script HDFS Location (optional)

Leave empty to create file automatically.

Cancel

Create

+ New Script

Actions

[History](#) [Copy](#) [Delete](#)[History](#) [Copy](#) [Delete](#)[History](#) [Copy](#) [Delete](#)

Show:

10

1 - 3 of 3







Oldest five-star movie

☐ Execute on Tez

Execute

PIG helper

UDF helper

/tmp/.pigscripits/mostrated\_one\_star\_movie\_copy-2016-11-15\_05-09.pig

```
1 ratings = LOAD '/user/maria_dev/ml-100k/u.data' AS (userID:int, movieID:int, rating:int, ratingTime:int);
2
3 metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING PigStorage('|')
4           AS (movieID:int, movieTitle:chararray, releaseDate:chararray, videoRelease:chararray, imdbLink:chararray);
5
6 nameLookup = FOREACH metadata GENERATE movieID, movieTitle,
7             ToUnixTime(ToDate(releaseDate, 'dd-MMM-yyyy')) AS releaseTime;
8
9 ratingsByMovie = GROUP ratings BY movieID;
10
11 avgRatings = FOREACH ratingsByMovie GENERATE group AS movieID, AVG(ratings.rating) AS avgRating;
12
13 fiveStarMovies = FILTER avgRatings BY avgRating > 4.0;
14
15 fiveStarsWithData = JOIN fiveStarMovies BY movieID, nameLookup BY movieID;
16
17 oldestFiveStarMovies = ORDER fiveStarsWithData BY nameLookup::releaseTime;
18
19 DUMP oldestFiveStarMovies;
```

Toggle  
fullscreen  
(F11)



Oldest five-star movie



Save



Copy



Delete

Script

History

Oldest five-star movie - Running

Oldest five-star movie - **RUNNING**

Kill Job

Job ID job\_1479310897382\_0024

Started 2016-11-16 15:49

Results

Logs

Script Details

Script

History

Oldest five-star movie - Completed

Oldest five-star movie

Save

Copy

Delete

Script

History

Oldest five-star movie - Completed

Oldest five-star movie - COMPLETED

Job ID

job\_1479310897382\_0024

Started

2016-11-16 15:49

Results

Download

(493,4.15,493,Thin Man, The (1934),-1136073600)

(604,4.012345679012346,604,It Happened One Night (1934),-1136073600)

(615,4.0508474576271185,615,39 Steps, The (1935),-1104537600)

(1203,4.0476190476190474,1203,Top Hat (1935),-1104537600)

(613,4.037037037037037,613,My Man Godfrey (1936),-1073001600)

(633,4.057971014492754,633,Christmas Carol, A (1938),-1009843200)

(136,4.123809523809523,136,Mr. Smith Goes to Washington (1939),-978307200)

(1122,5.0,1122,They Made Me a Criminal (1939),-978307200)

(132,4.0772357723577235,132,Wizard of Oz, The (1939),-978307200)

(524,4.021739130434782,524,Great Dictator, The (1940),-946771200)

(478,4.115384615384615,478,Philadelphia Story, The (1940),-946771200)

(484,4.2101449275362315,484,Maltese Falcon, The (1941),-915148800)

(134,4.292929292929293,134,Citizen Kane (1941),-915148800)

(483,4.45679012345679,483,Casablanca (1942),-883612800)

(611,4.1,611,Laura (1944),-820540800)

(659,4.078260869565217,659,Arsenic and Old Lace (1944),-820540800)

(525,4.027397260273973,525,Big Sleep, The (1946),-757382400)

(489,4.115384615384615,489,Notorious (1946),-757382400)

(496,4.121212121212121,496,It's a Wonderful Life (1946),-757382400)

(1064,4.25,1064,Crossfire (1947),-725846400)

(519,4.1,519,Treasure of the Sierra Madre, The (1948),-694310400)

(513,4.333333333333333,513,Third Man, The (1949),-662688000)

(488,4.2,488,Sunset Blvd. (1950),-631152000)

(606,4.045454545454546,606,All About Eve (1950),-631152000)

(498,4.184210526315789,498,African Queen, The (1951),-599616000)

(648,4.029850746268656,648,Quiet Man, The (1952),-568080000)

(661,4.1022727272727275,661,High Noon (1952),-568080000)

(487,4.102941176470588,487,Roman Holiday (1953),-536457600)


(603,4.3875598086124405,603,Rear Window (1954),-504921600)

(490,4.02.490.To Catch a Thief (1955),-473385600)

Save

Copy

Delete

Oldest five-star movie  Execute on Tez

Execute

PIG helper ▾

UDF helper ▾

/tmp/.pigscripts/oldest\_five\_star\_movie\_copy-2016-11-15\_05-09.pig

```
1 ratings = LOAD '/user/maria_dev/ml-100k/u.data' AS (userID:int, movieID:int, rating:int, ratingTime:int);
2
3 metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING PigStorage('|')
4           AS (movieID:int, movieTitle:chararray, releaseDate:chararray, videoRelease:chararray, imdbLink:chararray);
5
6 nameLookup = FOREACH metadata GENERATE movieID, movieTitle,
7           ToUnixTime(ToDate(releaseDate, 'dd-MMM-yyyy')) AS releaseTime;
8
9 ratingsByMovie = GROUP ratings BY movieID;
10
11 avgRatings = FOREACH ratingsByMovie GENERATE group AS movieID, AVG(ratings.rating) AS avgRating;
12
13 fiveStarMovies = FILTER avgRatings BY avgRating > 4.0;
14
15 fiveStarsWithData = JOIN fiveStarMovies BY movieID, nameLookup BY movieID;
16
17 oldestFiveStarMovies = ORDER fiveStarsWithData BY nameLookup::releaseTime;
18
19 DUMP oldestFiveStarMovies;
20
21
22
```

Save

Copy

Delete

Oldest five-star movie ☒ Execute on TezExecute 


PIG helper ▾

UDF helper ▾

/tmp/.pigscripsts/mostrated\_one\_star\_movie\_copy-2016-11-15\_05-09.pig

```
1 ratings = LOAD '/user/maria_dev/ml-100k/u.data' AS (userID:int, movieID:int, rating:int, ratingTime:int);
2
3 metadata = LOAD '/user/maria_dev/ml-100k/u.item' USING PigStorage('|')
4           AS (movieID:int, movieTitle:chararray, releaseDate:chararray, videoRelease:chararray, imdbLink:chararray);
5
6 nameLookup = FOREACH metadata GENERATE movieID, movieTitle,
7               ToUnixTime(ToDate(releaseDate, 'dd-MMM-yyyy')) AS releaseTime;
8
9 ratingsByMovie = GROUP ratings BY movieID;
10
11 avgRatings = FOREACH ratingsByMovie GENERATE group AS movieID, AVG(ratings.rating) AS avgRating;
12
13 fiveStarMovies = FILTER avgRatings BY avgRating > 4.0;
14
15 fiveStarsWithData = JOIN fiveStarMovies BY movieID, nameLookup BY movieID;
16
17 oldestFiveStarMovies = ORDER fiveStarsWithData BY nameLookup::releaseTime;
18
19 DUMP oldestFiveStarMovies;
20
21
22
```



 Save Copy DeleteOldest five-star movie - **RUNNING** Kill Job

Job ID job\_1479310897382\_0029

Started 2016-11-16 15:54

## ▼ Results

(493,4.15,493,Thin Man, The (1934),-1136073600)  
(604,4.012345679012346,604,It Happened One Night (1934),-1136073600)  
(615,4.0508474576271185,615,39 Steps, The (1935),-1104537600)  
(1203,4.0476190476190474,1203,Top Hat (1935),-1104537600)  
(613,4.037037037037037,613,My Man Godfrey (1936),-1073001600)  
(633,4.057971014492754,633,Christmas Carol, A (1938),-1009843200)  
(132,4.0772357723577235,132,Wizard of Oz, The (1939),-978307200)  
(1122,5.0,1122,They Made Me a Criminal (1939),-978307200)  
(136,4.123809523809523,136,Mr. Smith Goes to Washington (1939),-978307200)  
(478,4.115384615384615,478,Philadelphia Story, The (1940),-946771200)  
(524,4.021739130434782,524,Great Dictator, The (1940),-946771200)  
(484,4.2101449275362315,484,Maltese Falcon, The (1941),-915148800)  
(134,4.292929292929293,134,Citizen Kane (1941),-915148800)  
(483,4.45679012345679,483,Casablanca (1942),-883612800)  
(659,4.078260869565217,659,Arsenic and Old Lace (1944),-820540800)  
(611,4.1,611,Laura (1944),-820540800)  
(496,4.121212121212121,496,It's a Wonderful Life (1946),-757382400)  
(525,4.027397260273973,525,Big Sleep, The (1946),-757382400)  
(489,4.115384615384615,489,Notorious (1946),-757382400)  
(1064,4.25,1064,Crossfire (1947),-725846400)  
(519,4.1,519,Treasure of the Sierra Madre, The (1948),-694310400)  
(513,4.333333333333333,513,Third Man, The (1949),-662688000)  
(488,4.2,488,Sunset Blvd. (1950),-631152000)  
(606,4.045454545454546,606,All About Eve (1950),-631152000)  
(498,4.184210526315789,498,African Queen, The (1951),-599616000)