PROGRAMMING PIG

CKME 134 – BIG DATA ANALYTICS TOOLS

RYERSON UNIVERSITY

SPRING 2015

Instructor: Shaohua Zhang

Course Outline

1. Intro to Big Data
2. Distributed Computing and MapReduce
3. Hadoop Ecosystem
4. Programming Hive
5. Advanced Hive
6. Mid-Term Review
7. Programming Pig
8. Advanced Pig
9. Hadoop Performance Optimization
10. Hadoop In Action: Building Data Pipelines
9. Location Analytics and Recommender Systems
11. Beyond Hadoop: Spark
12. Beyond Hadoop: Graph Analytics

Announcements

on behalf of program director Anne-Marie

- Spring/Summer courses registration
  - Few spots left, need to register asap
  - CKME 105 – only 3 spots left
- Course shell materials
  - Course shell won't be available a week after the course ends
  - If you need to keep a copy of the notes, assignments etc., make sure you download/print for your future reference

More Regex!

4

```
(.*)@USER_(\w{8})(.*)
```

```
RT @USER_2FF4FACA: IF SHE DO IT 1 MORE TIME......IMA KNOCK HER DAMN KOOFIE OFF.....ON MY MOMMA&GT;&GT;HAHA. #CUTTHATOUT
```

```
\.{7}
```

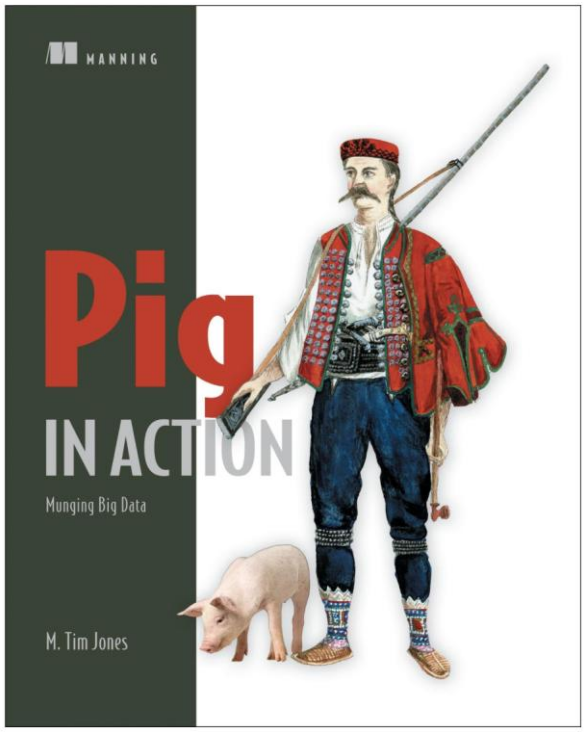
```
RT @USER_2FF4FACA: IF SHE DO IT 1 MORE TIME......IMA KNOCK HER DAMN KOOFIE OFF.....ON MY MOMMA&GT;&GT;HAHA. #CUTTHATOUT
```

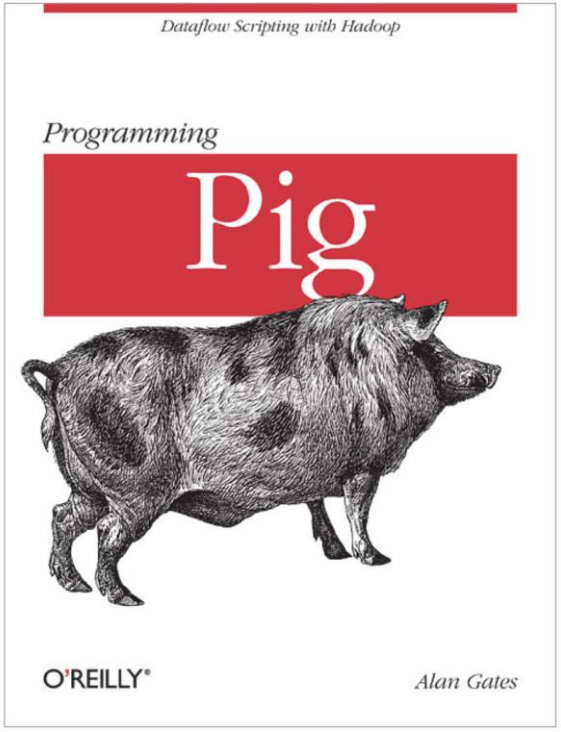
```
[^0-9A-Z\s]
```

```
RT @USER_5D4D777A: YOURE A FAG FOR GETTING IN THE MIDDLE OF THIS @USER_AB059BDC WHO THE FUCK ARE YOU ? A FUCKING NOBODY !!!!
```

Apache Pig

Learning Pig





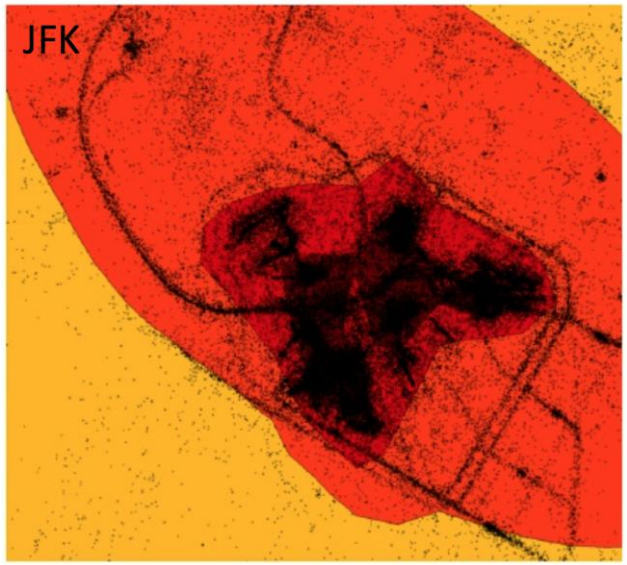
Your best friend? Apache Pig Online Manual → http://pig.apache.org/docs/r0.14.0/

Demo Data – Geotagged Tweets

ID	DateTime	Latitude	Longitude	Tweet
USER_8d0e8566	2010-03-02T23:00:44	30.387524	-91.109663	Pre-workout prep has begun.
USER_8d0e8566	2010-04-02T23:04:20	30.387524	-91.109663	I really don't like that a college FB player's ON-FIELD production can be negated by a single workout. So proof's NOT in the pudding?
USER_87b48222	2010-03-02T23:23:29	37.530819	-77.475577	@USER_9bb099c2 15 pages??? fuck u mean!!?? damn.
USER_87b48222	2010-07-02T23:43:57	37.530819	-77.475577	@USER_e97d1292 lol do u know that song?
USER_01b8a291	2010-03-03T00:56:16	41.51179	-95.893286	HAHAHA OMG! I just found a baggie of weed that I hid from like four/five years ago!! hahaha!!!
USER_2e5f877	T02:06:15	39.669307	-79.85002	@USER_2b2bd61b light skin fr y and ou n pies
USER_942c68d	2010- T02:21:36	41.220425	-85.861873	These judge being this yea
USER_8d0e856	03T02:28:1	24	-	@ 7cdabe3 Peo ke that to get the b ym frowned n..LOL.
USER_8d0e856	2010-0 2:2	30.39	- 502	USER_ 5e2 ally ge d of my iPh s dysfu l button this ke t a 970 ckertheBerrytheSweetertheUse
USER_2e5f877	2010-0 2:4	39.6	002	@ 7ac8d y Cuz...Where been at?
USER_8d0e856	T02:43:0	85	0458	RT 5e2: @ I think that ove! | Mal pen and we ca Wo le against ea her.
USER_8d0e8566	2010-03-03T02:53:19	30.393485	-91.110458	@USER_b7cdabe3 Oh, okay!
USER_942c68df	2010-07-03T02:55:36	41.234181	-85.812994	@USER_20c15b69 Me too.
USER_8d0e8566	2010-06-03T03:00:37	30.387524	-91.109663	The next 2hrs of tweets are @USER_fe579e73 for gibing me the idea with his #theory tweet
USER_942c68df	2010-03-03T03:14:53	41.234181	-85.812994	@USER_21fe08ea Aww that sucks. If ya dont mind me asking, whats ruining your relationship?
USER_8d0e8566	2010-05-03T03:26:46	30.387524	-91.109663	@USER_fe579e73 did u change ur settings to use twitlonger?
USER_8d0e8566	2010-03-03T03:29:41	30.387524	-91.109663	RT @USER_de057bc2: Twitter is jacked up tonight | Just on iPhones. #BlackertheBerrytheSweetertheUse
USER_8d0e8566	2010-03-03T03:33:47	30.387524	-91.109663	RT @USER_de057bc2: @USER_8d0e8566 EFF YO Blackberry | Sore Loser
USER_8d0e8566	2010-06-03T03:47:43	30.387524	-91.109663	@USER_45b5c066 @USER_2b5b12ff The body nice but that had to be a contest at a Bukket Nekked.
USER_8d0e8566	2010-06-03T03:57:23	30.387524	-91.109663	#PeterWisdom "If u wake up and ur gal or the gal ur in bed with is staring at u,take solace in knowing she'll be sleep when u escape." LOL
USER_87b48222	2010-03-03T03:59:01	37.530819	-77.475577	Where do you those rip away jeans?!! @USER_af454d84 and where can I get some?!
USER_8d0e8566	2010-03-03T04:17:29	30.387524	-91.109663	@USER_b7cdabe3 LOL
USER_8d0e8566	2010-03-03T04:37:07	30.387524	-91.109663	RT @USER_45b5c066: #FamilyGuy Meg and Brian make out. Meg stalks him like Misery << did u just use a shag blog term?? #CLASSIC | Did I?

Data for Practice

- UMN/Sarwat Foursquare Dataset
  - https://archive.org/details/201309_foursquare_dataset_umn
- Data
  - User.dat
  - Venues.dat
  - Checkins.dat
  - Socialgraph.dat
  - Ratings.dat



Data infrastructure – present

Introduced a lot of new systems
- Cloudera's Hadoop Distro - CDH3u3
- Oozie for workflow / data management
- Pig for reporting
- Scaled back ruby / hive dashboard
- BSON mongo dumps
- Scala MapReduce
- Scoobi

cloudera

OOZIE

BSON {01010100 11101011 10101110 01010101}

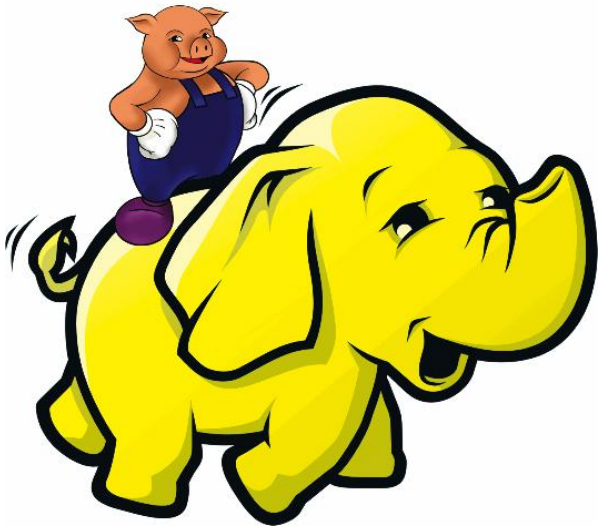
36

2012 FOURSQUARE LABS

foursquare

Apache Pig

- Pig Latin, a high level data processing language.
- An engine that executes Pig Latin locally or on a Hadoop cluster



Pig Latin Example – Word Count

This is a relation, NOT a variable

loads a file into a relation

file schema

```
a = LOAD '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
```

projection operation

explode function

```
b = FOREACH a GENERATE FLATTEN(TOKENIZE(tweet)) AS token;
```

group

tokenization function

```
c = GROUP b BY token;
```

count function

```
d = FOREACH c GENERATE group AS token, COUNT_STAR(b) AS cnt;
```

order operation

```
e = ORDER d BY cnt DESC;
```

limit operation

```
f = LIMIT e 20;
```

```
DUMP f;
```

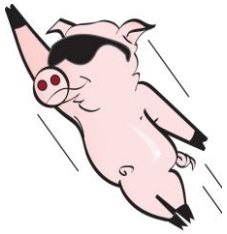
pig doesn't execute until it sees keyword such as dump or store

Why Pig?

- Faster development
  - Fewer lines of code
  - Don't re-invent the wheel
  - No M/R programming
  - Joins in M/R is painful
  - Chaining together M/R jobs is tedious
- Flexible
  - Metadata is optional
  - Extensible (UDFs → Piggybank, DataFu, etc.)
  - Procedural programming

Pig Philosophy

- Pigs eats anything
  - Pig can operate on data whether it has metadata or not. It can operate on data that is relational, nested, or unstructured. And it can easily be extended to operate on data beyond files, including key/value stores, databases, etc.
- Pigs live anywhere
  - Pig is intended to be a language for parallel data processing. It is not tied to one particular parallel framework (e.g., Pig on Spark, Pig on Storm)
- Pigs are domestic animals
  - Designed to be easily controlled and modified by users via User-Defined-Functions (UDF) and Stream command, etc.
- Pigs fly



Apache Pig Philosophy: http://pig.apache.org/philosophy.html

Pig Grunt/Shell

Linux	Pig Grunt Shell	FsShell (working with HDFS from pig grunt)

$ pig -x local
$ pig
$ pig -e script.pig
$ pig –param YEAR=2015 script.pig
	grunt> sh pwd
	grunt> sh ls -alF /home/lab/grunt> sh cat test.pig
		grunt> fs -ls /user
		grunt> fs -ls /user/lab/pig
		grunt> fs -put /home/lab/GeoText.2010-10-12/full_text.txt /user/lab/pig/full_text_1.txt
		grunt> copyFromLocal
		grunt> copyToLocal
		grunt> rmf filename
		grunt> kill jobid
		grunt> exec test.pig
		grunt> run test.pig
		grunt> describe fieldA;

DEMO

Data Preparation


```
[root@sandbox ~]# hadoop fs -mkdir /user/lab/pig
[root@sandbox ~]# pwd
/root
[root@sandbox ~]# cd /home/lab
[root@sandbox lab]# ll
total 125724
drwxr-xr-x 5 nagios games     4096 Jan 24 05:23 GeoText.2010-10-12
-rw-r--r-- 1 root   root  60973289 Jan 16 21:47 GeoText.2010-10-12.tgz
-rw-r--r-- 1 root   root   4994090 Jan 30 05:02 cities15000.txt
-rw-r--r-- 1 root   root       115 Jan 30 05:03 dayofweek.txt
-rw-r--r-- 1 root   root  57139942 Jan 16 15:34 full_text.txt
-rwxrwxr-- 1 root   root      1027 Jan 23 21:38 sc_reducer.py
-rw-r--r-- 1 root   root   5589917 Jan 30 05:03 shakespeare.txt
-rw-r--r-- 1 root   root     13880 Jan 30 05:03 timeZones.txt
-rwxrwxr-- 1 root   root       537 Jan 23 21:38 wc_mapper.py
[root@sandbox lab]# cd GeoText.2010-10-12
[root@sandbox GeoText.2010-10-12]# ll
total 111624
-rw-r--r-- 1 nagios games     2695 Oct 12  2010 README.txt
-rw-r--r-- 1 nagios games 57139942 Oct 12  2010 full_text.txt
-rw-r--r-- 1 root   root  57139942 Jan 24 05:23 full_text_2.txt
drwxr-xr-x 2 nagios games     4096 Jan 16 21:52 geo_eval
drwxr-xr-x 2 nagios games     4096 Jan 16 21:52 preproc
drwxr-xr-x 2 nagios games     4096 Jan 16 21:52 processed_data
[root@sandbox GeoText.2010-10-12]# hadoop fs -put full_text.txt /user/lab/pig/
[root@sandbox GeoText.2010-10-12]# hadoop fs -cat /user/lab/pig/full_text.txt | head
USER_79321756   2010-03-03T04:15:26     ÜT: 47.528139,-122.197916       47.528139       -122.197916     RT @USER_2ff4faca: IF SHE DO IT 1 MORE TIME.....
&gt;haha. #cutthatout
USER_79321756   2010-03-03T04:55:32     ÜT: 47.528139,-122.197916       47.528139       -122.197916     @USER_77a4822d @USER_2ff4faca okay:) lol. Saying
USER_79321756   2010-03-03T05:13:34     ÜT: 47.528139,-122.197916       47.528139       -122.197916     RT @USER_5d4d777a: YOURE A FAG FOR GETTING IN THE
OU ? A FUCKING NOBODY !!!!&gt;&gt;Lol! Dayum! Aye!
USER_79321756   2010-03-03T05:28:02     ÜT: 47.528139,-122.197916       47.528139       -122.197916     @USER_77a4822d yea ok..well answer that cheap as
USER_79321756   2010-03-03T05:56:13     ÜT: 47.528139,-122.197916       47.528139       -122.197916     A sprite can disappear in her mouth - lil kim hmm
USER_79321756   2010-03-03T16:52:44     ÜT: 47.528139,-122.197916       47.528139       -122.197916     Lmao! I still get txt when AJ tweets before they
s me dyin! @USER_a5b463b2 what's ur issue!
USER_79321756   2010-03-03T16:57:24     ÜT: 47.528139,-122.197916       47.528139       -122.197916     Alright twitters tryna take me over!
USER_79321756   2010-03-03T20:20:40     ÜT: 47.528139,-122.197916       47.528139       -122.197916     Just got to work. Got my pizza bagel and my raspb
 not til 2. I just wanna get it done!:D
USER_79321756   2010-03-03T23:23:33     ÜT: 47.528139,-122.197916       47.528139       -122.197916     Just got a txt from my cousin! Yes! So happy for
USER_79321756   2010-03-03T23:37:36     ÜT: 47.528139,-122.197916       47.528139       -122.197916     Why is this woman in the bathroom everytime I'm i
```

create a /usr/lab/pig HDFS directory for all pig labs

enter /home/lab on linux where you store the full_text.txt file

NOTE: this is a directory

enter the directory

Move the twitter data to HDFS under the pig folder

DEMO

Pig Grunt Shell Demo


```
[root@sandbox ~]# cd /home/lab
[root@sandbox lab]# ll
total 181584
drwxr-xr-x 5 nagios games     4096 Mar  6 03:32 GeoText.2010-10-12
-rw-r--r-- 1 root   root  60973289 Mar  8 19:29 GeoText.2010-10-12.tgz
-rw-r--r-- 1 root   root   4994090 Jan 30 05:02 cities15000.txt
-rw-r--r-- 1 root   root       115 Jan 30 05:03 dayofweek.txt
-rw-r--r-- 1 root   root       486 Mar  8 19:34 full_text-2.txt?dl=0
-rw-r--r-- 1 root   root  57139942 Mar  8 19:35 full_text-2.txt?dl=0.1
-rw-r--r-- 1 root   root  57139942 Jan 16 15:34 full_text.txt
-rw-r--r-- 1 root   root     15866 Mar  8 03:18 pig_1425784400308.log
-rw-r--r-- 1 root   root      2543 Mar  8 03:22 pig_1425784903845.log
-rw-r--r-- 1 root   root      5278 Mar  8 03:43 pig_1425785107653.log
-rw-r--r-- 1 root   root      1280 Mar  9 04:08 pig_1425874092273.log
-rwxrwxr-- 1 root   root      1027 Jan 23 21:38 sc_reducer.py
-rw-r--r-- 1 root   root        28 Mar  8 03:24 session.txt
-rw-r--r-- 1 root   root   5589917 Jan 30 05:03 shakespeare.txt
-rw-r--r-- 1 root   root       184 Mar  9 04:10 test1.pig
-rw-r--r-- 1 root   root       223 Mar  9 04:10 test2.pig
-rw-r--r-- 1 root   root     13880 Jan 30 05:03 timeZones.txt
-rw-r--r-- 1 root   root       144 Mar  8 03:44 url.txt
-rw-r--r-- 1 root   root        42 Mar  8 03:20 user.txt
-rwxrwxr-- 1 root   root       537 Jan 23 21:38 wc_mapper.py
[root@sandbox lab]# pig
15/03/09 18:24:12 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
15/03/09 18:24:12 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
15/03/09 18:24:12 INFO pig.ExecTypeProvider: Picked MAPREDUCE as the ExecType
2015-03-09 18:24:12,160 [main] INFO  org.apache.pig.Main - Apache Pig version 0.14.0.2.2.0.0-2041 (rexported) compil
2015-03-09 18:24:12,160 [main] INFO  org.apache.pig.Main - Logging error messages to: /home/lab/pig_1425925452159.lo
2015-03-09 18:24:12,200 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /root/.pigbootup not found
2015-03-09 18:24:13,444 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to
fs://sandbox.hortonworks.com:8020
grunt> sh pwd
/home/lab
grunt> sh ls
GeoText.2010-10-12
GeoText.2010-10-12.tgz
cities15000.txt
dayofweek.txt
full_text-2.txt?dl=0
full_text-2.txt?dl=0.1
full_text.txt
```

DEMO

Pig Grunt Shell Demo


```
grunt> fs -ls /user
Found 12 items
drwxr-xr-x   - hue       hdfs          0 2015-01-15 06:14 /user/4sq
drwxrwx---   - ambari-qa hdfs          0 2014-12-16 19:04 /user/ambari-qa
drwxr-xr-x   - guest     guest         0 2014-12-16 19:28 /user/guest
drwxr-xr-x   - hcat      hdfs          0 2014-12-16 19:13 /user/hcat
drwxr-xr-x   - hdfs      hdfs          0 2015-03-06 03:51 /user/hdfs
drwx------   - hive      hdfs          0 2014-12-16 19:08 /user/hive
drwxr-xr-x   - hue       hue           0 2014-12-16 19:27 /user/hue
drwxr-xr-x   - root      hdfs          0 2015-03-04 22:05 /user/lab
drwxrwxr-x   - oozie     hdfs          0 2014-12-16 19:10 /user/oozie
drwx------   - root      hdfs          0 2015-02-23 06:58 /user/root
drwxr-xr-x   - solr      hdfs          0 2014-12-16 19:24 /user/solr
drwxr-xr-x   - hue       hdfs          0 2015-02-23 07:07 /user/twitter
grunt> fs -ls /user/lab/pig
Found 9 items
-rw-r--r--   1 root hdfs         30 2015-03-07 20:23 /user/lab/pig/data_test_map
-rw-r--r--   1 root hdfs        115 2015-03-06 03:18 /user/lab/pig/dayofweek.txt
-rw-r--r--   1 root hdfs   57139942 2015-03-05 19:59 /user/lab/pig/full_text.txt
drwxr-xr-x   - root hdfs          0 2015-03-09 04:13 /user/lab/pig/full_text_limit3
drwxr-xr-x   - root hdfs          0 2015-03-06 17:55 /user/lab/pig/full_text_t_map
drwxr-xr-x   - root hdfs          0 2015-03-06 15:37 /user/lab/pig/full_text_t_sample
-rw-r--r--   1 root hdfs         28 2015-03-08 03:25 /user/lab/pig/session.txt
-rw-r--r--   1 root hdfs        144 2015-03-08 03:44 /user/lab/pig/url.txt
-rw-r--r--   1 root hdfs         42 2015-03-08 03:21 /user/lab/pig/user.txt
grunt> fs -put /home/lab/GeoText.2010-10-12/full_text.txt /user/lab/pig/full_text_1.txt
grunt> fs -ls /user/lab/pig/
Found 10 items
-rw-r--r--   1 root hdfs         30 2015-03-07 20:23 /user/lab/pig/data_test_map
-rw-r--r--   1 root hdfs        115 2015-03-06 03:18 /user/lab/pig/dayofweek.txt
-rw-r--r--   1 root hdfs   57139942 2015-03-05 19:59 /user/lab/pig/full_text.txt
-rw-r--r--   1 root hdfs   57139942 2015-03-09 18:25 /user/lab/pig/full_text_1.txt
drwxr-xr-x   - root hdfs          0 2015-03-09 04:13 /user/lab/pig/full_text_limit3
drwxr-xr-x   - root hdfs          0 2015-03-06 17:55 /user/lab/pig/full_text_t_map
drwxr-xr-x   - root hdfs          0 2015-03-06 15:37 /user/lab/pig/full_text_t_sample
-rw-r--r--   1 root hdfs         28 2015-03-08 03:25 /user/lab/pig/session.txt
-rw-r--r--   1 root hdfs        144 2015-03-08 03:44 /user/lab/pig/url.txt
-rw-r--r--   1 root hdfs         42 2015-03-08 03:21 /user/lab/pig/user.txt
grunt> rmf /user/lab/pig/full_text_1.txt
2015-03-09 18:26:15,162 [main] INFO  org.apache.pig.tools.grunt.GruntParser - Waited 0ms to delete file
```

Pig Data Types

Type	Description	Example
int, long, float, duble, chararray, bytearray, boolean, datetime	• *primitive data types in pig*	
tuple	• *Fixed length, **ordered** set of fields, like a row with multiple columns in SQL* • *Allows random access* • *Must fit in memory*	*(toronto, 3)* *('bob', 53, 'toronto', 'male')*
bag	• *An **unordered** collection of tuples* • *Can have tuples with differing numbers of fields* • *Can spill to disk (doesn't have to fit in memory*	*{(toronto, 3),(chicago, 5)}* *{('bob',53,'toronto','male'), ('sally', 23), 'george', 'montreal')}*
map	• *A set of key/value pairs* • *Key/values in a relation must be unique* • *Key must be chararray, data element can be any type*	*[city#toronto]* *[name#bob,age#53,city#toronto ,gender#male]*

DEMO

Pig Map[] Demo


```
-- map example 2
-- data prep
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float, tweet:chararray);
b = foreach a generate id, ts, TOTUPLE(lat, lon) as loc_tuple:tuple(lat:chararray, lon:chararray), flatten(TOKENIZE(tweet)) as
token;
c = group b by (id, token);
d = foreach c generate flatten(group) as (id, token), COUNT(b) as cnt;
e = group d by id;
f = foreach e generate group as id, flatten(TOP(10, 2, d)) as (id1, word,cnt);
g = foreach f generate id, TOMAP(word, cnt) as freq_word:map[];
h = group g by id;
store h into '/user/lab/pig/full_text_t_map';

a = load '/user/lab/pig/full_text_t_map';
b = limit a 3;
dump b;

-- load map type and get top 5 frequent words of a tweeter
a = load '/user/lab/pig/full_text_t_map' as (id:chararray, freq_word:bag{t:(id1:chararray, freq_word_m:map[])});
b = foreach a generate id, flatten(freq_word) as (id1, freq_word_m);
c = filter b by (int)freq_word_m#'l' > 5;
d = limit c 10;
dump d;
```

DEMO

Pig Bag{} Demo


```
grunt> a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float, tweet:chararray);
grunt>
grunt> describe a;
a: {id: chararray,ts: chararray,location: chararray,lat: float,lon: float,tweet: chararray}
grunt>
grunt> b = foreach a generate FLATTEN(TOKENIZE(tweet)) as token;
grunt> describe b;
b: {token: chararray}
grunt> illustrate b;
-------------------------------------------------------------------------------------------------------------------------------------------------------
|       | USER_8527124d     | 2010-03-06T15:54:35 | ?T: 25.721908,-80.333783 | 25.721909      | -80.333786     | Breakeven&lt;3.      |
-------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------
| b     | token:chararray      |
--------------------------------
|       | Breakeven&lt;3.      |
--------------------------------

grunt> c = group b by token;
grunt> describe b;
b: {token: chararray}
grunt> illustrate c;
-------------------------------------------------------------------------------------------------------------------------------------------------------
|       | USER_87b48222     | 2010-03-04T07:10:56 | ?T: 37.530819,-77.475577 | 37.53082       | -77.47558      | Bye;                 |
|       | USER_87b48222     | 2010-03-03T05:05:29 | ?T: 37.530819,-77.475577 | 37.53082       | -77.47558      | Bye;                 |
-------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------
| b     | token:chararray      |
--------------------------------
|       | Bye;                 |
|       | Bye;                 |
--------------------------------
-------------------------------------------------------------------------------
| c     | group:chararray      | b:bag{:tuple(token:chararray)}               |
-------------------------------------------------------------------------------
|       | Bye;                 | {(Bye;), (Bye;)}                             |
-------------------------------------------------------------------------------
```

Pig Functions – Date/Time


```
---------------------------
-- DATE/Time functions
-- CurrentTime()
-- GetYear()
-- GetMonth()
-- GetDay()
-- GetWeek()
-- ToDate()
-- ToString()
-- ToUnixTime()
-------------------------


-- Date/Time functions
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float, tweet:chararray);
b = foreach a generate id, ts, ToDate(ts) as ts1;
c = foreach b generate id, ts, ts1, ToString(ts1) as ts_iso, ToUnixTime(ts1), GetYear(ts1) as year, GetMonth(ts1) as month,
GetWeek(ts1) as week;
d = limit c 5;
dump d;
```

DEMO

Date/Time Function Demo


```
grunt> a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float, tweet:chararray);
grunt> b = foreach a generate id, ts, ToDate(ts) as ts1;
grunt> c = foreach b generate id, ts, ts1, ToString(ts1) as ts_iso, ToUnixTime(ts1), GetYear(ts1) as year, GetMonth(ts1) as month, GetWeek(ts1) as week;
grunt> d = limit c 5;
grunt> dump d;
2015-03-09 19:18:31,207 [main] INFO  org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script: LIMIT
2015-03-09 19:18:31,377 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.schematuple] was not set... will not generate code.
2015-03-09 19:18:31,501 [main] INFO  org.apache.pig.newplan.logical.optimizer.LogicalPlanOptimizer - {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, Const
ptimizer, LoadTypeCastInserter, MergeFilter, MergeForEach, PartitionFilterOptimizer, PredicatePushdownOptimizer, PushDownForEachFlatten, PushUpFilter, Sp
2015-03-09 19:18:31,566 [main] INFO  org.apache.pig.newplan.logical.rules.ColumnPruneVisitor - Columns pruned for a: $2, $3, $4, $5
2015-03-09 19:18:32,599 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.schematuple] was not set... will not generate code.
2015-03-09 19:18:32,770 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2015-03-09 19:18:32,780 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
2015-03-09 19:18:33,456 [main] INFO  org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter - Saved output of task 'attempt__0001_m_000001_1' to hdfs
2/tmp-1176900911/_temporary/0/task__0001_m_000001
2015-03-09 19:18:33,518 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2015-03-09 19:18:33,534 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2015-03-09 19:18:33,534 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(USER_79321756,2010-03-03T04:15:26,2010-03-03T04:15:26.000Z,2010-03-03T04:15:26.000Z,1267589726,2010,3,9)
(USER_79321756,2010-03-03T04:55:32,2010-03-03T04:55:32.000Z,2010-03-03T04:55:32.000Z,1267592132,2010,3,9)
(USER_79321756,2010-03-03T05:13:34,2010-03-03T05:13:34.000Z,2010-03-03T05:13:34.000Z,1267593214,2010,3,9)
(USER_79321756,2010-03-03T05:28:02,2010-03-03T05:28:02.000Z,2010-03-03T05:28:02.000Z,1267594082,2010,3,9)
(USER_79321756,2010-03-03T05:56:13,2010-03-03T05:56:13.000Z,2010-03-03T05:56:13.000Z,1267595773,2010,3,9)
```

Pig Functions - String


```
-- Use SUBSTRING() to extract year from ts string
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float,
tweet:chararray);
b = foreach a generate id, ts, SUBSTRING(ts, 0,4);
c = limit b 5;
dump c;

-- Find first twitter handles mentioned in a tweet using regex_extract() function
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float,
tweet:chararray);
b = foreach a generate id, ts, location, LOWER(tweet) as tweet;
c = foreach b generate id, ts, location, REGEX_EXTRACT(tweet, '(.*)@user_(\\S{8})([: | ])(.*)',2) as tweet;
d = limit c 5;
dump d;

-- Finding users who tweet long tweets
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float,
tweet:chararray);
b = foreach a generate id, ts, location, SIZE(REPLACE(tweet, '@USER_\\w{8}', '') ) as tweet_len;
c = order b by tweet_len desc;
d = limit c 10;
dump d;
```

DEMO

String Function Demo


```
grunt> a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float, lon:float, tweet:chararray);
grunt> b = foreach a generate id, ts, location, SIZE(REPLACE(tweet, '@USER_\\w{8}', '') ) as tweet_len;
grunt> c = order b by tweet_len desc;
grunt> d = limit c 10;
grunt> dump d;


(USER_2f88c77e,2010-03-06T16:57:30,?T: 36.888035,-76.227592,158)
(USER_3ca302f1,2010-03-06T01:49:43,?T: 30.019751,-95.444788,151)
(USER_3ca302f1,2010-03-04T01:15:19,?T: 30.019751,-95.444788,150)
(USER_2c8d1305,2010-03-05T18:48:13,?T: 33.994123,-118.280561,149)
(USER_c8613ca2,2010-03-04T17:20:27,?T: 40.623387,-73.917237,147)
(USER_4cecd527,2010-03-05T03:22:45,?T: 47.573729,-122.312608,146)
(USER_11ac7eaf,2010-03-03T20:02:46,?T: 30.452975,-91.108623,146)
(USER_77ee1910,2010-03-04T03:01:28,?T: 39.950974,-75.166685,146)
(USER_ae406f1d,2010-03-03T17:25:55,Pre: 40.729685,-74.006611,146)
(USER_2dcd8488,2010-03-06T21:37:02,iPhone: 30.426498,-91.137184,146)
```

Pig Relational Operations

- load/store
- foreach
- filter
- group
- flatten
- limit
- order by
- join
- sample

Relations

- Pig's fundamental building blocks
- Similar to a **table**
- Don't confuse it with variables in other languages

```
a = LOAD '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
```

ID	DateTime	Latitude	Longitude	Tweet
USER_8d0e8566	2010-03-02T23:00:44	30.387524	-91.109663	Pre-workout prep has begun.
USER_8d0e8566	2010-04-02T23:04:20	30.387524	-91.109663	I really don't like that a college FB player's ON-FIELD production can be negated by a single workout. So proof's NOT in the pudding?
USER_87b48222	2010-03-02T23:23:29	37.530819	-77.475577	@USER_9bb099c2 15 pages??? fuck u mean!!?? damn.
USER_87b48222	2010-07-02T23:43:57	37.530819	-77.475577	@USER_e97d1292 lol do u know that song?

DEMO

Load Data


```
a = LOAD '/user/lab/pig/full_text.txt' USING
PigStorage('\t') AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
b = foreach a generate id, ts, lat, lon;
c = limit b 5;
DUMP c;
```

```
(USER_79321756,2010-03-03T04:15:26,47.528137,-122.197914)
(USER_79321756,2010-03-03T04:55:32,47.528137,-122.197914)
(USER_79321756,2010-03-03T05:13:34,47.528137,-122.197914)
(USER_79321756,2010-03-03T05:28:02,47.528137,-122.197914)
(USER_79321756,2010-03-03T05:56:13,47.528137,-122.197914)
```

DEMO

Store Data


```
a = LOAD '/user/lab/pig/full_text.txt' USING PigStorage('\t') AS
(id:chararray, ts:chararray, location:chararray, lat:float, lon:float,
tweet:chararray);

b = foreach a generate id, ts, lat, lon;

c = limit b 5;

STORE c INTO '/user/lab/pig/full_text_proj' USING PigStorage('#') ;
```

```
USER_79321756#2010-03-03T04:15:26#47.528137#-122.197914
USER_79321756#2010-03-03T04:55:32#47.528137#-122.197914
USER_79321756#2010-03-03T05:13:34#47.528137#-122.197914
USER_79321756#2010-03-03T05:28:02#47.528137#-122.197914
USER_79321756#2010-03-03T05:56:13#47.528137#-122.197914
```

DEMO

Foreach (Projection)

- Apply transformations on every row in a relation
- Projections can be chained
- Creates a new relation
- Analogous to “select” statement in Hive

```
a = LOAD '/user/lab/pig/full_text.txt' USING PigStorage('\t') AS
(id:chararray, ts:chararray, location:chararray, lat:float, lon:float,
tweet:chararray);

b = FOREACH a GENERATE id, To_Date(ts), (lat, lon) as location;

c = limit b 5;

STORE c INTO '/user/lab/pig/full_text_proj' USING PigStorage('#') ;
```

Filtering Data

- An evaluation is done for each row in a relation
  - The row is tossed if it does not meet certain criteria
- Can be used with boolean logic, regular expression etc.

```
b = FILTER a by id=='1234567'
```

For more on pig comparison operators, check out
http://pig.apache.org/docs/r0.14.0/basic.html#comparison

Filtering Data

DEMO

find retweets in NYC on 12th with length smaller than 50 characters


```
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray,
lat:float, lon:float, tweet:chararray);
b = FILTER a by lat > 40.4774 and lat < 40.9176 and lon > -74.2589 and lon < -
73.7004 and
        SIZE(tweet)<50 and
        GetHour(ToDate(ts))==12 and
        tweet matches 'RT.*';
c = foreach b generate id, ts, lat, lon, tweet;
d = limit c 10;
dump d;
```

```
(USER_a8cecd53,2010-03-04T12:44:36,40.812267,-73.95978,RT @USER_a55ef2d4: on da dam bus goin to skool)
(USER_f8eb395d,2010-03-03T12:37:19,40.677395,-73.87458,RT @USER_ca1182eb: Hate has no color or age!)
(USER_827611e3,2010-03-04T12:40:03,40.75,-73.997,RT @USER_a2a7b7e5: Watching Spongebob !)
(USER_18c466a9,2010-03-06T12:54:41,40.899567,-73.85717,RT @USER_86e28cc0 hella bored - Tweet people)
(USER_838d6d62,2010-03-05T12:37:48,40.672306,-73.92697,RT @USER_cb2f3989: @USER_838d6d62 smh/im up)
(USER_838d6d62,2010-03-05T12:52:00,40.672306,-73.92697,RT @USER_231d82a1: @USER_838d6d62 gud mornin/:))
(USER_0b58828b,2010-03-04T12:19:51,40.83705,-73.86019,RT @USER_32550f0c: Good Morning | goodmorning)
(USER_c6710d1e,2010-03-03T12:45:57,40.664516,-73.945206,RT @USER_e517c738: yeah im about to let him go!)
```

Group (aggregation)

- The group statement collects together records with the same key into a bag
- Grouping does not change the data
  - Reorganizes it based on the given key
  - Can group on multiple keys
- First column is always called **group**
  - A compound group key will be a Tuple ("group") whose elements are the keys
- Second column is a Bag
  - Name is the grouped relation
  - Contains every row associated with key

DEMO

Group (aggregation)


```
-- Calculate # of tweets per user

a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
b = GROUP a BY id;
c = FOREACH b GENERATE group AS id, COUNT(a) as cnt;
d = order c by cnt desc;
e = limit d 5;
dump e;

-- visualize group
illustrate b;
```

```
--------------------------------------------------------------------------------------------------------------------------------------------------------
| a      | id:chararray      | ts:chararray         | location:chararray         | lat:float      | lon:float      | tweet:chararray                                          |
--------------------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_c296a14f     | 2010-03-03T13:41:06 | ?T: 42.133812,-87.887853 | 42.133812      | -87.887856     | @USER_b8034b39 eh? You dnt NEVER be on. How u doin love?    |
|        | USER_c296a14f     | 2010-03-03T06:05:51 | ?T: 42.133812,-87.887853 | 42.133812      | -87.887856     | @USER_7b31238d lmao nawww not a hater -- a regulator. [smh] |
--------------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| b      | group:chararray       | a:bag{:tuple(id:chararray,ts:chararray,location:chararray,lat:float,lon:float,tweet:chararray)}                                                   |
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_c296a14f         | {(USER_c296a14f, ..., @USER_b8034b39 eh? You dnt NEVER be on. How u doin love?), (USER_c296a14f, ..., @USER_7b31238d lmao nawww not a hater -- a regulator. [smh])} |
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
```

DEMO

Group (aggregation)


```
-- Count total number of records

a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
b = GROUP a ALL;
c = FOREACH b GENERATE COUNT_STAR(a);
dump c;

-- visualize relation b
illustrate b;
```

```
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| a     | id:chararray       | ts:chararray        | location:chararray        | lat:float       | lon:float       | tweet:chararray                                    |
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|       | USER_06f07cc0      | 2010-03-05T23:29:11 | ?T: 40.805065,-73.563726  | 40.805065       | -73.56373       | My reflection has a hard time being me.            |
|       | USER_6ccd47dd      | 2010-03-03T19:37:11 | ?T: 44.968375,-124.012122 | 44.968376       | -124.01212      | @USER_3903ee32 dopppeee!!!                         |
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| b     | group:chararray       | a:bag{:tuple(id:chararray,ts:chararray,location:chararray,lat:float,lon:float,tweet:chararray)}                                                                             |
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|       | all                   | {(USER_06f07cc0, ..., My reflection has a hard time being me.), (USER_6ccd47dd, ..., @USER_3903ee32 dopppeee!!!)}                                                          |
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
```

Flatten (explode/transpose)

- **flatten** is the inverse of **group**
- flatten is pig's explode function, similar to Excel's transpose function, but is more flexible
- flatten turns tuples into columns
- flatten turns bags into rows
- flatten reference (naming)
  - After flatten, if there're ambiguous field names, you need to use the disambiguate operator **::**

Flatten Tuples

DEMO

-- Calculate # of tweets per user per day

```
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray,
lat:float, lon:float, tweet:chararray);
b = foreach a generate id, SUBSTRING(ts, 0, 10) as date, lat, lon, tweet;
c = GROUP b BY (id, date);
d = FOREACH c GENERATE FLATTEN(group) AS (id,date) , COUNT(b) as cnt;
e = order d by cnt desc;
f = limit e 5;
dump f;
```

```
(USER_18c466a9,2010-03-07,89)
(USER_943f9c88,2010-03-04,89)
(USER_b2bb70e4,2010-03-03,85)
(USER_943f9c88,2010-03-03,83)
(USER_9e14b9d7,2010-03-05,81)
```

```
-- visualize group
illustrate c;
```

a	id:chararray	ts:chararray	location:chararray	lat:float	lon:float	tweet:chararray
	USER_583b948c	2010-03-04T23:27:10	?T: 33.551659,-84.563961	33.55166	-84.56396	@USER_11f14a3d what movie u looking at
	USER_583b948c	2010-03-04T23:44:35	?T: 33.551659,-84.563961	33.55166	-84.56396	@USER_11f14a3d U didnt look at it cause u were on twitte

b	id:chararray	date:chararray	lat:float	lon:float	tweet:chararray
	USER_583b948c	2010-03-04	33.55166	-84.56396	@USER_11f14a3d what movie u looking at
	USER_583b948c	2010-03-04	33.55166	-84.56396	@USER_11f14a3d U didnt look at it cause u were on twitter smh!!! Twitter nympho

c	group:tuple(id:chararray,date:chararray)	b:bag{:tuple(id:chararray,date:chararray,lat:float,lon:float,tweet:chararray)}
	(USER_583b948c, 2010-03-04)	{(USER_583b948c, ..., @USER_11f14a3d what movie u looking at), (USER_583b948c, ..., @USER_11f14a3d U ... tter smh!!! Twitter nympho)}

Flatten Tuples – cont'd

DEMO


```
-- Calculate # of tweets per user per day

a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float,
lon:float, tweet:chararray);
b = foreach a generate id, SUBSTRING(ts, 0, 10) as date, lat, lon, tweet;
c = GROUP b BY (id, date);
d = FOREACH c GENERATE FLATTEN(group) AS (id,date) , COUNT(b) as cnt;
e = order d by cnt desc;
f = limit e 5;
dump f;

-- visualize group
illustrate d;
```

```
--------------------------------------------------------------------------------------------------------------------------------------------
| a      | id:chararray      | ts:chararray          | location:chararray        | lat:float      | lon:float      | tweet:chararray
--------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_2dfabc44     | 2010-03-07T02:28:13 | ?T: 33.92368,-84.304425 | 33.92368       | -84.30443      | @USER_cc540e1c another non invite.... rotten..
|        | USER_2dfabc44     | 2010-03-07T15:51:14 | ?T: 33.92368,-84.304425 | 33.92368       | -84.30443      | @USER_a8349405 good lookin on the link
--------------------------------------------------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------------------------------------
| b      | id:chararray      | date:chararray       | lat:float      | lon:float      | tweet:chararray                                       |
------------------------------------------------------------------------------------------------------------------------------
|        | USER_2dfabc44     | 2010-03-07           | 33.92368       | -84.30443      | @USER_cc540e1c another non invite.... rotten.. just rotten |
|        | USER_2dfabc44     | 2010-03-07           | 33.92368       | -84.30443      | @USER_a8349405 good lookin on the link                |
------------------------------------------------------------------------------------------------------------------------------

--------------------------------------------------------------------------------------------------------------------------------------------
| c      | group:tuple(id:chararray,date:chararray)          | b:bag{:tuple(id:chararray,date:chararray,lat:float,lon:float,tweet:chararray)}
--------------------------------------------------------------------------------------------------------------------------------------------
|        | (USER_2dfabc44, 2010-03-07)                       | {}
|        | (USER_2dfabc44, 2010-03-07)                       | {}
--------------------------------------------------------------------------------------------------------------------------------------------

-------------------------------------------------------------------
| d      | id:chararray      | date:chararray       | cnt:long      |
-------------------------------------------------------------------
|        | USER_2dfabc44     | 2010-03-07           | 2             |
-------------------------------------------------------------------
```

Flatten Bags


```
-- Flatten Bags Example
a = load '/user/lab/pig/full_text.txt' AS (id:chararray, ts:chararray, location:chararray, lat:float,
lon:float, tweet:chararray);
b = foreach a generate id, SUBSTRING(ts, 0, 10) as date, lat, lon, tweet;
c = GROUP b BY (id, date);
d = FOREACH c GENERATE FLATTEN(b) AS (id, date, lat, lon, tweet);
e = order d by id, date;
f = limit e 50;
dump f;

-- visualize group
illustrate d;
```

```
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| a      | id:chararray       | ts:chararray          | location:chararray       | lat:float       | lon:float       | tweet:chararray
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_d54c0e1e      | 2010-03-06T14:41:08 | ?T: 37.187699,-77.344635 | 37.1877         | -77.344635      | Morning every1
|        | USER_d54c0e1e      | 2010-03-06T05:47:09 | ?T: 37.26009,-77.394015  | 37.26009        | -77.39401       | Aawwww bak n da college life w/ my sistas @ da Pretty Fr
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| b      | id:chararray       | date:chararray        | lat:float       | lon:float       | tweet:chararray                                                                                    |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_d54c0e1e      | 2010-03-06            | 37.1877         | -77.344635      | Morning every1                                                                                     |
|        | USER_d54c0e1e      | 2010-03-06            | 37.26009        | -77.39401       | Aawwww bak n da college life w/ my sistas @ da Pretty Fresh house party smh lol |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
---------
| c      | group:tuple(id:chararray,date:chararray)                 | b:bag{:tuple(id:chararray,date:chararray,lat:float,lon:float,tweet:chararray)}
         |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
---------
|        | (USER_d54c0e1e, 2010-03-06)                              | {(USER_d54c0e1e, ..., Morning every1), (USER_d54c0e1e, ..., Aawwww bak n da college life w/ my sistas
h lol)} |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
---------
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
| d      | id:chararray       | date:chararray        | lat:float       | lon:float       | tweet:chararray                                                                                    |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
|        | USER_d54c0e1e      | 2010-03-06            | 37.1877         | -77.344635      | Morning every1                                                                                     |
|        | USER_d54c0e1e      | 2010-03-06            | 37.26009        | -77.39401       | Aawwww bak n da college life w/ my sistas @ da Pretty Fresh house party smh lol |
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
```

Join

find tweets on weekends


```
-- prepare lookup table 'dayofweek'
fs -put /home/lab/dayofweek.txt /user/lab/pig/

-- Find Weekend Tweets (inner join)
a = load '/user/lab/pig/full_text.txt' using PigStorage('\t') AS (id:chararray, ts:chararray,
location:chararray, lat:float, lon:float, tweet:chararray);
a1 = foreach a generate id, ts, SUBSTRING(ts,0,10) as date;

b = load '/user/lab/pig/dayofweek.txt' using PigStorage('\t') as (date:chararray,
dow:chararray);
b1 = filter b by dow=='Saturday' or dow=='Sunday';

c = JOIN a1 BY date, b1 BY date;
d = foreach c generate a1::id .. a1::date, b1::dow as dow;
e = limit d 5;
dump e;
```

More on joins in next session

```
(USER_606adf97,2010-03-06T08:16:55,2010-03-06,Saturday)
(USER_8c704efa,2010-03-06T17:51:27,2010-03-06,Saturday)
(USER_8c704efa,2010-03-06T18:49:45,2010-03-06,Saturday)
(USER_8c704efa,2010-03-06T19:03:48,2010-03-06,Saturday)
(USER_8c704efa,2010-03-06T19:19:05,2010-03-06,Saturday)
```

Assignment #3

- *(Pig Latin) Find first 3 twitter handles mentioned in a tweet using regex_extract() function*
- *(Pig Latin) Find 3 most popular topics (hashtags)*