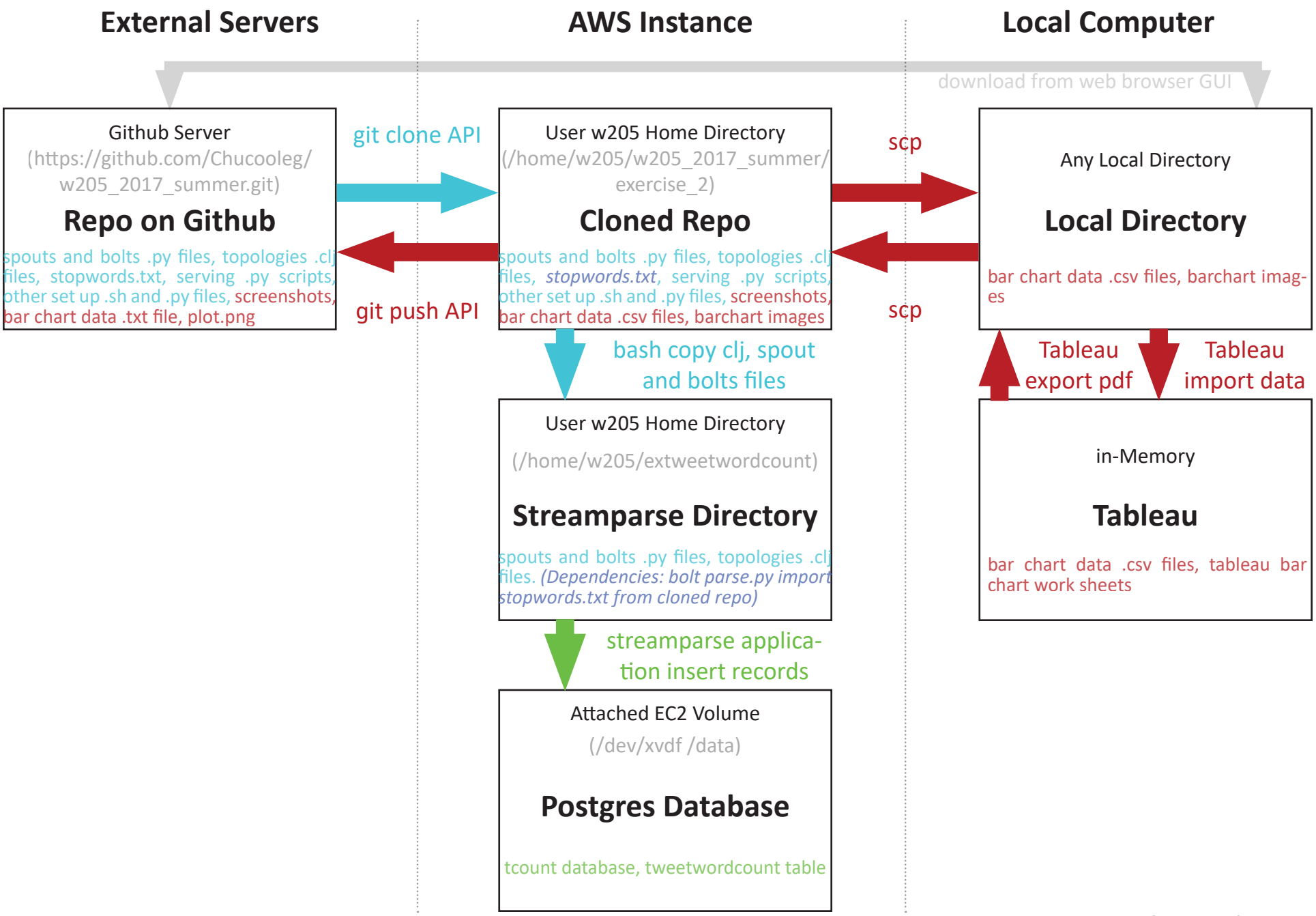
DIRECTORIES STRUCTURE AND FILES FLOW



LEGG (HO MAN) YEUNG

RUNNING THE APPLICATION (after all files are cloned and in place)

AWS Instance

```
[w205@ip-172-31-26-154 exercise_2]$ python finalresults.py >> finalresults
worthwhile 2
would've 6
woulda 2
wouldn't 13
wow 45
wrap 5
wrapping 1
wrestle 1
write 3
writer 3
writers 2
writing 9
written 2
wrong 35
wrong...all 1
```

```
w205@ip-172-31-26-154 exercise_2]$ python finalresults.py trump
otal number of occurrences of of "trump": 79
w205@ip-172-31-26-154 exercise_2]$ python finalresults.py hate
otal number of occurrences of of "hate": 53
w205@ip-172-31-26-154 exercise_2]$ python finalresults.py government
otal number of occurrences of of "government": 14
w205@ip-172-31-26-154 exercise_2]$ python finalresults.py russia
```

/home/w205/w205_2017_summer/exercise_2

2. Serving Scripts

Get quick summaries of word counts or ranges by running: (usage will show)

$python finalresults.py

$python histogram.py

```
2321 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt kindness: 1
2323 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt precious: 4
2325 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt friends: 30
2326 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt thank: 47
2328 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt sunflowers: 1
2330 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt blowing: 1
2332 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt wind: 6
2333 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt lot: 23
2335 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt sounds: 7
2337 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt mate: 2
2338 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt kids: 24
2340 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt futures: 1
2342 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt bright: 3
2343 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt lads: 1
2345 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt didn't: 33
2347 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt pls: 8
2348 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt support: 29
2350 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt girly: 2
2352 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt she's: 9
2353 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt sweetest: 3
2355 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt kid: 10
2357 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt tomorrow: 18
2358 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt who's: 7
2360 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt spent: 7
2362 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt life: 65
2363 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt avoiding: 1
2365 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt armed: 4
2367 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt combat: 3
2368 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt try: 18
2370 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt define: 2
2372 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt image: 1
2373 [Thread-27] INFO  backtype.storm.task.ShellBolt - ShellLog pid:13197, name:count-bolt yeah: 13
```

/home/w205/extweetwordcount

1. Storm Streaming

Use Streamparse to stream tweets with tweeter API and credentials by running: (python library dependencies: Tweepy and psycopg2

$sparse run

```
tcount=# select * from tweetwordcount order by count desc  limit 20;
   word   | count
----------+-------
 love     |   159
 people   |   132
 time     |    94
 shit     |    85
 fuck     |    64
 life     |    63
 day      |    62
 feel     |    59
 trump    |    55
 thank    |    44
 video    |    43
 military |    42
 care     |    42
 trans    |    41
 lol      |    41
 ur       |    41
 fucking  |    40
 happy    |    39
 look     |    39
 please   |    37
(20 rows)
```

/home/w205

3. Postgres Queries

Ad hoc queries to postgres database:

$psql -U postgres

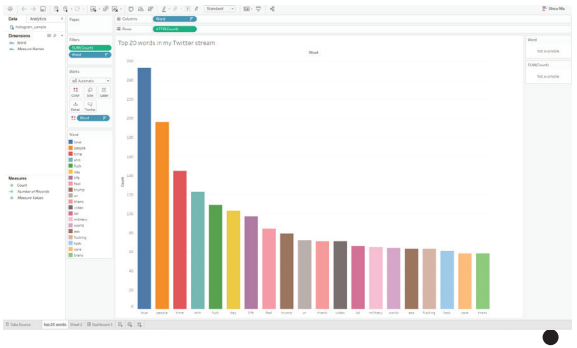
>>\c tcount

>>SELECT * FROM tweetwordcount;

Export data and clear up table for tableau bar chart at end of each day:

$~/w205_2017_summer/exercise_2/write_barchart_data_clear_table.sh

Local Computer



local directory of your choice

4. Tableau Visualization

Make daily Tableau bar chart:

Bring in data as text file, create new sheet. Use dimensions-word as column and measures-count as row. Filter column to give top 20 results

LEGG (HO MAN) YEUNG

IDEA OF APPLICATION: how do tweets become insights?

RUN DAILY TO GENERATE VISUALIZATIONS

STORM

Live Tweets

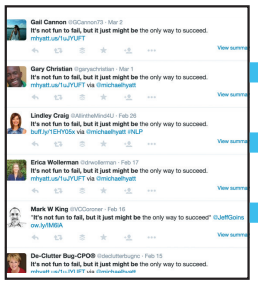
Spouts

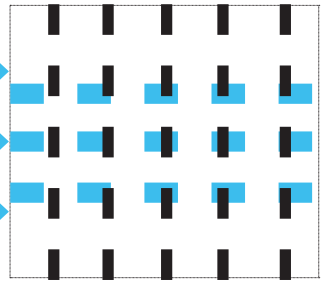
Bolt (parse)
Filters: stopwords,
!@#$%^&*()/\|~

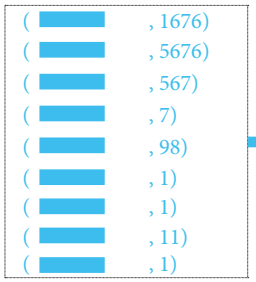
Bolt (wordcount)

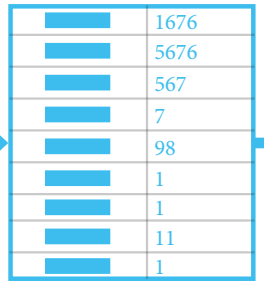
Postgres db
Export daily
Clear up daily

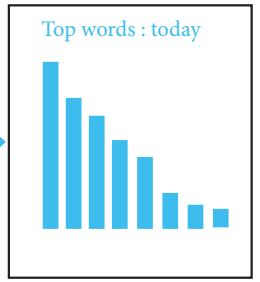
Tableau Reports











Raw Tweets

Split Words

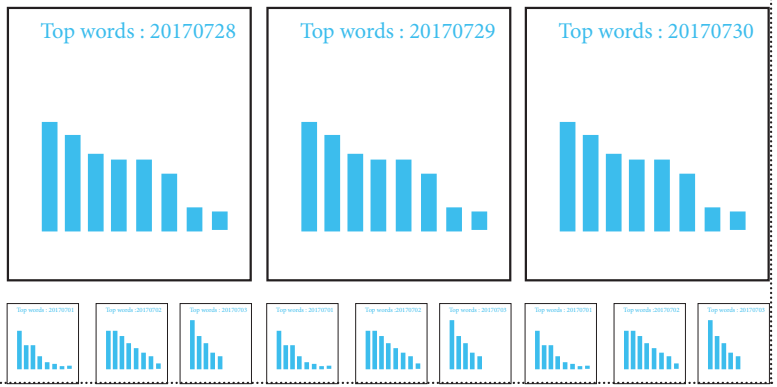
(Word , Count)

Tweetwordcount table

Barcharts (sheets)

COMPARE WEEKLY / MONTHLY
TO STUDY WORD TRENDS

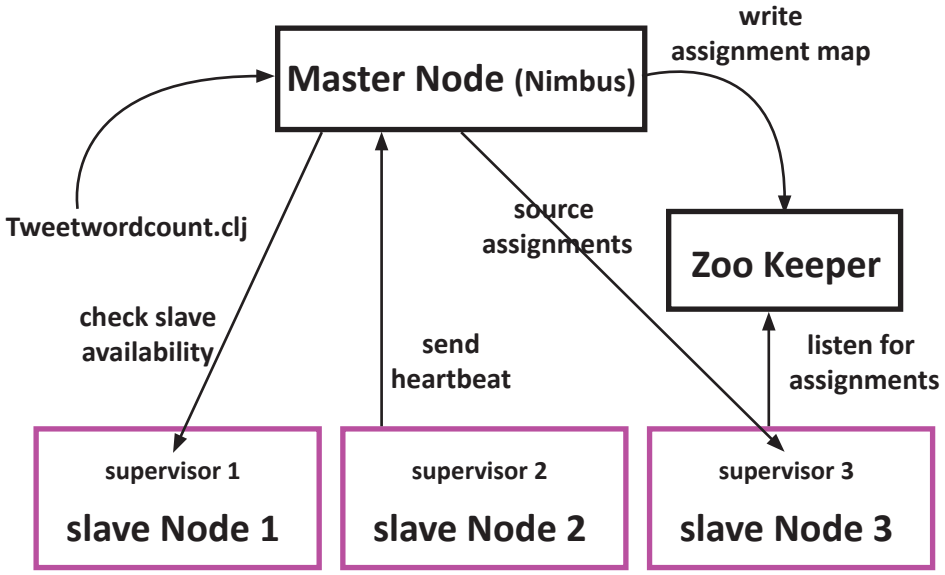
Local Directory / Github Repo



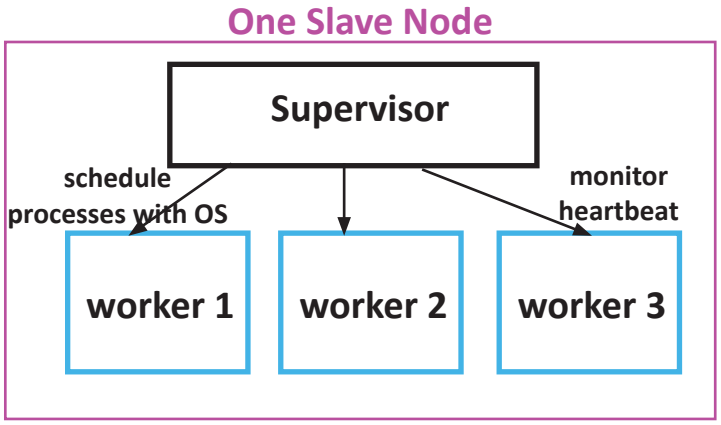
LEGG (HO MAN) YEUNG

STORM ARCHITECTURE

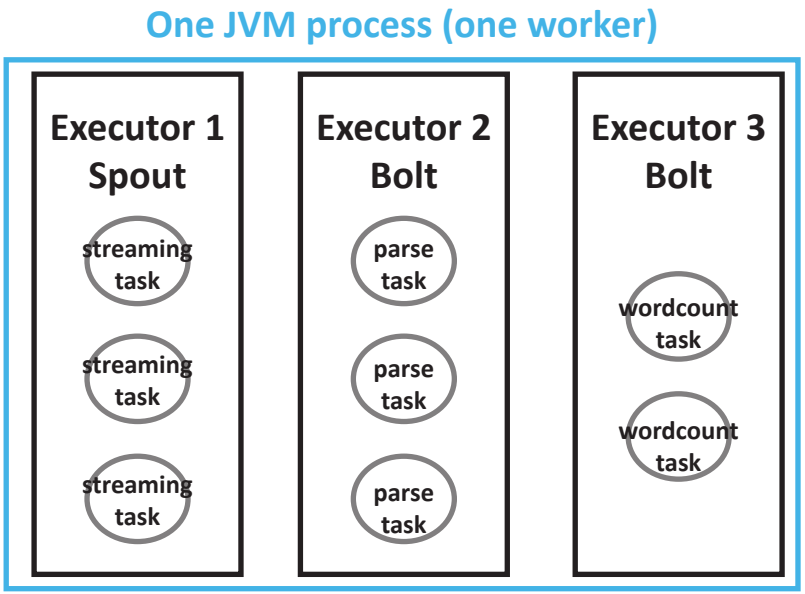
1. Typology is submitted to Master Node, which checks on slave nodes availability and write assignment maps to zoo keeper.
1.1 Supervisors in slave nodes checks assignment map from zoo keeper and source tasks directly from master node.



2.Supervisors monitor heart beat of each worker and report to the master node.
2.1 Operating System Schedule JVM processes among workers.



3. JVM scheduling algorithm schedule executors. Our typology requested one instance (executor) for each typology.



(assume all the tasks in our tweetwordcount typology can be executed in a single JVM process)