

COP 5615 Fall 2014

**Distributed Operating Systems
Alin Dobra**

Project 4

Twitter Simulator and REST API

Submitted By:

Team Member : 1

**Name : Abhishek Singh Panesar
Email : asp.abhi18@ufl.edu
UFID : 5111-8921**

Team Member : 2

**Name : Nishit Sadhwani
Email : n.sadhwani@ufl.edu
UFID : 5393-5598**

Table of Contents

<u>How to run the code</u>	3
<u>Twitter Architecture</u>	5
<u>Twitter Stats</u>	6
<u>Functionalities Implemented</u>	9
<u>Observations and Graphs</u>	9
<u>Output</u>	18
<u>Some Cool Stuff</u>	19
<u>References</u>	20

How to run the code

Normal

You need sbt and Java version 7 (not version 8) installed on the machine. Open two terminals, or run the server part of code on server machine and client part of code on client machines.

For Server machine, open the terminal, navigate to the server folder :

cd //path

and then type the command

run sbt

Now you'll enter the sbt console.

Server:

> compile

> run ScalingFactor Number_of_Client_Machines Time

> run 5 1 60

For Client machine, open the terminal, navigate to the client folder :

cd //path

and then type the command

run sbt

Now you'll enter the sbt console.

Client:

> compile

```
> run      IP_address      ScalingFactor  Event/Regular
```

```
> run 172.16.102.114 5 Regular|
```

* We are reading the tweets from a file, so we need to change the exact path location of where the file is present. So while before running the code on client machine, just change the path of text file (located in the main folder).

Change file path in client code. (Line Number 139)

REST API

The requirements are similar, you need to open up three terminals and cd into 3 folders, namely: Server, RestServer and Client. After that you need to run sbt and compile.

Following this, the commands are written below with an example:

Server:

```
> compile
```

```
> run      ScalingFactor  Number_of_Client_Machines  Time
```

```
> run 4 1 60|
```

Rest Server:

```
> compile
```

```
> run      IPServer IPSelf
```

```
> run 172.16.102.138 172.16.102.138|
```

Client:

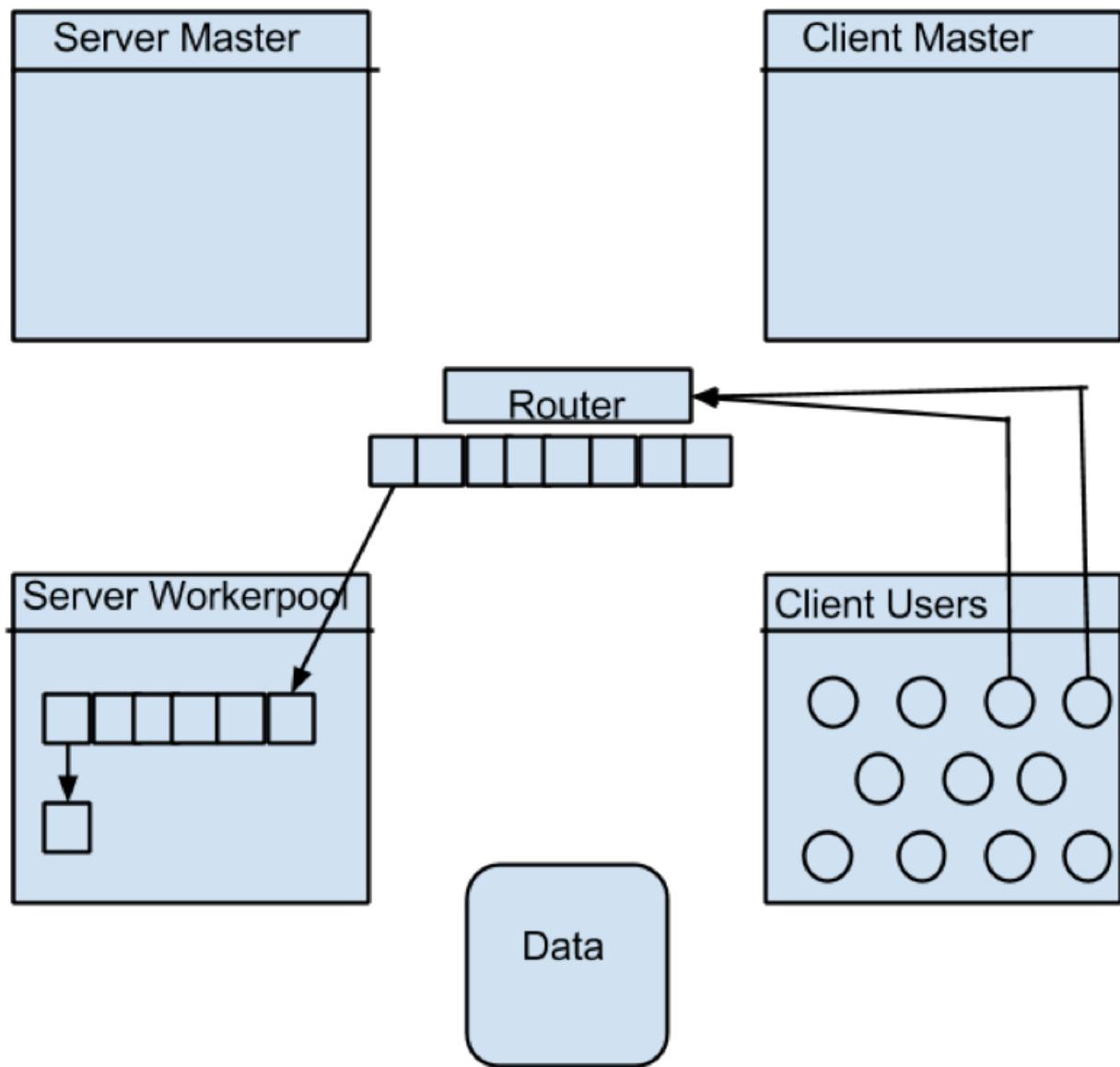
```
> compile
```

```
> run IPServer ScalingFactor Event/Regular IPRestServer
```

```
> run 172.16.102.138 4 Regular 172.16.102.138|
```

Twitter Architecture

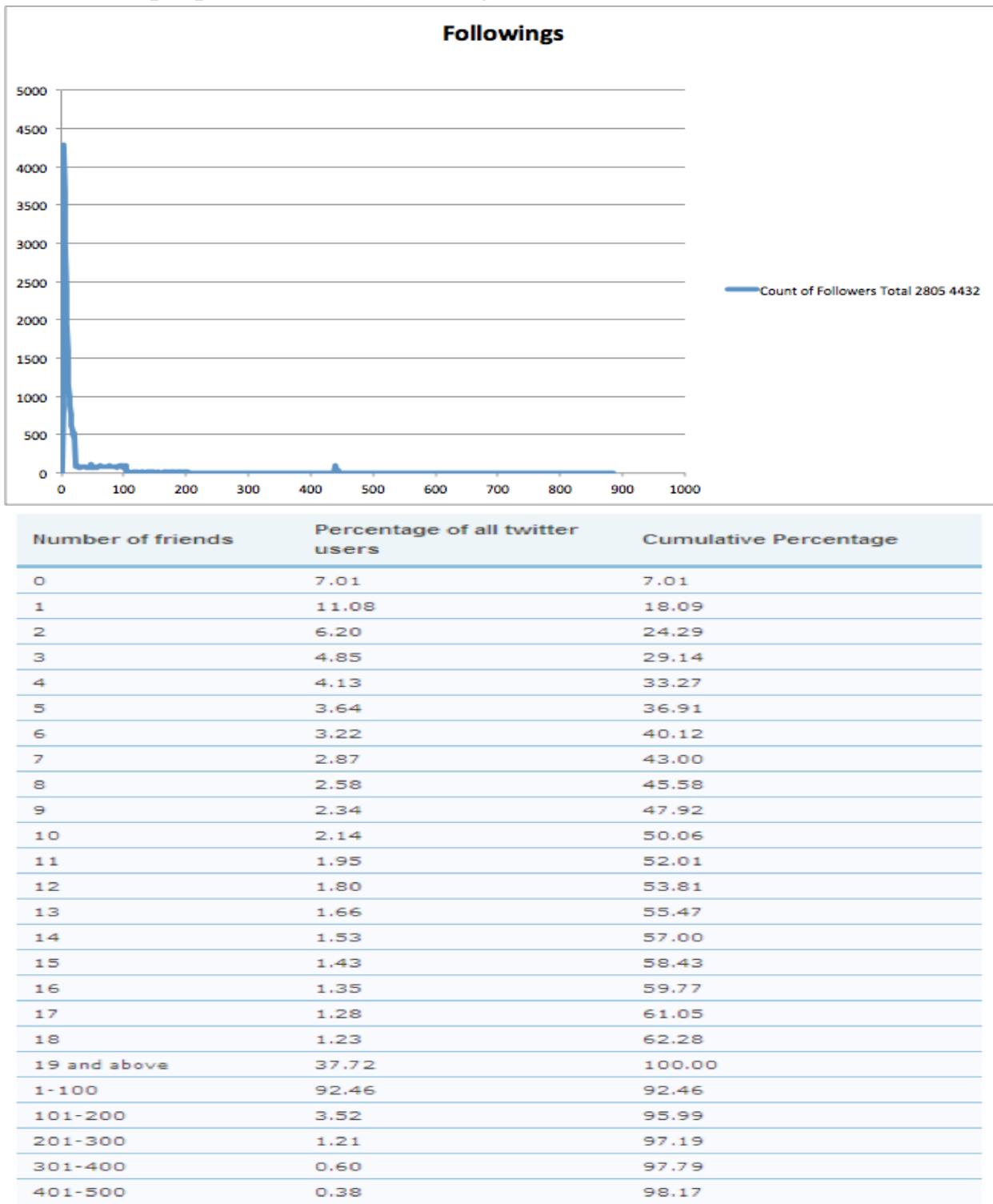
The architecture that we've implemented is shown below:



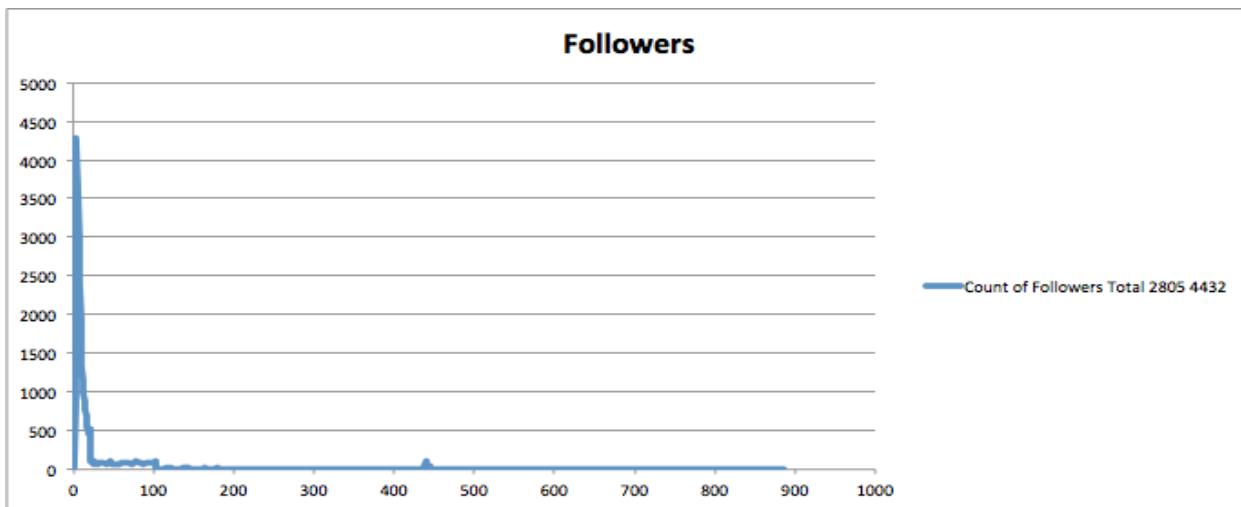
Twitter Stats

The stats we have implemented are:

Number of people followed (Following)



Followers



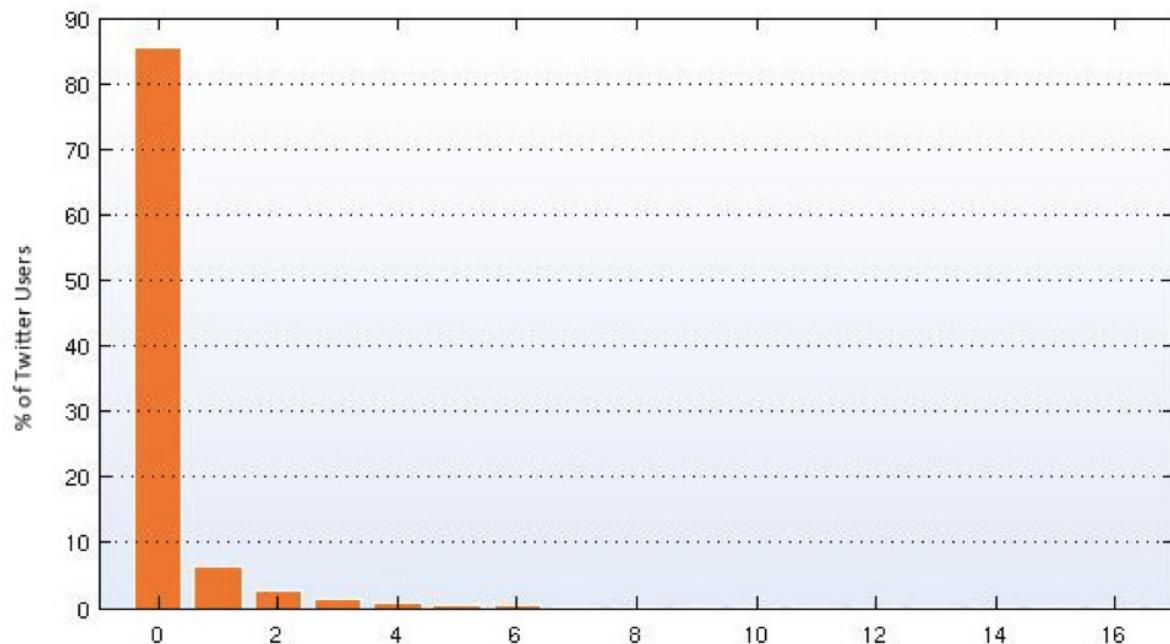
Number of follower	Percentage of all twitter users	Cumulative Percentage
0	2.09	2.09
1	10.73	12.82
2	10.74	23.56
3	9.18	32.73
4	7.45	40.18
5	5.99	46.17
6	4.85	51.02
7	4.01	55.03
8	3.37	58.40
9	2.88	61.29
10	2.49	63.77
11	2.19	65.97
12	1.94	67.90
13	1.72	69.62
14	1.56	71.18
15	1.39	72.57
16	1.26	73.83
17	1.15	74.98
18	1.05	76.03
19 and above	23.97	100.00
1-100	93.68	93.68
101-200	2.75	96.43
201-300	1.04	97.47
301-400	0.57	98.03
401-500	0.36	98.40

Tweeting stats

We implemented two types of tweeting stats namely **Regular** and **Event**.

Regular

Tweets Per Day



Average number of tweets	Percentage of all twitter users	Cumulative Percentage
0	85.37	85.37
1	6.48	91.86
2	2.80	94.66
3	1.53	96.19
4	0.93	97.12
5	0.62	97.73
6	0.42	98.16
7	0.31	98.47
8	0.23	98.70
9	0.18	98.88
0-9	98.88	98.88

Event

In this stats, we bombard tweets every second.

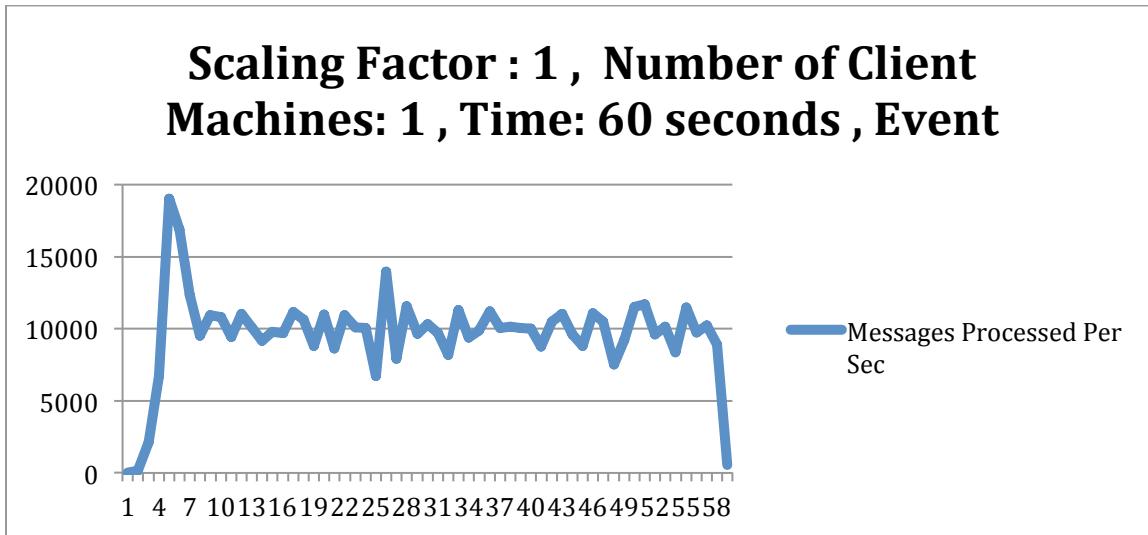
Functionalities Implemented

We've implemented six functionalities, namely :

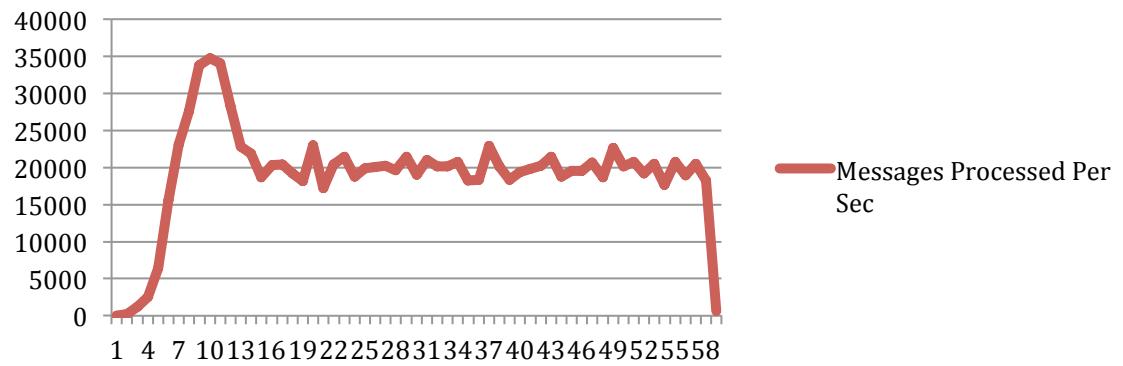
- **Return Followers**
- **Return Tweets**
- **Return Mentions**
- **Re-tweet**
- **Post Mention**
- **Post Tweet**

Observations and Graphs

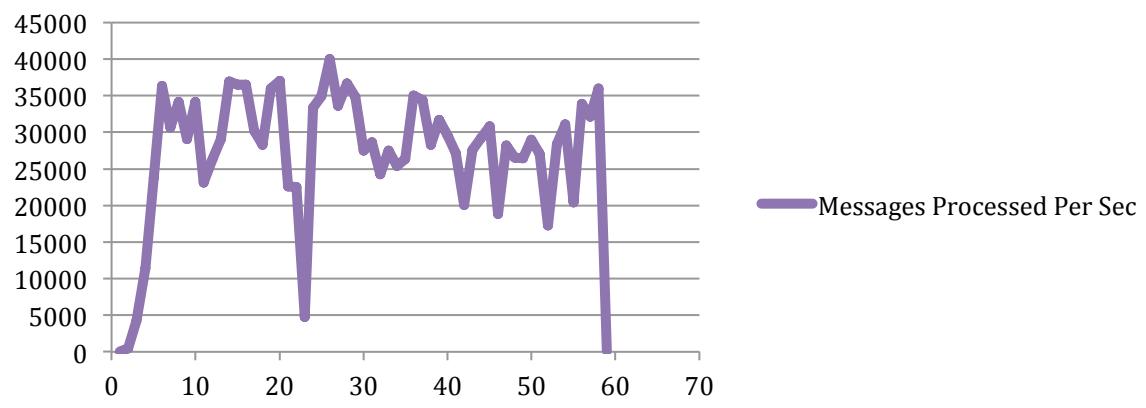
Event Normal



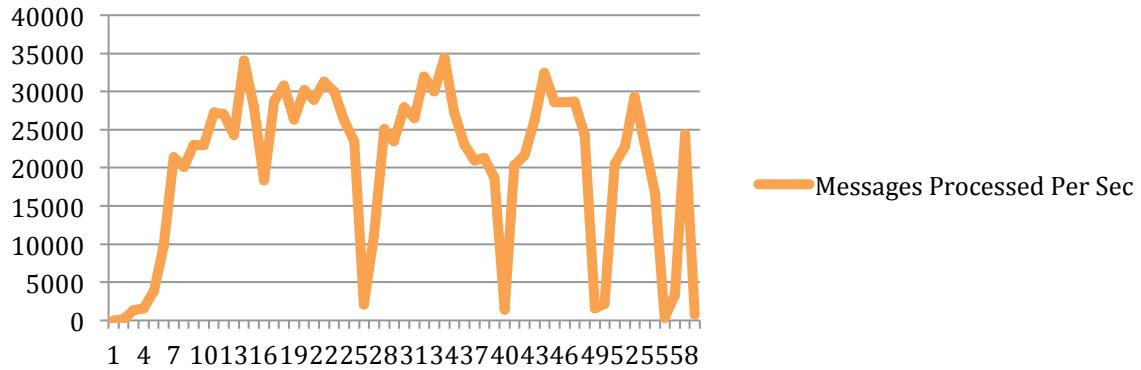
Scaling Factor : 2 , Number of Client Machines: 1 , Time: 60 seconds , Event



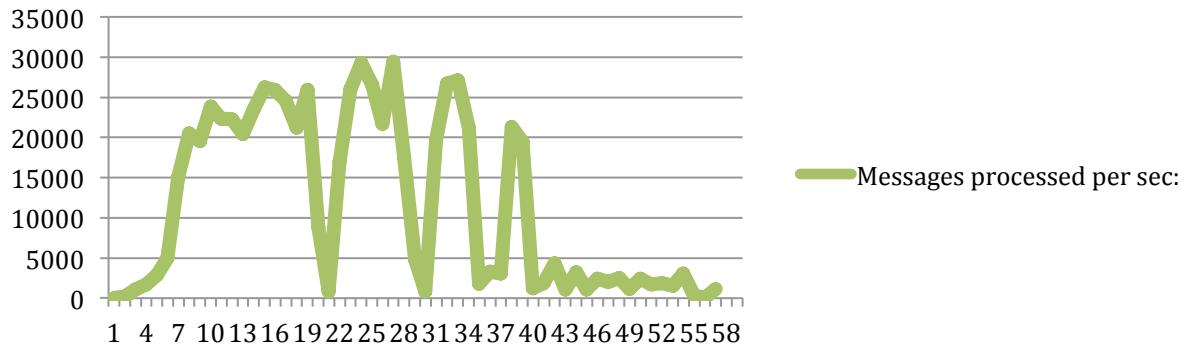
Scaling Factor : 3 , Number of Client Machines: 1 , Time: 60 seconds , Event



Scaling Factor : 4, Number of Client Machines: 1 , Time: 60 seconds , Event

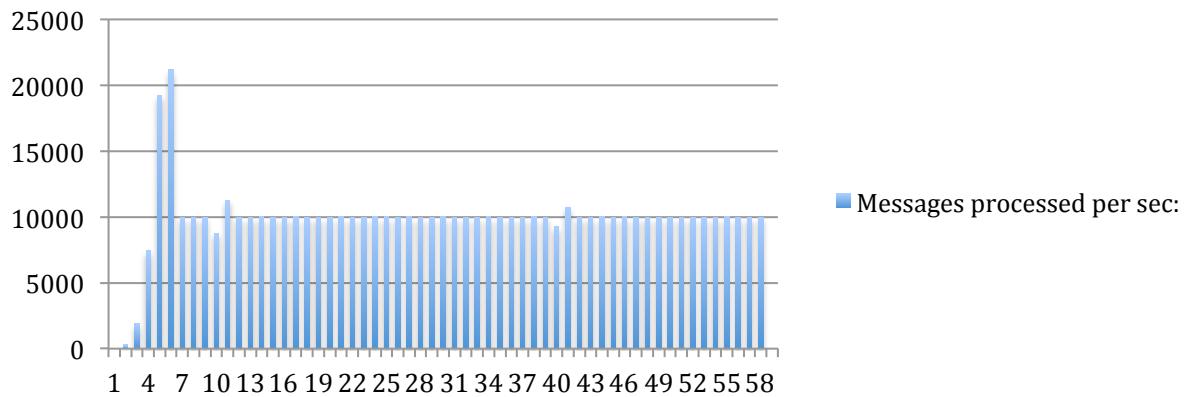


Scaling Factor : 5 , Number of Client Machines: 1 , Time: 60 seconds , Event

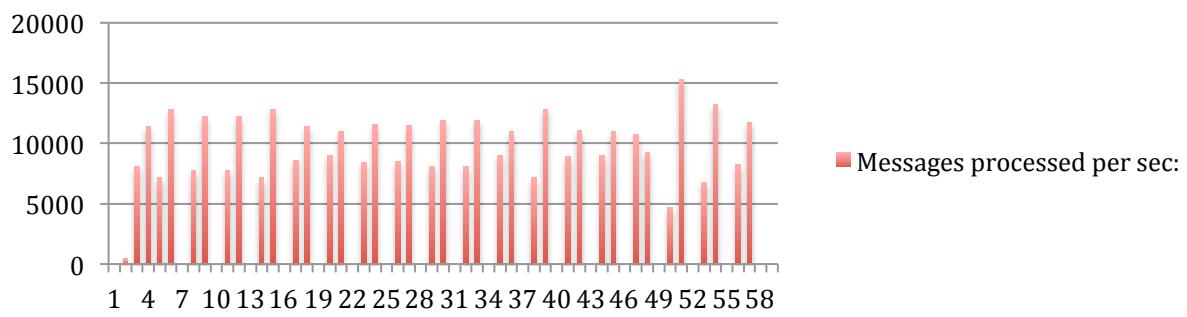


Regular Normal

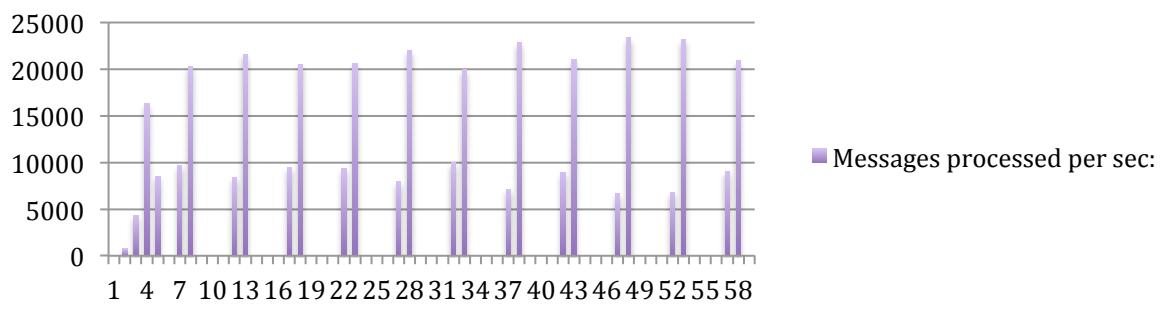
**Scaling Factor : 1 , Number of Machines: 1 ,
Time: 60 seconds , Regular**



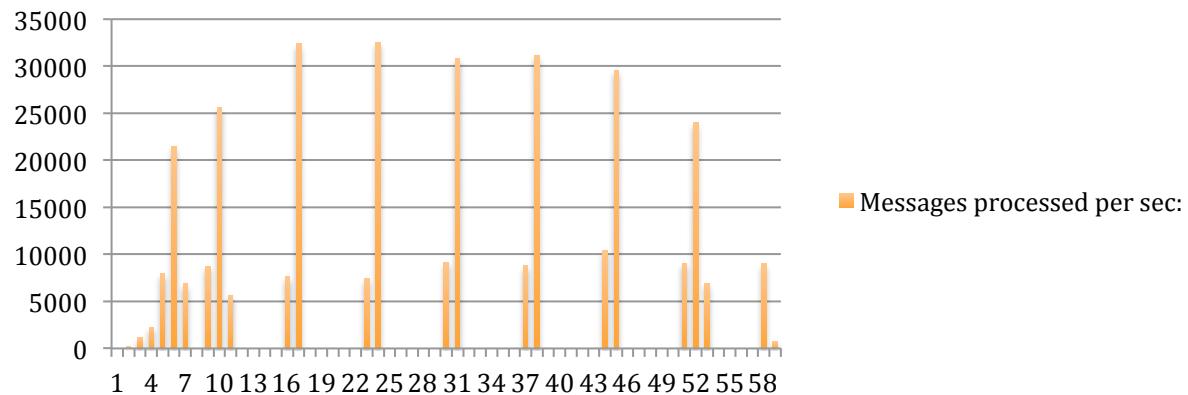
**Scaling Factor : 2 , Number of Machines: 1 ,
Time: 60 seconds , Regular**



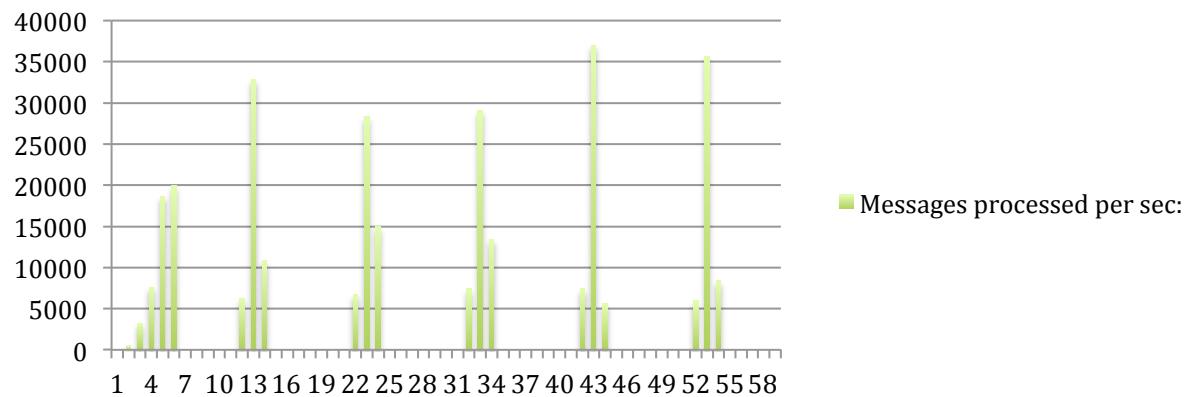
**Scaling Factor : 3 , Number of Machines: 1 ,
Time: 60 seconds , Regular**



**Scaling Factor : 4 , Number of Machines: 1 ,
Time: 60 seconds , Regular**



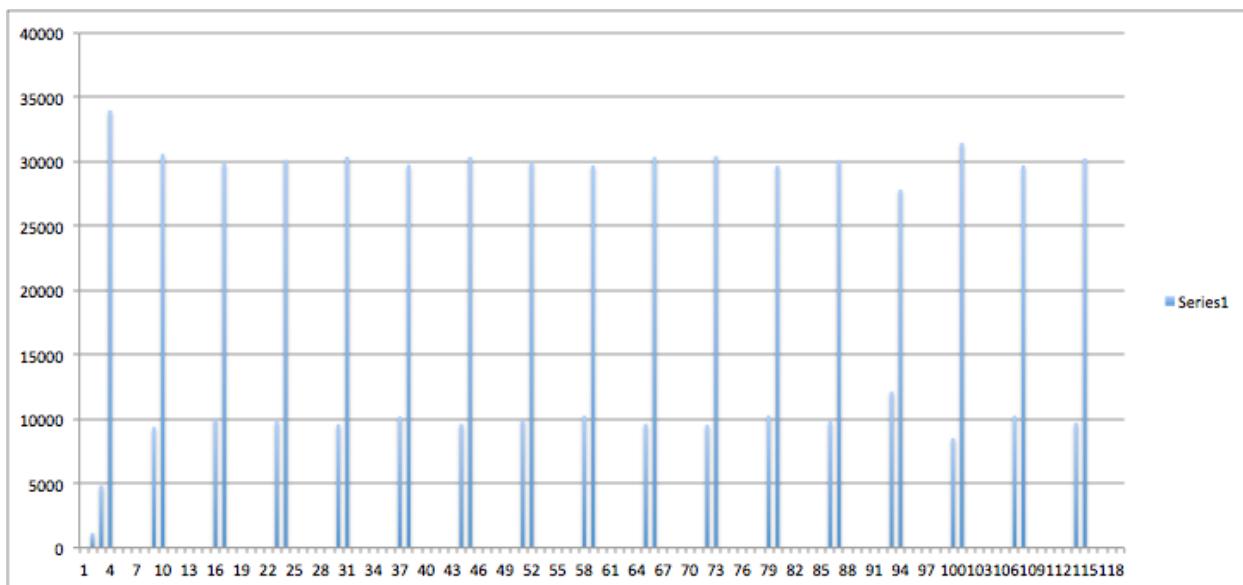
**Scaling Factor : 5 , Number of Machines: 1 ,
Time: 60 seconds , Regular**



The **Regular** tweeting graph is shown below:

Number of users: 40,000

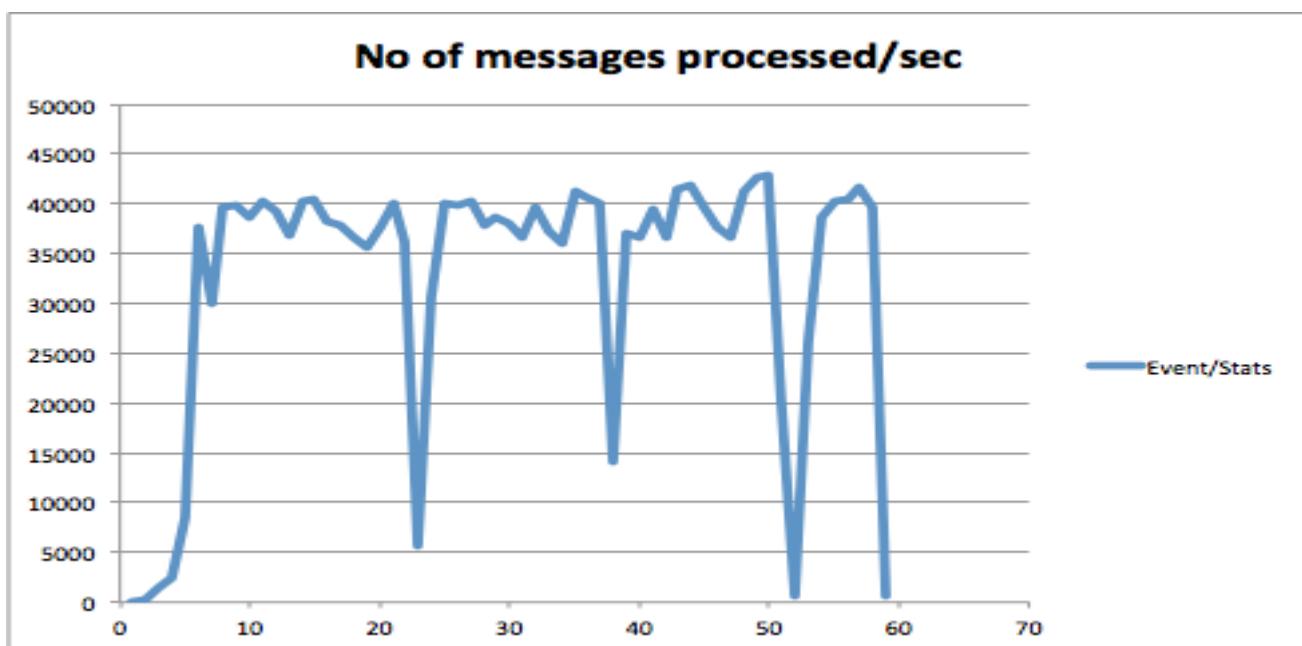
Number of seconds : 60



The **Event** tweeting graph is shown below:

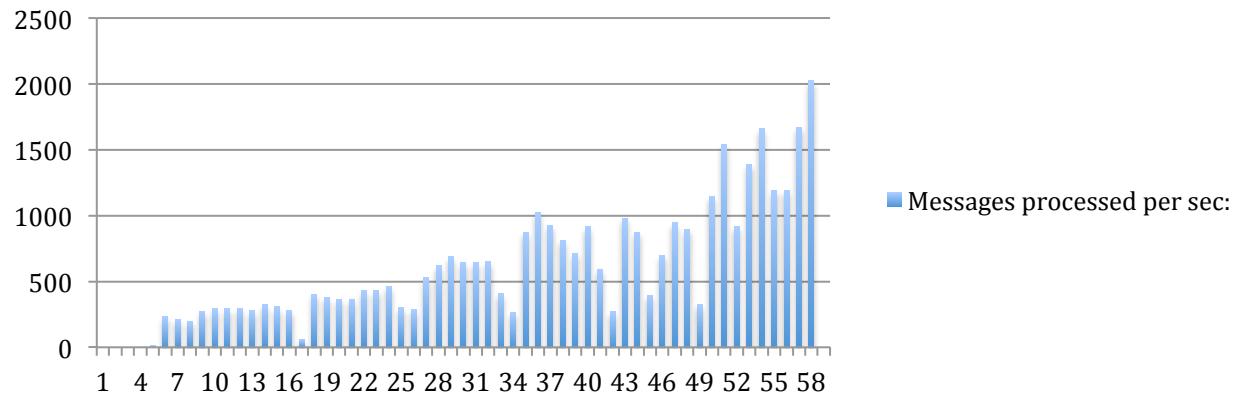
Number of users: 40,000

Number of seconds: 120

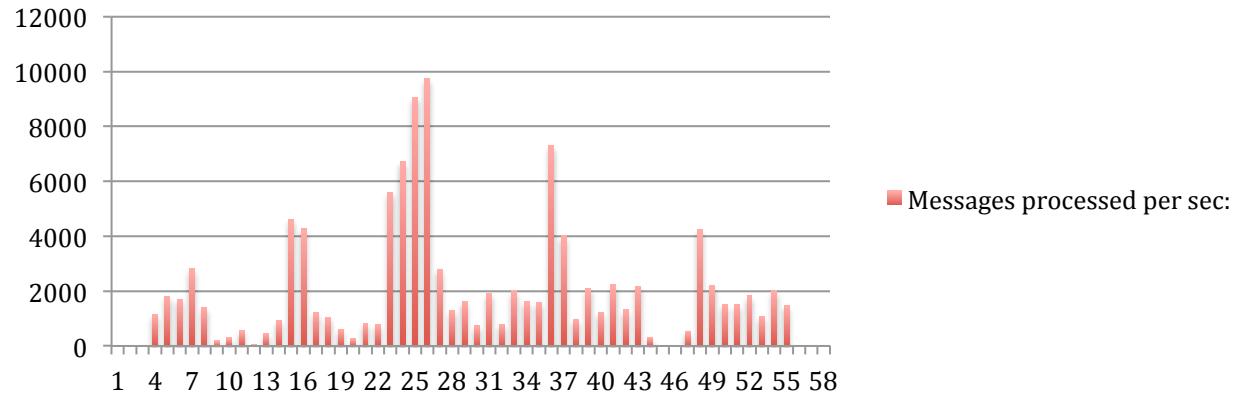


EVENT Rest API

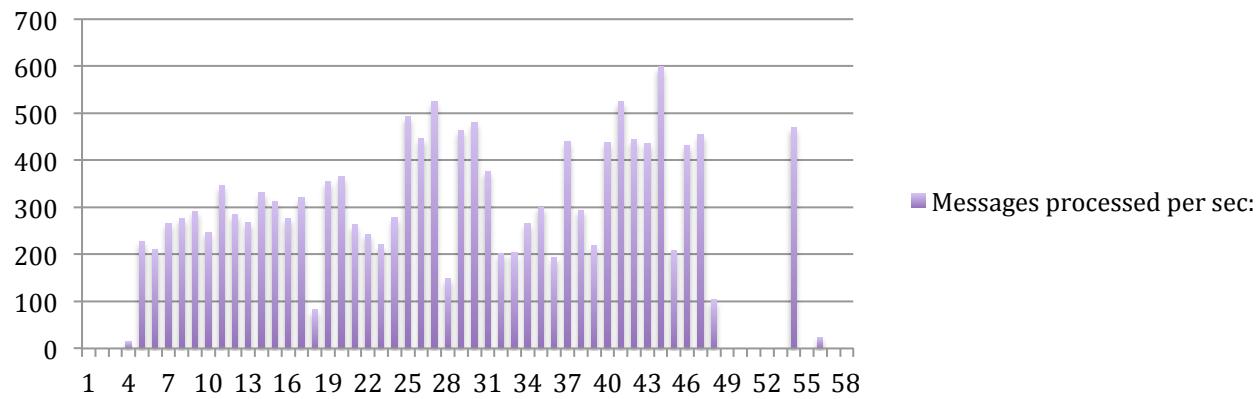
Scaling Factor : 1 , Number of Client Machines: 1 , Time: 60 seconds , Event



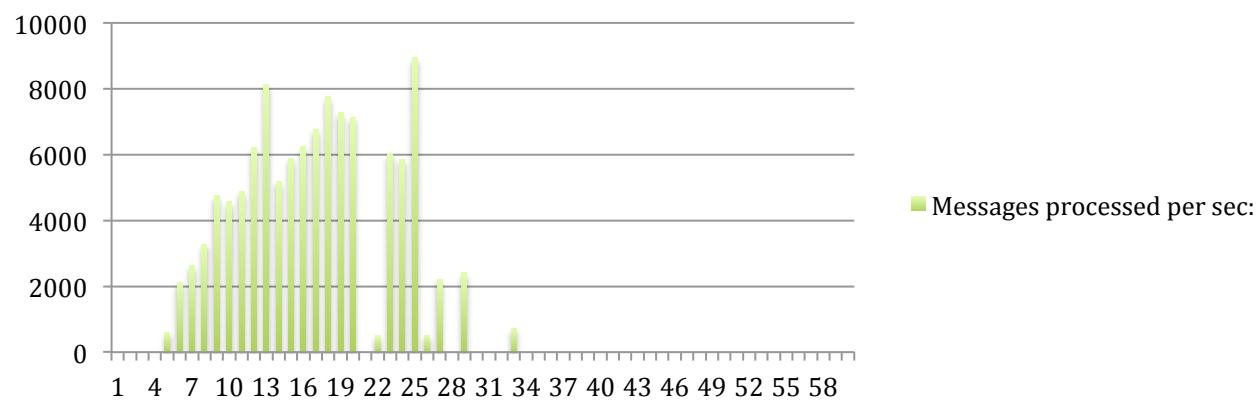
Scaling Factor : 2 , Number of Client Machines: 1 , Time: 60 seconds , Event



**Scaling Factor : 3, Number of Client Machines:
1 , Time: 60 seconds , Event**

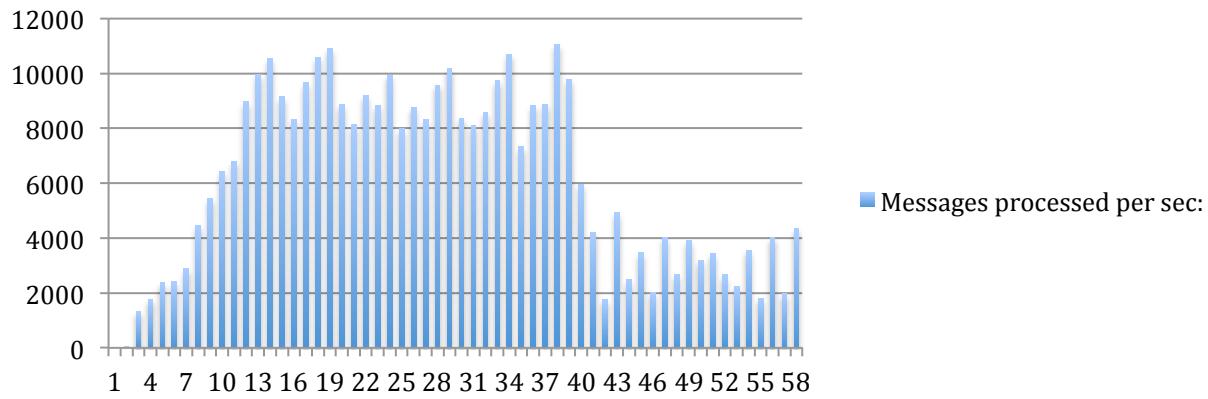


**Scaling Factor : 4 , Number of Client Machines:
1 , Time: 60 seconds , Event**

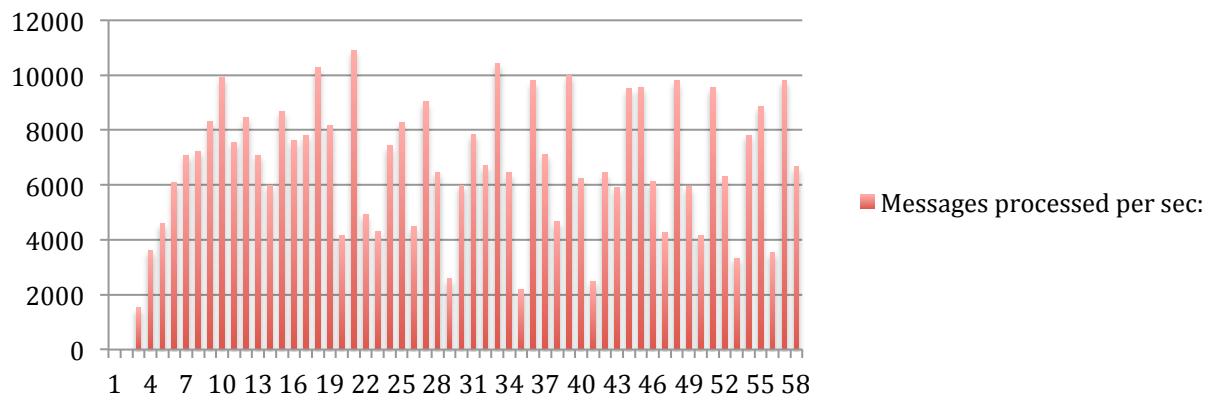


Regular Rest API

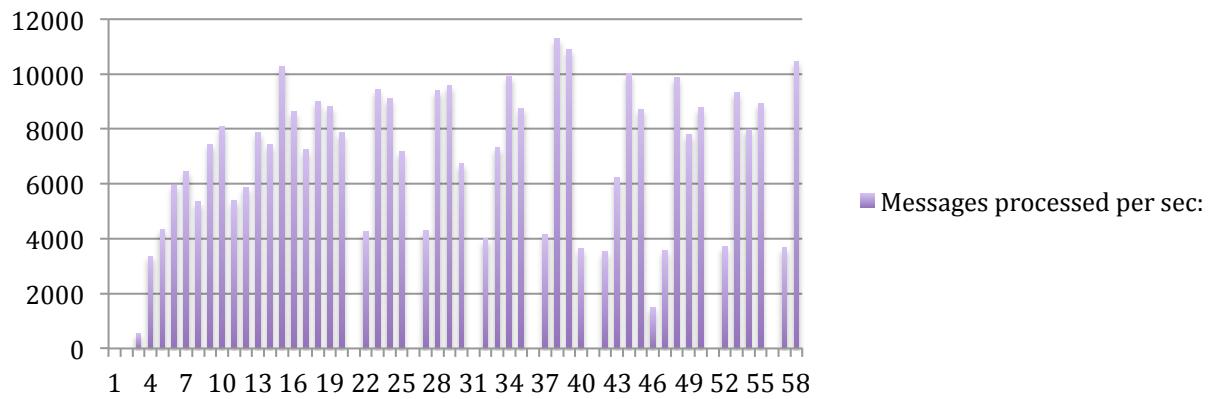
Scaling Factor : 1, Number of Client Machines: 1 , Time: 60 seconds , Regular



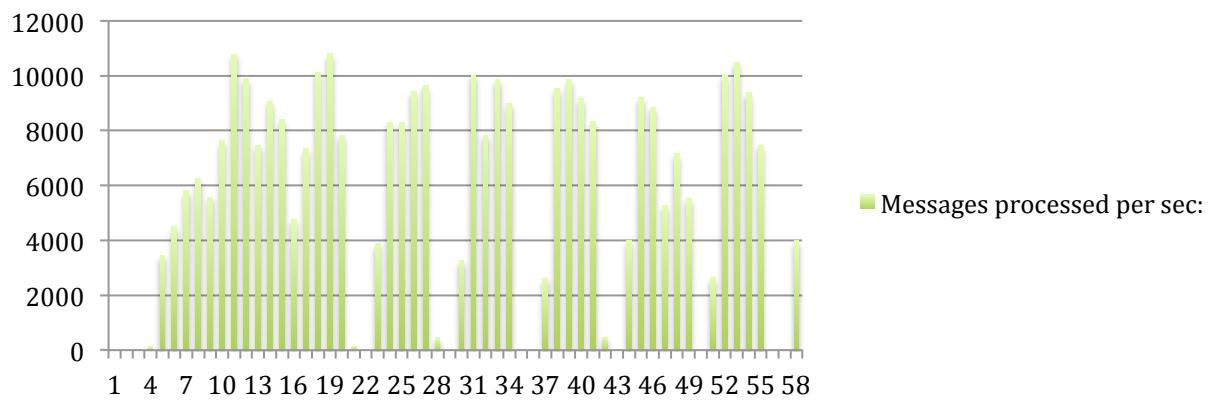
Scaling Factor : 2, Number of Client Machines: 1 , Time: 60 seconds , Regular



Scaling Factor : 3, Number of Client Machines: 1 , Time: 60 seconds , Regular



Scaling Factor : 4, Number of Client Machines: 1 , Time: 60 seconds , Regular



Output

The output screen of running REST API is shown below:

The image shows three terminal windows on a Mac OS X desktop. The top-left window is titled 'AbhishekRestServer - java - 80x24' and displays a continuous stream of 'Tweet....' entries. The top-right window is titled 'AbhishekClient1 - java - 80x24' and shows a log of a client application starting up, connecting to a server, and shutting down. The bottom window is titled 'AbhishekServer - java - 80x24' and displays a log of message processing statistics per second, ranging from 0 to over 10,000. To the right of the windows, a file transfer interface shows files being moved between 'trip' and 'Twitter_FinalPart' folders.

```

AbhishekRestServer - java - 80x24
Tweet....
GetMentions....
Tweet....
Tweet....
Tweet....
Tweet....
Tweet....
Tweet....
Tweet....
GetTweet..
GetFollowers....
Tweet....
Tweet....
GetFollowers....
Tweet....
> []

AbhishekClient1 - java - 80x24
[success] Total time: 63 s, completed Dec 15, 2014 9:32:57 AM
> run 172.16.102.138 4 Regular 172.16.102.138
[info] Running main.scala.Client 172.16.102.138 4 Regular 172.16.102.138
[INFO] [12/15/2014 09:33:35.752] [run-main] [Remoting] Starting remoting
[INFO] [12/15/2014 09:33:35.897] [run-main] [Remoting] Remoting started; listening on addresses :[akka.tcp://TwitterClient@172.16.102.138:52010]
I am here
172.16.102.138
IPAddress of Server: 172.16.102.138
Client Users: 40000
Master reference defined 172.16.102.138
 akka://TwitterClient/user/clientmaster In requestnow
Runningtime: 60
[INFO] [12/15/2014 09:34:38.513] [TwitterClient-akka.remote.default-remote-dispatcher-13] [akka.tcp://TwitterClient@172.16.102.138:52010/system/remoting-terminal]
Shutting down remote daemon.
[INFO] [12/15/2014 09:34:38.514] [TwitterClient-akka.remote.default-remote-dispatcher-13] [akka.tcp://TwitterClient@172.16.102.138:52010/system/remoting-terminal]
Remote daemon shut down; proceeding with flushing remote transports.
[INFO] [12/15/2014 09:34:38.532] [TwitterClient-akka.remote.default-remote-dispatcher-5] [akka.tcp://TwitterClient@172.16.102.138:52010/system/remoting-terminal]
Remoting shut down.
[success] Total time: 64 s, completed Dec 15, 2014 9:34:39 AM
> []

AbhishekServer - java - 80x24
Messages processed per sec: 9899
Messages processed per sec: 7459
Messages processed per sec: 9079
Messages processed per sec: 8405
Messages processed per sec: 4744
Messages processed per sec: 7361
Messages processed per sec: 10118
Messages processed per sec: 10794
Messages processed per sec: 7828
Messages processed per sec: 146
Messages processed per sec: 0
Messages processed per sec: 3872
Messages processed per sec: 8300
Messages processed per sec: 8279
Messages processed per sec: 9426
Messages processed per sec: 9653
Messages processed per sec: 450
Messages processed per sec: 0
Messages processed per sec: 3251
Messages processed per sec: 10011
Messages processed per sec: 7840
Messages processed per sec: 9870
Messages processed per sec: 9001
Messages processed per sec: 0
> []

```

Some Cool Stuff

- Implemented exact twitter statistics.
- The average number of tweets per second is greater than 30,000.
- The max number of tweets per second that we recorded were 53,000 tweets for 1,50,000 users.
- The people who have more followers tweet more, which is true in the real scenario as well.

References

www.sysomos.com - For Tweeter Stats