**Twitter Simulation Project**

**Input Parameter:**

Tweets per Second, Number of Users, IP Address of Server

**Client Simulation:**

* The given statistics were gathered from the Internet:

1. Avg. # of tweets per user - 307 (Follows an Exponential Curve)
2. Avg. # of Followers per User – 208 (Follows an Exponential Curve)

* Based on above statistics we first generate the number of tweets and followers of each user using a probabilistic distribution.
* We generate the social graph based on the followers per user.
* We then distribute the tweets per user over given duration to match the desired Tweets per Second. This is done using a normal distribution, which also generates a degree of variance around the mean.
* The event timeline generated per user is then synchronized and played out using a scheduler.

**Server Simulation:**

We tried two approaches and observed roughly the same performance.

**Approach 1:**

* Use a SmallestMailBox Router.
* The Routees can independently handle any incoming request to write or read tweets.
* All the Routees write to a concurrent and synchronized List, which maintains the Home Timelines for each user.

**Approach 2:**

* Use a Consistent Hashing Router.
* The Routees store the social graph data for a subset of users.
* We have a set of Actors for the Timeline Service, which is partitioned, based on userId. The Routees then forward the write (or read) requests to the corresponding Timeline Service.
* There is no shared global synchronized list for the Timeline. This is more in spirit with the Actor Model.

**Performance Observations:**

System Configuration: 1.4 Ghz Dual Core i5 Processor, 4GB RAM

* We were able to sustain write rates of 5000 tweets per second among 5000 users.
* We were able to sustain read rates of 25000 requests per second among 5000 users.
* Memory Requirement – Heap Space Required was 1GB.

**Twitter API Project**

**API’s implemented:**

* POST a Tweet (/tweet)
* POST a direct message (/msg)
* GET direct messages (/msg/{userId})
* GET Home Timeline (/home\_timeline/{userId})
* GET User Timeline (/user\_timeline/{userId})
* GET User Profile(/user/{userId})
* GET Followers (/followers/{userId})
* GET Friends (/following/{userId})
* GET Mentions (/mentions/{userId}

**Test on Large Cluster**

* ab -n 1000000 -c 1000 -T "Content-Type: application/json" -p post\_data.json <http://192.168.1.3:8080/tweet>

Requests per second: 11473.14 [#/sec] (mean)

Time per request: 87.160 [ms] (mean)

* ab -n 1000000 -c 1000 <http://192.168.1.3:8080/home_timeline/9000>

Requests per second: 22010.97 [#/sec] (mean)

Time per request: 40.634 [ms] (mean)