

# **PROJECT 4-PART I**

## **TWITTER- SIMULATOR**

Lavanya Chennupati(UFID: 1129---9429)

Sowmya Duggimpudi(UFID: 1171---2252)

## **OBJECTIVE:**

To simulate twitter with real statistics using Scala-Akka actors. Main objective is to check the throughput of the server.

## **WORKING:**

### **Functionalities implemented:**

- User tweeting
- Home TimeLine Display
- User TimeLine Display

### **Server:**

Server maintains the entire user list, each user's followers and the tweet messages of every user and whom he follows. Whenever requests(tweet/home timeline display/user timeline) from a client are made, these requests are routed to several akka actors where the request is processed depending on the availability of the actors.

### **Client:**

The Client makes a request to the server whenever a user tweets or makes a request for home timeline or user timeline display. The requests are routed among the actors of the client before sending to the server.

## **STATISTICS USED:**

81.8% of the users have 0-50 followers.  
6.3% of the users have 51-100 followers.  
9.2 % of the users have 101-500 followers.  
2.7% of the users have above 500 followers.

Based on this information, the user and their follower lists are updated.

As of now there are 300 million twitter users.  
The number of tweets in a day are 500 million  
The number of user timeline requests in a day are 216 billion  
The number of home timeline requests in a day are 650 billion

Based on this information we generated the number of requests from the client.

## **COMPILE AND RUN:**

Server:

Compile: scalac twitter.scala

Run: scala twitter.scala <num of clients>

Client:

Compile: scalac twitterClient.scala

Run: scala twitterClient.scala< num of clients><server IP>

## **RESULTS AND OBSERVATIONS:**

We are printing the number of requests processed by the server every second.

We have observed that the peak throughput is 1300 requests/sec when the number of users are 100.