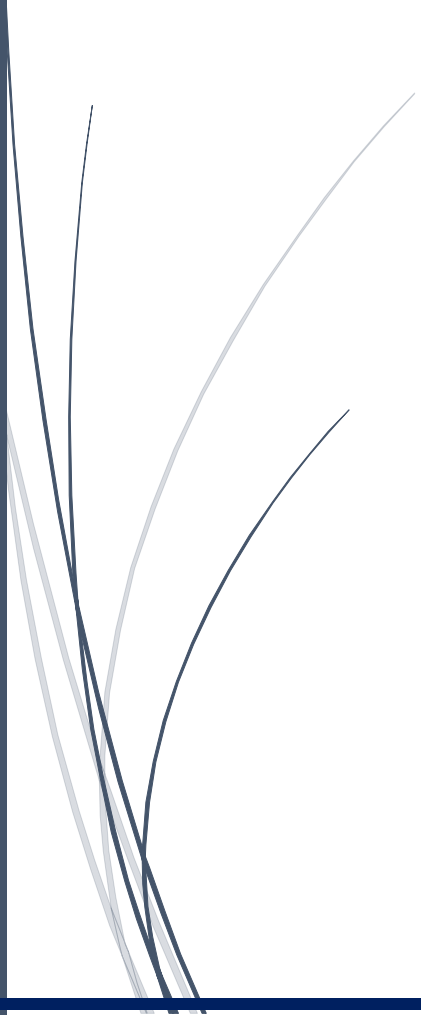
COP 5615

FALL 2014

Project 4: Twitter Simulator

Part - I

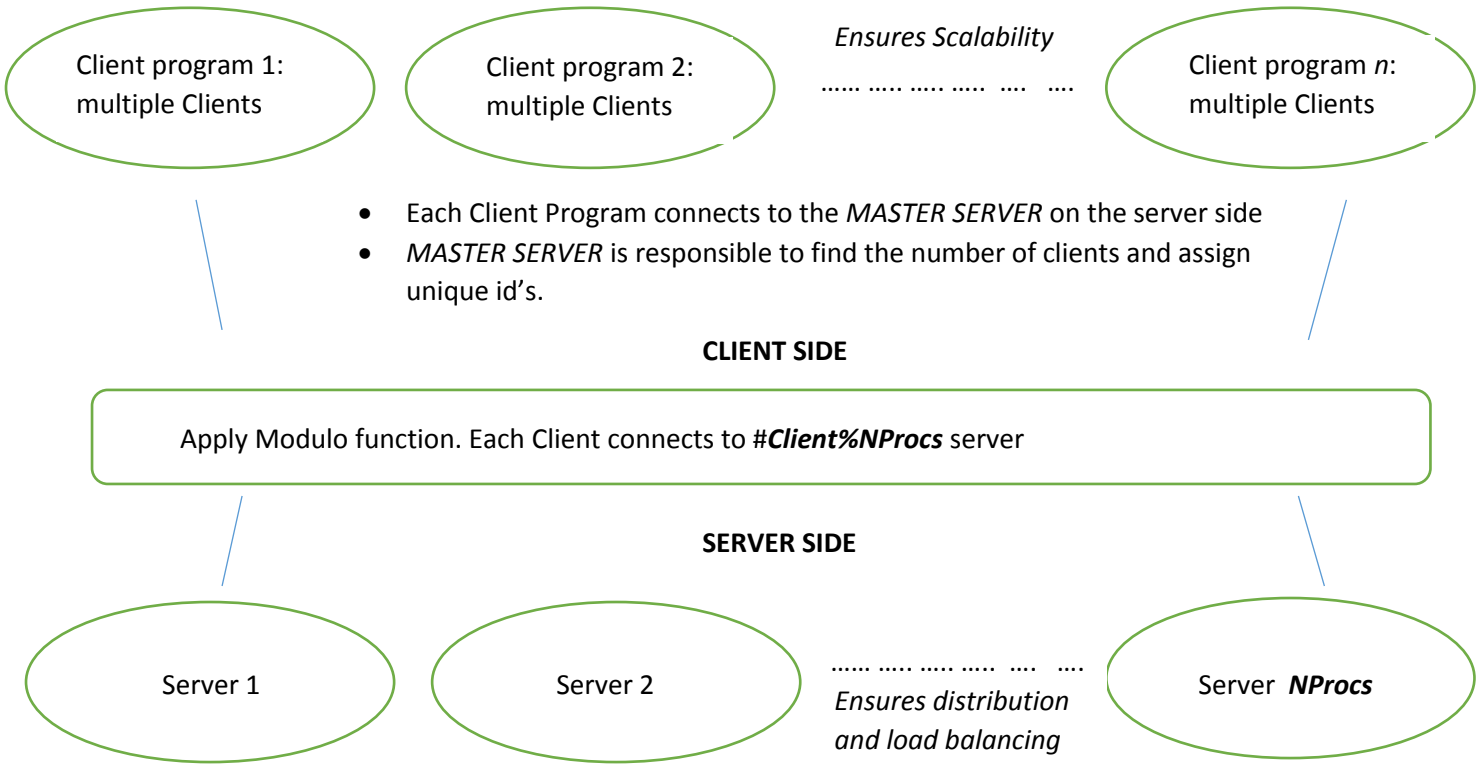
SANCHITA ROY	**9570-7915**
SRITAPA DUTTA	**9896-2658**



Program Design

- Multiple Clients can hit the twitter server at any time – scalability considered
- Tweets and Followers are distributed across the cores of the server – distributed architecture/load balancing

The design:



The algorithm:

1. Each client connects to #client%NProcs server where NProcs is the number of processors in the server side
2. Foreach(client c : c ϵ numClients) {
   FollowPeopleAndTweet () =>
   List *follows*: List[random **Integers** in range (0 to numClients)]
   Send *follows* to server # x%NProcs
   Sleep(250ms)
   Tweet()
   Sleep(1000ms)
   askForTweetsItCanSee()
   }

3. Each server s contains *AllClient : List[ActorRef]*
   Foreach(server s : s = c%NProcs && s ∈ NProcs) {
   Add to *FollowersMap* = map[ClientID, List[FollowingClient's Index in *AllClient*]]
   receive case Tweet()
   Add to *TweetMap* = map[ClientID,Queue[TweetQueue](100)]
   receive case askForTweetsItCanSee()
   prepare *clientTweetQueue* =>
   forEach(client && its followers from *FollowersMap*)
   add to ownTweetMap = map[ClientID,Queue[TweetQueue]]
   send *clientTweetQueue* to sender client
   FollowPeopleAndTweet()
   }
4. Program is not designed to terminate.