CHB = interval of heartbeats
SHB = interval of refresh/sync
Default Address = localhost, 0.0.0.0

Design Document (1)

Below, the structure and details of implementation for each of the main processes are described. All are run using the run.sh script provided.

Coordinator Default Port: 9000

The coordinator is responsible for maintaining live data structures organizing client/server/synchronizer connections, and handling various retrievals/updates to this info. The main functionality is in the bidirectional stream rpc to HandleHeartbeats. This call adds servers and synchronizers to the registry, and maintains pings (Heartbeats) between itself and the caller every CHB seconds, configurable via run.sh. When a server does not respond it's status is set to inactive (0).

Client

The client is the process that connects to the servers, and provides the IF for the TinySNS. Initially it connects to the coordinator and recieves the address of the highest prio online server. If it ever disconnects from this server a thread is launched to continually attempt reconnection with the master server. Disconnects are detected when commands fail, and the user is reprompted after reconnection.

Server Default Port: 9020 +

The server process has two types: Master & Slave. Master servers are the default to maintain client interactions, and forward all state changing actions to the corresponding slave server in the cluster. The handle local changes caused by client calls to Login, List, Follow, & Timeline. The basis for these functions is maintained through corresponding files on each server. These files are synced every SHB seconds by the synchronizer. Structure for each call is largely the same as in mp_2, but using the file system instead of a local data structure, to keep changes persistent.

Continued on next page...

CHB = Interval of heartbeats
SHB = Interval of refresh/sync

Design Document (2)

Synchronizer   Default Port : 9001 - 9014

The synchronizer is responsible for synchronizing the file system between clusters, using a primary thread that first syncs the local data structures responsible for keeping updates to the global list of users and their relations (who follows who) and then second, updates the files of the servers with corresponding changes to followers and posts by users a client is following. This is accomplished in 3 steps:

1) Server syncs all users, their relations, and their respective cluster locations.

2) checks "newpost" file for each user in its own cluster, and sends them to other clusters if necessary. based on current followers.

3) Writes new posts to the "updates" file of the following user, and their timeline. This action is copied between master/slave servers.

The "updates" file functions as a queue of new, unseen posts from users, and is cleared after being written to clients. The timeline for a user keeps a permanent record of all a users posts and updates. This file is read to users when they enter timeline mode (The most recent 20 posts, from most recent to least recent).

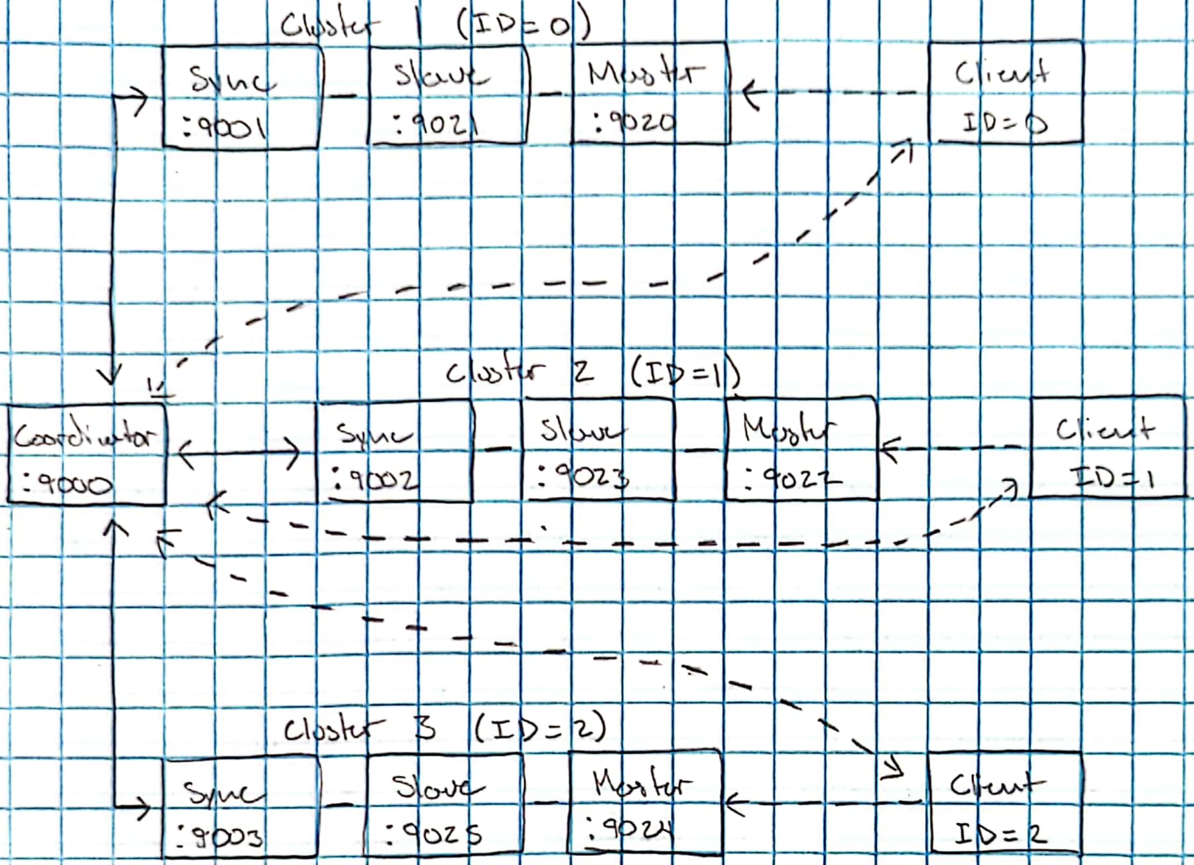
All file reading and writing is thread safe using a mutex lock. The threads are largely detached to operate in parallel.

Script Usage

Both CHB & SHB default to 1 second if arguments unspecified in run.sh command. All refreshes will be effectively instant.

Command	Parameters		Purpose
sh run.sh	CHB	SHB	Runs all background processes, initializing clusters
sh kill.sh			Kills all background processes and clients
sh clean.sh			Removes all server generated files (reset)

Design Document (sketch)



- Each server keeps an updated copy of the auxilliary Files via the synchronizer.
- Client will immediately reconnect to next available online server if connection fails.
- Files and data structures are persistent, server shut down and restart maintains state.