

Keys to be put in the Chord for a user, when he signup or sign-ins.

{userID: P , password}

{userID: FN , fullName}

{userID: UID ,REG:<userID>} //used in retrieving the registered users

{userID: IP, IpAddress}

{nodeID.toString() , nodeID}

{userID: ChatPort , chatPortNumber}

{userID: ConfPort , ConferencePortNumber}

{userID: mgmtPort , managementPortNumber}

1. Each live user in the chord maintains following tables
 - 1.1 chatTable :: Holding the communicating sockets for the users with whom this user chatting.
 - 1.2 confTable:: Holding all the conferences this user participating.
 - 1.3 confOutSocketsTable :: The communicating sockets of users with whom this user sharing the conference.

2. Conference :

ConfListenerServerThread will start listening on ConferencePortNumber. When a request to join the conference comes, ChildConfListenerThread will be started for each request.

The childConferece Thread will accepts following messages

1. InviteMessage: When some one invites this user to join the conference.
2. InvitedMessage: When someother user invited someother user into the conference.
3. ByeMessage: When somebody left the confence.
4. Message: When a conference user wants to test the aliveness of this user.

A conference object is created for each conference this user participating. Conference objects maintains a fingerTable. A fingerTable will have finger entries like chord. Whenever a request to join the user comes the code will put the inviter and all other partictiapping users in the fingerTable. For each interval if atleast one user is present, the application will select one user as the leader for that fingerEntry.

2.1 HandlingInviteMessage:

When an Invite message comes , the thread will create a conference Object and stores in the ConfTable. InviteMessage contains users who are already present in the conference. The thread will add them to the fingerTable of the conference object created and makes connections appropriately as explained above.

2.2 HandlingInvitedMessage

When the thread receives this message it adds the invited user into its fingerTable . And forwards the message to the other users present in the InvitedMessage it has received based on the users's ids and this conference fingerTable .

2.3 HandlingByeMessage

When the thread receive this message it removes the user who is leaving the conference and selects if any leader for the fingerEntry in which this leaving user present. Then it forwards the message to the other users present in the InvitedMessage it has received based on the users's ids and this conference fingerTable .