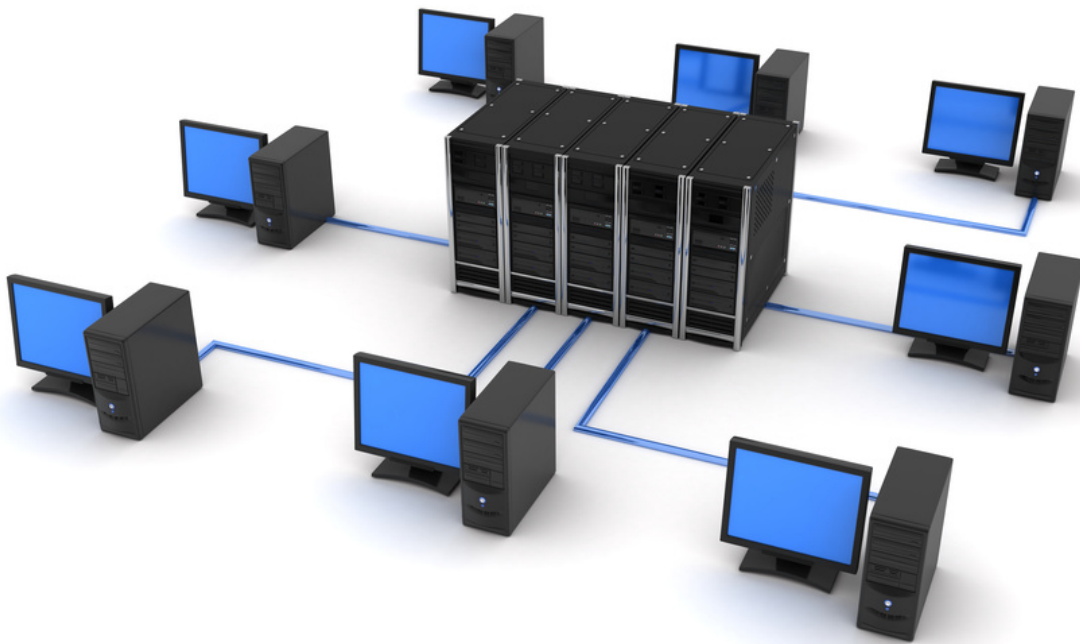


# Report

## *P2P Circular DHT*



Victor Choudhary

# Report Statement

## • Program Design

There are 5 major Classes which contain the program design :-

Name of Class	Working principles
<b>cdht.java</b>	Responsible for starting the network
<b>Peer.java</b>	This class initialises the peer when provided with 3 parameters ( identity, Successor 1 & Successor 2 ). It also maintains the connection sockets and serves as abstract datatype.
<b>PingServer.java</b>	This class starts the ping mechanism and starts a tcp client to predecessor when response message is received
<b>TCPServer.java</b>	This class is used to start TCP Server for the peer.
<b>TCPClient.java</b>	This class is used to contact Predecessor and other Peers for message passing & receiving.

## • How the System Works ?

- ◆ When *cdht.java* program is run with the 3 essential parameters, it initialises the object 'Peer', which immediately creates a TCP & UDP Socket and has various methods required by peer for its functioning at various stages of assignment.
- ◆ Next, we run 2 different threads to start PingServer and TCPServer.
- ◆ PingServer maintains the alive successor and predecessor count, by self defined ping mechanism. Whenever it finds a predecessor, it becomes a TCPClient for predecessor and listens for its message via TCP.
- ◆ TCPServer starts a server for peer and waits for any TCPClient that wants to connect to it.
- ◆ TCPClient is a independent class which can be launched by any section of classes to establish a tcp connection to desired peer.

### • Message Design.

#### ♦ Ping Mechanism :-

- ♦ Peer send's a ping request by sending message : “**\$-PREx-\$A Ping Request**”, where x is the predecessor number with respect to target peer.
- ♦ When a peer receives ping request message, it responds by sending message : “**A Ping Response**”, to the same peer.

#### ♦ Request Mechanism :-

- ♦ Queried peer sends message “**\$Request\$Filename\$RequestingPeer\$**” to his successor via TCP Connection, here ‘Filename’ & ‘RequestingPeer’ are desired parameters which are sent.
- ♦ If the receiving TCPCClient contains the file, it will revert to the Queried peer a response message “**Answer**”.
- ♦ Else if it doesn't contain the file, it will pass the same message to successor via TCP connection.

#### ♦ Quit Mechanism :-

- ♦ Queried peer sends message “**Quit S1 S2**” where s1 and s2 are the identity of the successors.
- ♦ Predecessor's revert to departing successor with message “**OK**”, which makes the end of departing peer.

### • Possible Improvements.

There are many possible improvements which can be done to enhance the working of this peer network, but the most important according to my perspective is to allow a independent joining mechanism. i.e. we should enhance the ability of peer's able to join the new network without being initialised at the beginning of the network.