AOS Project 3 - Napster-style peer-to-peer (P2P) file sharing system

-KIRAN PATRUDU GOPALASETTY

INTRODUCTION:

Peer-to-Peer(P2P) Technologies are being widely used for sharing the data between the servers and the clients. One of the major technology for file sharing that is implemented nowadays is the Napster-Style Peer-to-Peer File Sharing System.

The older versions of the systems used to have a single server which stores the files in its directory that are received from the clients. The major drawback of these systems was that if a new file has been created in one of the peers, it must be transferred to the server before another peer can access it, which delays the process of transfer from one peer to another. This can be conquered using the Napster system which allows the peer to peer file transfer.

SYSTEM ARCHITECTURE:

Processor Type: Intel®Core™i5-7200U CPU
Frequency: 2.50GHz
Memory: 6GB

SYSTEM REQUIREMENTS:

- JDK and Java to be installed
- Server to execute the program on multiple systems

DESIGN:

Entire project is designed using Java where I have used the concepts of Socket Programming and Multi-threading. For establishing the connections between the Server and the Clients, I have used TCP/IP protocol using the sockets.

Major Components of the Project:

- Server and
- Client

Server (Central Index Server):

This server indexes the content of all the peers (i.e., Clients) that register with it. It also provides search facility to peers.

Server Functionalities:

- Registry and
- Search

Client:

As a client, the user specifies a file name with the indexing server using "lookup". The indexing server returns a list of all other peers that hold the file. The user can pick one such peer and the client then connects to this peer and downloads the file.

Major function of the peer:

- Download

As a server, the peer waits for requests from other peers and sends the requested file when receiving a request. The Peers (i.e., Clients) here, act as both the client and the server. This server is different from the central index server which only indexes the files. But, the server functionality of the peer can be used to download the files from its directory. The peer acts a client to download the files from other peers into its directory.

The peers provide the following interface to the users:

1. Register – registers the file into the server

2. Search – searches the server for a file and returns the list of Clients

3. Download – downloads the file from another Client

TRADEOFFS:

- Instead of using the Array List for indexing, we can make using of the Data Structures. Even though Array List works fine in our case, but in case of randomly searching the files, hashing techniques serves well.

POSSIBLE IMPROVEMENTS:

- ➔ We can improve the performance using the Data Structures
- ➔ Could develop an User Interface.
- ➔ Port numbers can be eliminated.