Napster-Style Peer-to-Peer (P2P) File Sharing System

Vaishnavi Manjunath

INTRODUCTION:

Peer-to-Peer(P2P) Technologies are used for sharing the data between the servers and the peers. One of the most implemented technology these days is the Napster-Style Peer-to-Peer File Sharing System. The concept is file distributed throughout the nodes.

The older versions of the systems used to have a central server which stores the files in its directory that are received from the peers. All transfers would happen only between the central server and the peers.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 REQUIREMENTS:

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

DESIGN:

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

Major Components of the Project:

- Server and
- Peer

Server (Central Index Server):

This server indexes the content of all the peers that register with it. It also provides a search facility to peers.

Server Functionalities:

- Registry and
- Search

Peer:

As a peer, 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 peer then connects to this peer and downloads the file.

Major function of the peer:

- Download
- Multiple request search

As a server, the peer waits for requests from other peers and sends the requested file when receiving a request. The peers here, act as both the peer 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 downloads 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 peers
3. Download – downloads the file from another peer

TRADE OFFS:

Instead of using the Array List for indexing, we can make use 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 a User Interface.
- Port numbers can be eliminated.

EXECUTION OF THE PROJECT:

Below is the screenshot of the available java files present in the directory along with the Makefile

```
Last login: Wed Sep 25 15:59:06 2019 from 104.194.108.96
cc@group01-indexing-server:~$ cd /home/cc/Napster-style-peer-to-peer-P2P-file-sharing-system-master
cc@group01-indexing-server:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ ls
Design.pdf              GroupOneServer.class                README.md
GroupOneClient.class    GroupOneServer.java                 Report.pdf
GroupOneClient.java     GroupOneServerDownload.class        ServerDownloadThread.class
GroupOneFileInfo.class  GroupOneServerDownload.java         ServerTestClass.class
GroupOneFileInfo.java   GroupOneServerDownloadThread.class  Test.docx
GroupOneMain.class      GroupOneServerTestClass.class       Test.pdf
GroupOneMain.java       Makefile                            peer_to_peer_Files
cc@group01-indexing-server:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$
```

Server:

We Start our execution by executing the Server –

```
Last login: Wed Sep 25 15:59:28 2019 from 104.194.108.96
cc@group01-indexing-server:~$ cd /home/cc/Napster-style-peer-to-peer-P2P-file-sharing-system-master
cc@group01-indexing-server:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ javac *.java
cc@group01-indexing-server:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ java GroupOneMain
Welcome to Napster Style Peer to Peer File Sharing System :
Please enter your Choice ::
1. To run the Server
2. To run the Peer
1
Server started!!

Waiting for the Peer to be connected ..
```

Now Server will open its Socket and waits for the Peers to get connected

Peer:

We will register 4 Peers (Client) on to the Server as per the project requirement

For Peer1 -

```
cc@group01-peer1:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ java GroupOneMain
Welcome to Napster Style Peer to Peer File Sharing System :
Please enter your Choice ::
1. To run the Server
2. To run the Peer
2
Welcome to the Peer ::

Enter the directory that contain the files -->
```

For Peer2 -

```
cc@group01-peer2:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ javac *.java
cc@group01-peer2:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ java GroupOneMain
Welcome to Napster Style Peer to Peer File Sharing System :
Please enter your Choice ::
1. To run the Server
2. To run the Peer
2
Welcome to the Peer ::

Enter the directory that contain the files -->
```

For Peer3 -

```
cc@group01-peer3:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ java GroupOneMain
Welcome to Napster Style Peer to Peer File Sharing System :
Please enter your Choice ::
1. To run the Server
2. To run the Peer
2
Welcome to the Peer ::

Enter the directory that contain the files -->
```

For Peer4 –

```
cc@group01-peer4:~/Napster-style-peer-to-peer-P2P-file-sharing-system-master$ java GroupOneMain
Welcome to Napster Style Peer to Peer File Sharing System :
Please enter your Choice ::
1. To run the Server
2. To run the Peer
2
Welcome to the Peer ::

Enter the directory that contain the files -->
```

Now, we will enter the directory of the files for each Peer along with the port number and the peerid

For Peer1 –

```
Enter the directory that contain the files -->
C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-system-master\peer_to_peer_Files\Peer1
Enter the port number on which the peer should act as server ::
12341
Connection has been established with the Server
Enter the peerid for this directory ::
1
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
```

For Peer2 –

```
Enter the directory that contain the files -->
C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-system-master\peer_to_peer_Files\Peer2
Enter the port number on which the peer should act as server ::
12321
Connection has been established with the Server
Enter the peerid for this directory ::
2
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
```

For Peer3 –

```
Enter the directory that contain the files -->
C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-system-master\peer_to_peer_Files\Peer3
Enter the port number on which the peer should act as server ::
32121
Connection has been established with the Server
Enter the peerid for this directory ::
3
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
```

For Peer4 –

```
Enter the directory that contain the files -->
C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-system-master\peer_to_peer_Files\Peer4
Enter the port number on which the peer should act as server ::
22113
Connection has been established with the Server
Enter the peerid for this directory ::
4
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
```

Once all the Peers have registered on to the server, the total available filenames at each of the Peers will be registered in the Server –

```
Waiting for the Peer to be connected ..
All the available files from the given directory have been recieved to the Server!
Total number of files available in the Server that are received from all the connected peers: 10
All the available files from the given directory have been recieved to the Server!
Total number of files available in the Server that are received from all the connected peers: 20
All the available files from the given directory have been recieved to the Server!
Total number of files available in the Server that are received from all the connected peers: 30
All the available files from the given directory have been recieved to the Server!
Total number of files available in the Server that are received from all the connected peers: 40
```

1. **For Searching the Files and Downloading it to the desired folder:**

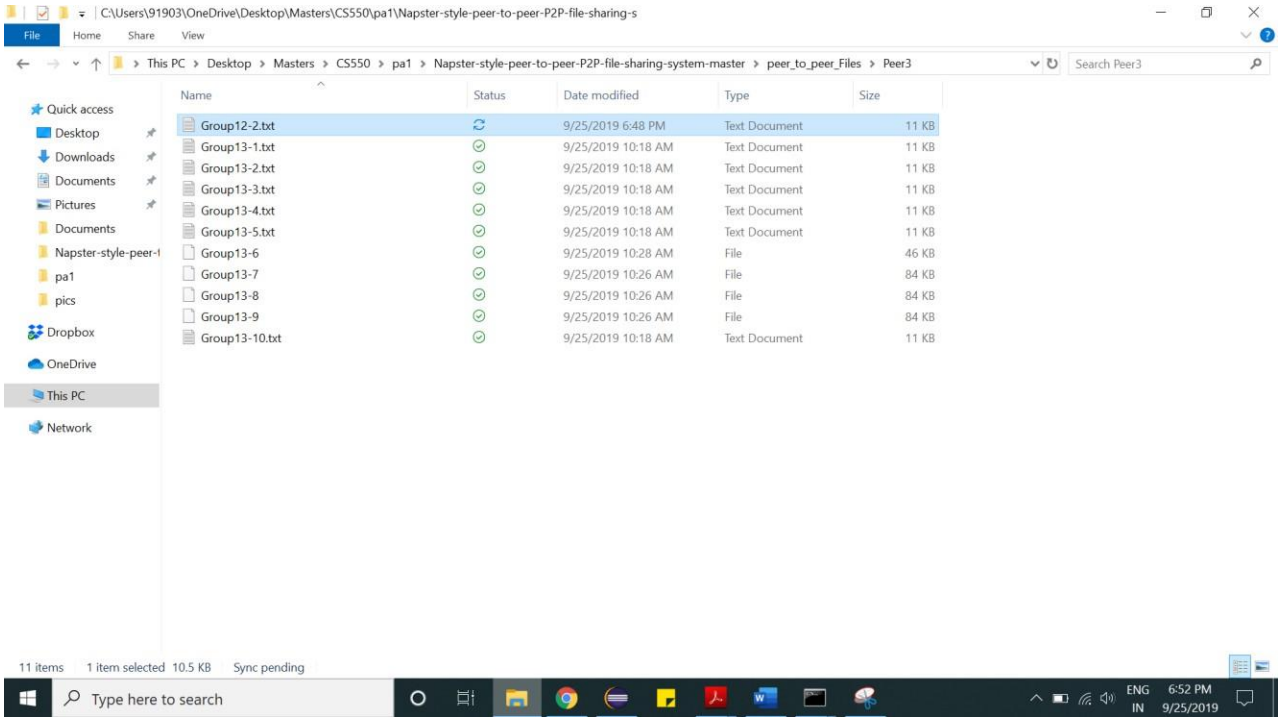
Go to the desired Peer (Say suppose, Peer 3) to where you want to download the file to.

Give the name of the desired file that you want to download from the list of the files available in the Server along with corresponding port number and the peerid

First file will be searched and then for downloading a message will be popped if given Yes file will be downloaded in the respective file folder and it displays File Name

```
Enter the peerid for this directory ::
3
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
1
Enter the desired file name that you want to search from the list of the files available in the Server ::
Group12-2.txt
Waiting for the reply from Server...!!
The file is stored at peer id 2 on port 12121
Do you want to download the file you searched? Please reply Yes or No:
Yes
Enter the respective port number of the above peer id :
12121
Enter the desired peer id from which you want to download the file from :
2
Requested file - Group12-2.txt, has been downloaded to your desired directory C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-
system-master\peer_to_peer_Files\Peer3

Display file Group12-2.txt
```



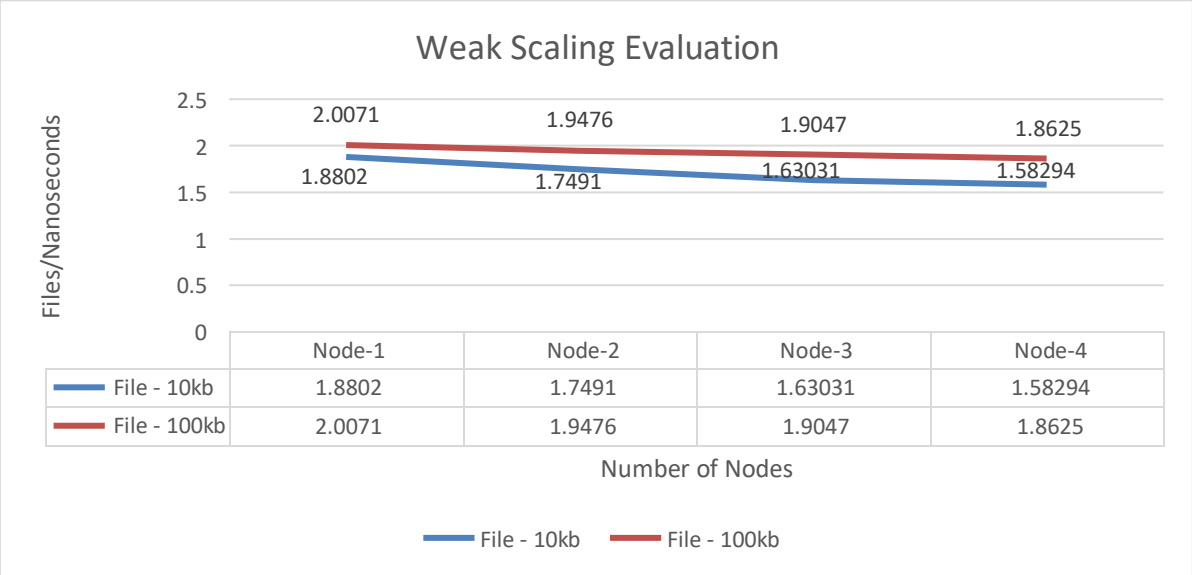
2. For multiple sequential requests:

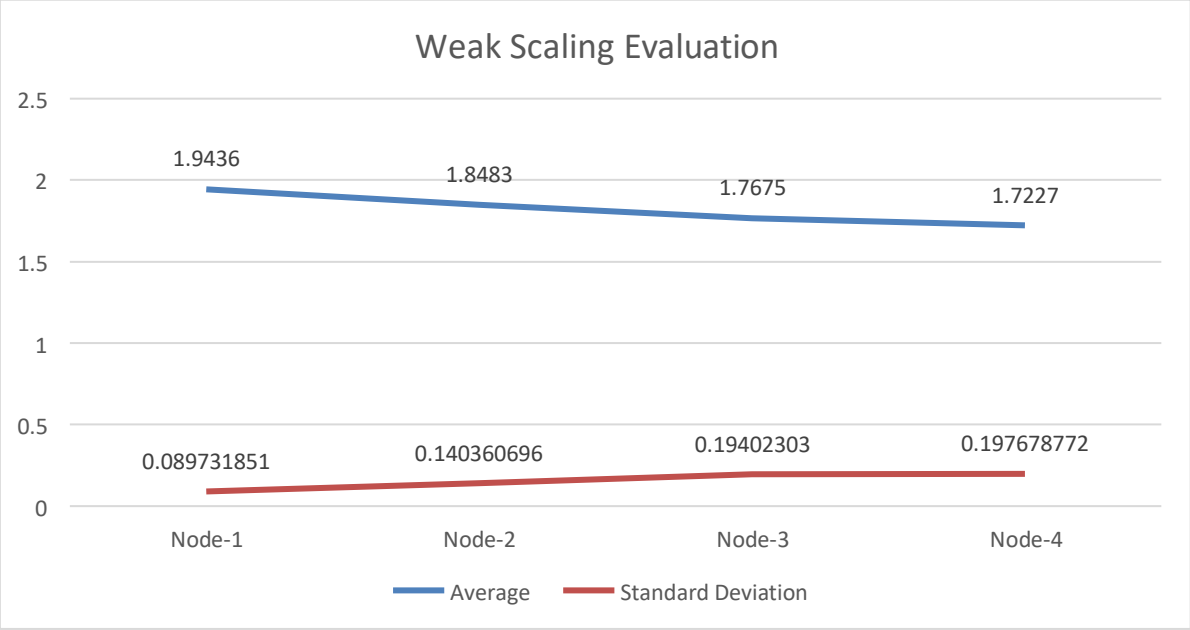
File from Peer-3 was requested to be searched for 50 times

```
Enter the directory that contain the files -->
C:\Users\91903\OneDrive\Desktop\Masters\CS550\pa1\Napster-style-peer-to-peer-P2P-file-sharing-system-master\peer_to_peer_Files\Peer1
Enter the port number on which the peer should act as server ::
12341
Connection has been established with the Server
Enter the peerid for this directory ::
1
Welcome to search and download:
Options available:
1. Search/download file
2. Test the Average Response Time for a single client performing multiple sequential search Requests
2
Enter the desired file name that you want to search from the list of the files available in the Server ::
Group13-9
Waiting for the reply from Server...!!
Number of Sequential requests:
50
Average Response time for 50 sequential search requests: 12.0 nanoseconds.
```

PERFORMANCE EVALUATION:

Weak Scaling:





Strong Scaling: