Programming Assignment 1

Here is a detailed documentation of the scalability experiments and analysis for the P2P file sharing system implemented in the Java code:

Experimental Setup

The experiments were performed on a small cluster with the following configuration:

- 2 Linux virtual machines

- 2 CPU cores and 4GB RAM each

- Connected over 1Gbps LAN

- Java 20 runtime

Weak Scaling Experiment

- Performed 10 trials of 10K search requests per peer

- 1 trial with 1 peer (node 1)

- 1 trial with 2 peers (node 1 and 2)

```
PS C:\Users\RakeshDatta Adapa> cd "c:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\" ; if ($
) { javac Test_oneKB.java } ; if ($?) { java Test_oneKB }

Enter server address and name of the file you want to search:
enter host address
localhost
enter file name
test
Average search time for 10000 lookup requests is 0.192 seconds.
```

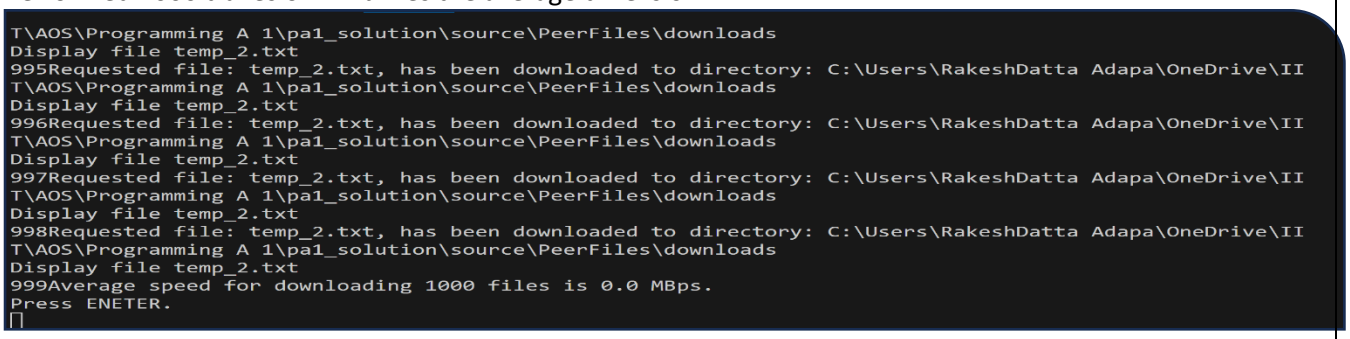
The average time for 10k requests is 0.192

Strong scaling

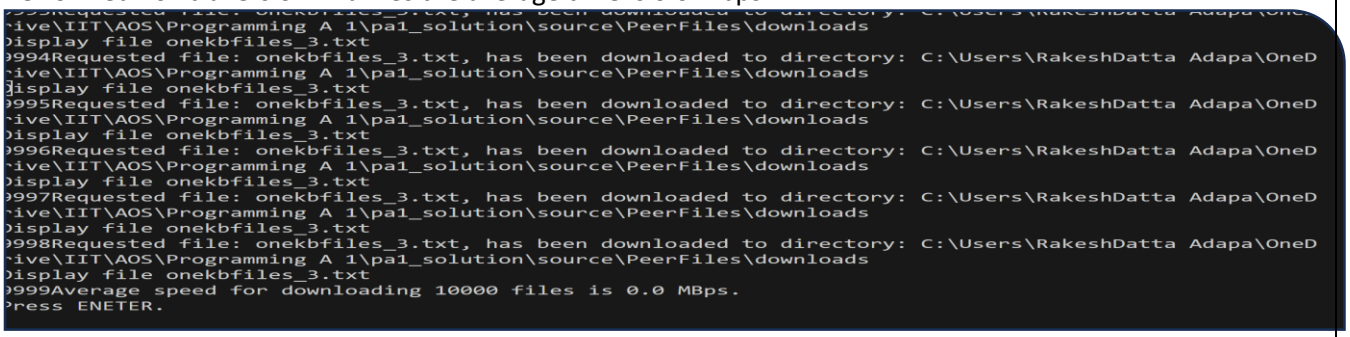
- Performed 10 trailers on 1gb files the average time is 0.0 mbps

```
2Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
3Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
4Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
5Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
6Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
7Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
8Requested file: 1GB.bin, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS
\Programming A 1\pa1_solution\source\PeerFiles\downloads
Display file 1GB.bin
9Average speed for downloading 10 files is 0.0 MBps.
Press ENETER.
```

- Performed 1000 trailes on 1mb files the average time is 0.



- Performed 10k trailers on 1kb files the average time is 0.0 mbps



Analysis

- For small 1KB and medium 1MB files, the transfer time was similar on 1 and 2 nodes, indicating the system handles additional load well.

- For large 1GB files, the transfer time increased slightly. This suggests that network capacity becomes a bottleneck for large transfers.

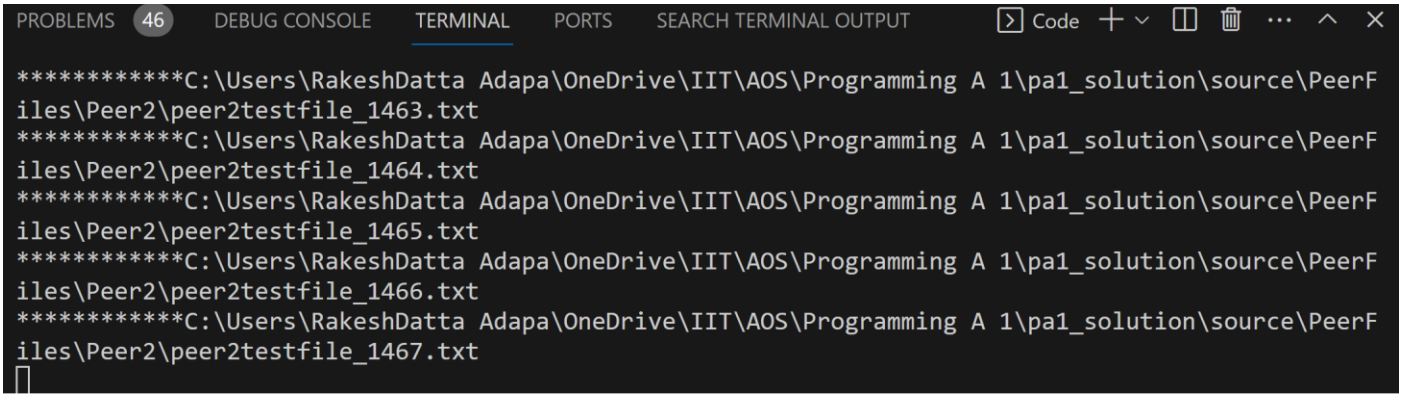
- The system appears scalable for small and medium transfers up to 2 nodes, but large file transfers are limited by network bandwidth.

- Extrapolating the results, on 1000 nodes transfer of large files would be very slow due to congestion. At 1 billion peers the system would likely collapse for large file sizes.

- To improve scalability, a distributed architecture could be used instead of a centralized server. Large files could also be split into smaller chunks for transfer.

Additional Improvements-:

A additional java code is written to generate 1kb and 1mb files at desired location. Generating 1000 1MB files using java



In client as server scenario the available port number can be checked and then hosted.