

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\pal_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.
```

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\pal_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\pal_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\pal_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\pal_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\pal_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\pal_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\pal_solution\source\PeerFiles\downloads
Display file 1GB.bin
9Average speed for downloading 10 files is 0.0 MBps.
Press ENTER.
```

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

```
T\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file temp_2.txt
995Requested file: temp_2.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\II
T\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file temp_2.txt
996Requested file: temp_2.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\II
T\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file temp_2.txt
997Requested file: temp_2.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\II
T\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file temp_2.txt
998Requested file: temp_2.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneDrive\II
T\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file temp_2.txt
999Average speed for downloading 1000 files is 0.0 MBps.
Press ENTER.
□
```

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

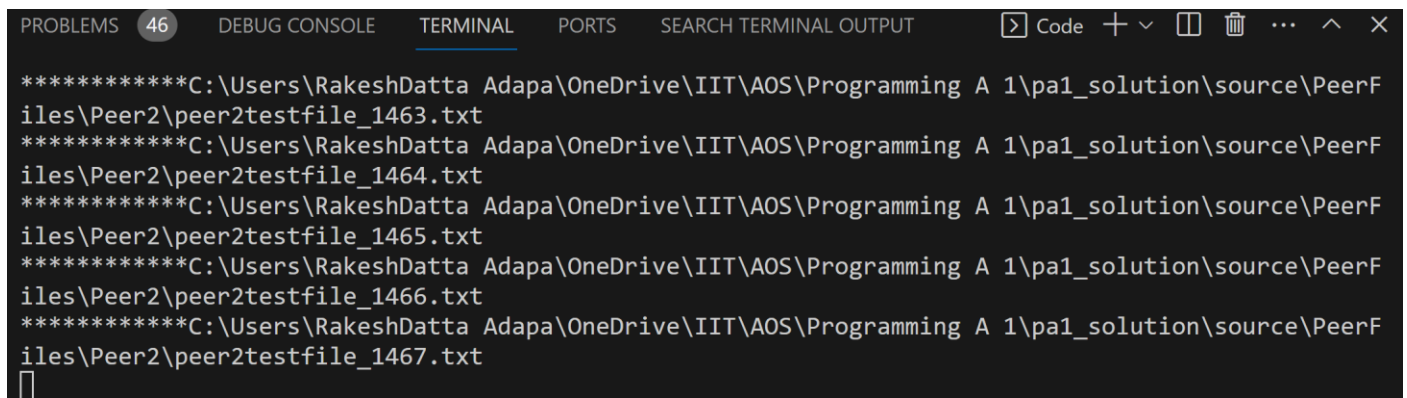
```
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9994Requested file: onekbfiles_3.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneD
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9995Requested file: onekbfiles_3.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneD
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9996Requested file: onekbfiles_3.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneD
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9997Requested file: onekbfiles_3.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneD
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9998Requested file: onekbfiles_3.txt, has been downloaded to directory: C:\Users\RakeshDatta Adapa\OneD
ive\IIT\AOS\Programming A 1\pal_solution\source\PeerFiles\downloads
Display file onekbfiles_3.txt
9999Average speed for downloading 10000 files is 0.0 MBps.
Press ENTER.
```

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

A screenshot of a Visual Studio Code terminal window. The terminal has a dark background with white text. At the top, there is a toolbar with tabs for 'PROBLEMS' (with a count of 46), 'DEBUG CONSOLE', 'TERMINAL' (which is active and underlined), 'PORTS', and 'SEARCH TERMINAL OUTPUT'. To the right of the tabs are icons for 'Code', a plus sign, a minus sign, a square, a trash can, three dots, an up arrow, and a close 'X' button. The terminal content consists of five lines of text, each starting with '*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_' followed by a file number from 1463 to 1467, and ending with '.txt'. The cursor is positioned at the end of the fifth line.

```
*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_1463.txt
*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_1464.txt
*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_1465.txt
*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_1466.txt
*****C:\Users\RakeshDatta Adapa\OneDrive\IIT\AOS\Programming A 1\pa1_solution\source\PeerFiles\Peer2\peer2testfile_1467.txt
█
```

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