Programming Assignment #2

A Simple Napster Style Peer to Peer File Sharing System Evalution

Jinyang Li

A20317851

```
Experiment Results:

Single Client Operation
==================================
total time for 10k register is
37398ms
Average time for registry is 3.7 ms

total time for 10k search is
136311ms
Average time for search is 13.6 ms

total time for 10k obtain is
179211ms
Average time for obtain is 17.9 ms


Two Clients Concurrent Operation
==================================
total time for 10k register is
68141ms
Average time for registry is 6.8 ms

total time for 10k search is
226109ms
Average time for search is 22.6 ms

total time for 10k obtain is
443451ms
Average time for obtain is 44.3 ms
```

```
Four Clients Concurrent Operation
==================================
total time for 10k register is
173189ms
Average time for registry is 17.3 ms

total time for 10k search is
492239ms
Average time for search is 49.2 ms

total time for 10k obtain is
901321ms
Average time for obtain is 90.1 ms


Eight Clients Concurrent Operation
==================================
total time for 10k register is
442391ms
Average time for registry is 44.2 ms

total time for 10k search is
1314623ms
Average time for search is 131.5 ms

total time for 10k obtain is
2544323ms
Average time for obtain is 254.4 ms
```

Conclusion

We can see that when perform concurrently requesting, the average cost time increase significantly.. this is due to method sync lock, the methods can be called by only one thread at same time.

The performance is as my expected since the sync lock would result thread safe but simply cost more time, nearly equal to call theses methods one by one.

This is exclude from any internet delay problems since we are at local machine..
So my conclusion is my system is running nicely when clients are making request solely..
But when clients making request concurrently,, the performance would drop significant.

Also, checkAlive cost a lot of time if the checked peer is offline.