Design Document

Server Design

A node process can be split to server part and client part.

When node set up, process starts server part to listen particular port that will be used to communicate with other nodes, and client part to listen another port to get command from clients and transfer command to one node's server part. When server part receive message from one node client part, server handle hash table in term of receive message. After operation for hash table, server parts of this node sends result back to original node and close socket. The node, that communicate with client, send back result to client and close socket.

Hash table places in node's server part. The type of hash table is dictionary. Put operation is to update the dictionary. Get operation is to get value of the key. Del operation is to delete particular pair.

Client Design

A peer process can be split to server part and client part.

Command operation and communication for server will be handled in client part.

Other peers can connect with server part of peer to request download files.

Synopsis Design

0 [FILE ABSOLUTE PATH]

Register file to server.

Return register result

1 [KEYWORD]

Look up files that name is KEYWORD.

Return file URL

2 [FILE URL]

Download file according URL.

Return download result

3

Exit system.

Test_register 1

0K register operation

Return time.

Test_lookup

10K look up operation

Return time.

Test_obtain [URL]

10K obtain time, each file less 10KB

Return time.

Possible Improvement

Using asymmetric communication for each request.

Using more than one table in each node.