A Server-mediated Peer-to-peer System

Kwok, S. H.
California State University, Long Beach
Chan, K. Y.
and
Cheung, Y. M.
Hong Kong University of Science and Technology

A peer-to-peer (P2P) system is a popular means of file distribution. Existing P2P systems do not adequately address a variety of common problems including (1) limited search scope, (2) dynamic nature of peers and (3) lack of collaboration among peers. In this paper, we propose a server-mediated peer-to-peer system to address these problems and to improve the performance of existing decentralized P2P systems by incorporating a central server into the decentralized P2P system to facilitate collaboration among peers. Two main features of our proposed system are Assisted-search and Assisted-download. Experimental results show that the search coverage was increased by 289.91% by using Assisted-search, while the overall download speed was improved by 33% by using Assisted-download.

Categories and Subject Descriptors: C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Network topology*; C.2.2 [**Computer-Communication Networks**]: Network Protocols—*Routing protocols*; C.2.4 [**Computer-Communication Networks**]: Distributed Systems—*Distributed applications*

General Terms: Design, Experimentation

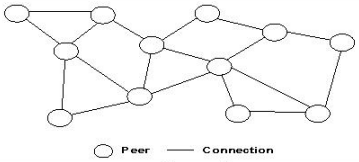
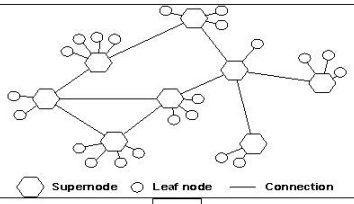
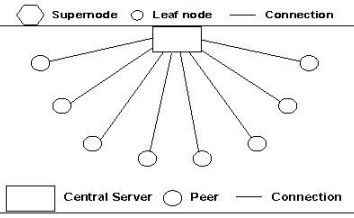
Additional Key Words and Phrases: Peer-to-peer, file distribution, server-mediated, decentralized, collaboration

1. INTRODUCTION

Peer-to-peer (P2P) computing can be best described as the direct exchange of data between two computers or peers in a common network. In a P2P network, all client computers in the network, known as nodes, are considered to be equal in their capacity for sharing resources with other network members. Unlike the traditional Client-Server system, a P2P system requires no central coordination of nodes. Gnutella [6] is an example of decentralized and unstructured P2P systems, in which peers join the network by broadcasting ping and pong messages arbitrarily, but such a flooding mechanism limits the scope of file searching and renders the networks

Authors' addresses: Kwok, S. H., Department of Information Systems, College of Business Administration ,California State University, Long Beach, 1250 Bellflower Boulevard, Long Beach, CA 90840-8506, USA; Chan, K. Y. AND Cheung, Y.M., Department of Information and Systems Management, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.

Table I. Summary of P2P systems

	Presence of File Directory Server	Topology
Decentralized (pure decentralized)	No	
Decentralized (supernode)	Yes (Dynamic Supernode)	
Centralized	Yes (Dedicated)	

less scalable. Advanced approaches to improve the P2P networks include implementation of hybrid architecture, in which some nodes are dedicated to serve other nodes, and employment of routing mechanisms which provide precise file lookup and placement [17, 18, 20]. However, these systems do not adequately address a variety of common problems including (1) limited search scope, (2) dynamic nature of peers and (3) lack of collaboration among peers.

In this article, we intend to address the aforementioned problems by proposing a server-mediated P2P system. Our proposed server-mediated P2P system aims at supplementing the decentralized and unstructured P2P systems to facilitate grouping of inter-trusted peers and sharing of their computing and network capabilities. We begin by briefly introducing existing P2P systems, and discuss some issues motivating the development of our server-mediated P2P system. Then we present our server-mediated P2P system prototype. We also present performance evaluation results.

2. REVIEWS OF P2P SYSTEMS

Existing P2P systems employ two common architectures, namely centralized and decentralized [12]. For decentralized systems, we further classify them as ones that employ the pure decentralized model and ones that employ the supernode model. Our study will focus on the decentralized systems which employ the "pure" decentralized model. Centralized P2P architecture will also be briefly introduced. A summary of the architectures and topologies of these systems is presented in Table 1.

2.1 Decentralized

These systems have neither a central server nor precise control over the network topology and file placement [12] . They employ the pure decentralized model and the supernode model.

2.1.1 *Pure decentralized model.* This model is featured by non-existence of a centralized directory. Nodes join the network by randomly connecting to the existing participants and all requests are broadcasted using the flooding algorithm. Each node, known as a servent, acts as a server to handle requests of other nodes, and at the same time acts as a client to receive services provided by other nodes. Gnutella [6] is an example of pure decentralized systems.

2.1.2 *Supernode model.* These systems have no fixed central directory server but supernodes, which are dynamically assigned based on the computer resources of peers. Supernodes are powerful peers which provide file indexing services for their connected peers known as leaf nodes, with less computer resources. Unlike the servers in the centralized systems, the supernodes keep a file index of their connected leaf nodes, reflecting only a partial view of the network. The file index records the information including the available files shared by leaf nodes and their corresponding locations. When the leaf nodes perform searching, they will first search for the files in the file index kept by the supernode. If results are found, they will be returned to the leaf nodes at once, so the search performance is better than that of the flooding algorithm. An example using this model is FastTrack [5].

2.2 Centralized

For the centralized P2P system, a central server is present to provide the directory service, yet the server does not involve in file transfer. Peers download files from other peers after they receive the location information of files from the central server. It is a system with centralized search architecture but decentralized file sharing. Napster [16] and eDonkey [4] are examples of centralized P2P systems.

3. THE GNUTELLA NETWORK

3.1 Background

The Gnutella network is chosen to be the platform for our proposed server-mediated system since it is a decentralized and unstructured P2P system that employs the pure decentralized model. In this model, all servents are equal in terms of functionality. They offer client-side functions such as accepting queries from users and returning search results, while they also perform server-side roles such as matching incoming queries against their local resources. Free-riding [1, 9, 11], which refers to contributing either no file or undesirable files to the P2P network, is a major problem in the Gnutella network. Examples of public servent programs include Bearshare [2] and Limewire [10].

3.2 Targeted problems

Our proposed server-mediated system is intended to address these problems: (1) limited search scope, (2) dynamic nature of peers, and (3) lack of collaboration among peers.

3.2.1 *Limited search scope.* In the Gnutella network, each peer directly connects to one or more neighboring peers. To look for a file, a peer generates a search query and broadcasts the query to all neighbors. Its neighbors check their shared files in order to locate matched files, and at the same time, broadcast this incoming query to their connecting neighbors except the source of query. The search query continues to travel to different nodes until the time-to-live (TTL) value in the search query reaches zero.

The Gnutella Protocol limits the coverage of a query to 10000 peers regardless of the network size [7]. Because some packets are dropped by peers which have low bandwidth, the effective reach of a Gnutella query is reduced to between 5,000 and 8,000 peers [8]. Increasing the value of TTL could be a possible solution but this approach will increase the amount of network traffic substantially. Prior researchers have devoted their efforts to improving querying algorithms. Menasce [14] proposed a probabilistic search protocol to reduce the traffic spent on locating resources. Lv et al. [12] suggested that random probing is a reasonable model for search performance in such decentralized unstructured P2P systems. These approaches, however, trade search coverage for preservation of network bandwidth.

3.2.2 *Dynamic nature of peers.* The nodes turnover rate is very high in the Gnutella network. It is estimated that half of the nodes participating in the network will be replaced by new nodes within one hour [19], resulting degraded performance in searching and downloading. Several researchers [17, 18] have proposed a highly structured network design in which the network can provide the exact location of a specific file. However, it is difficult to maintain the highly structured routing due to the dynamic nature of the network. Therefore, it is desirable to cluster stable peers and facilitate message exchange among them, in order to provide a guaranteed level of performance in such a dynamic environment. In addition, the dynamic nature of peers renders it difficult to determine peers reputation, since no central authority is present to store users records. Damiani et al. [3] proposed a distributed polling algorithm to access peers reputation. However, the polling results may be deviated because the peers involved in each poll may be different and these peers may possess inconsistent peers information.

3.2.3 *Lack of collaboration among peers.* P2P systems should facilitate efficient use of peer resources, such as processing power, storage and bandwidth. However, research shown that up to 96% of local peer node resources are idle [15]. Now, peers perform searching and downloading on an individual basis without collaborating with other peers. The efficiency of P2P systems can be enhanced if peer resources are aggregated and utilized more thoroughly.

4. THE PROPOSED SERVER-MEDIATED P2P SYSTEM

In general, existing P2P systems lack a mechanism for a group of inter-trust peers, whose reputations are known to other peers, to aggregate their resources to satisfy their mutual needs. The proposed server-mediated P2P system aims to address the targeted problems in the Gnutella networks. It (1) broadens the search coverage by implementing assisted-search (refer to 3.2.1), (2) clusters trustful peers through peer discrimination done by the server to enhance stability (refer to 3.2.2) and (3)

helps peers locate other assisting peers to facilitate collaboration (refer to 3.2.3). In short, our proposed system aims to improve the performance of decentralized P2P systems through collaboration among peers.

4.1 Overview

A servent (a Gnutella peer) named JWire is built to collaborate with our proposed system. JWire, developed based on the JAVA API of JTella [13], provides not only the ordinary functions supported by other public servents, but also the capability of connecting to our proposed central server to perform assisted-search and assisted-download.

A standalone server is to enable inter-trust peers to perform file searching and downloading collaboratively. The server is to coordinate the collaborative activities and to act as an extra point of connection for the P2P client, JWire. The JWire peers can establish a direct connection to the server such that the connection is independent of the pure P2P network. After making connection to the server, the JWire peer can send either assisted-search or assisted-download requests to the server directly.

A user database is to maintain user records as the third-tier. Only the server has privileges to directly access the database and update the user records. Apart from the permanent records at the database, a list of connected peers is maintained in the server in real time. This available-user list is to record the updated status of connected peers, specifically stating whether an individual peer is available to be an assisting peer for assisted-search or assisted-download. It should be noted that the server will not participate in the actual file transmission among JWire peers. The network traffic between the server and JWire peers is limited to the messages for collaborative activities. The architecture of our server-mediated P2P system is illustrated in Figure 1.

4.2 Interactions of the system

There are three main types of interactions between JWire peers and the server, namely log-in/registration, assisted-search and assisted-download. All interactions are independent of the public Gnutella network.

4.2.1 *Log-in/Registration.* To facilitate peer differentiation, each JWire peer is required to log in to the server before using the services offered by the server. For the concern of privacy, a user can check its own records but not others records. Each server can customize its rules so that it can accept or reject connections from existing users or registration from a new peer. The presence of a central server helps differentiate peers which have various stability and reputation by using its centralized records. The stability of a peer is a function of current network bandwidth and response time, while the reputation of a peer is based on its historical records of network bandwidth and the average time it connects to the centralized server. By selectively serving peers which have high stability and good reputation, the proposed system can cluster qualified peers and improve stability of the P2P community.

4.2.2 *Assisted-Search.* The aim of assisted-search is to increase the search coverage, which is currently limited by the Gnutella Protocol. The main idea behind

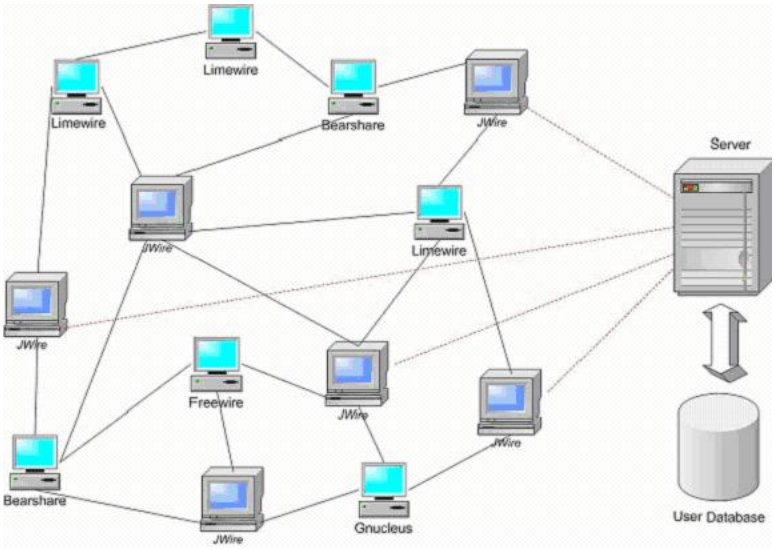


Fig. 1. Interactions of JWire in P2P networks.

assisted-search is to broadcast queries with the same search criteria to different segments of P2P networks to increase the search coverage. Figure 2 illustrates a simple case of assisted-search. The JWire peer first sends a request for assisted-search to the Server. Then the server identifies available peers and sends the search criteria to the assisting peers. Each assisting peer broadcasts the search query with the provided search criteria to its connected network segment and finally sends the search results to the original JWire peer. In this case, the search coverage is improved by 4 more peers.

4.2.3 *Assisted-Download.* Similar to assisted-search, assisted-download can improve performance of file downloading with assistance from assisting peers and coordination of the server. The main idea of assisted-download is to provide a JWire peer with additional and dedicated sources of different file segments (portions of a file) As a result, the JWire peer who is in need of the desired files is able to download different file segments from more available sources. By collaborating with other assisting peers, the JWire peer is able to achieve a higher file transmission rate.

Every time the server accepts a request for assisted-search from a JWire peer, it picks a certain number of available peers, if any, from the available-user list. A simple example is depicted in Figure 3. The JWire peer first sends a request for assisted-download to the server. Then the server identifies available peers and sends the information of the wanted file to the assisting peers. Each assisting peer downloads the assigned file segment from the file source and acts as an alternative source from which the JWire peer downloads. Finally, the JWire peer combines those segments downloaded from the original source and those from the assisting peers to form a complete file.

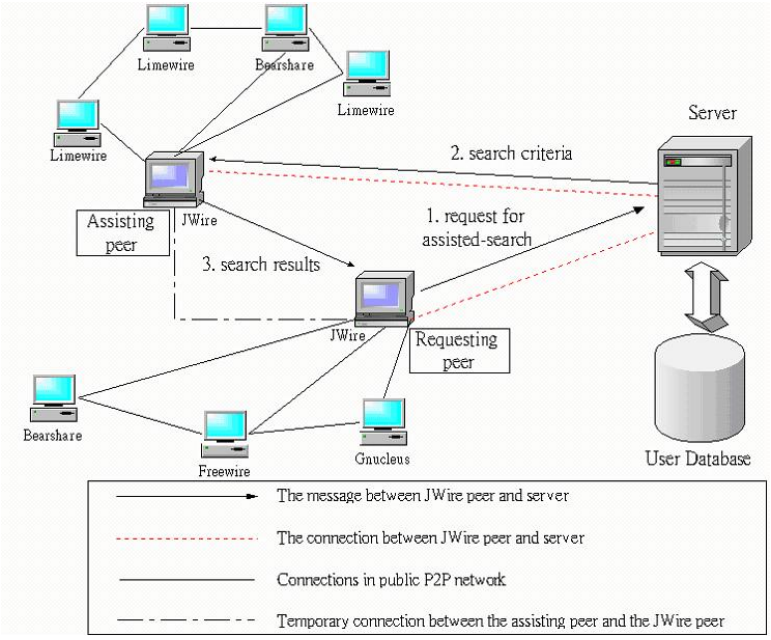


Fig. 2. Illustration of assisted-search.

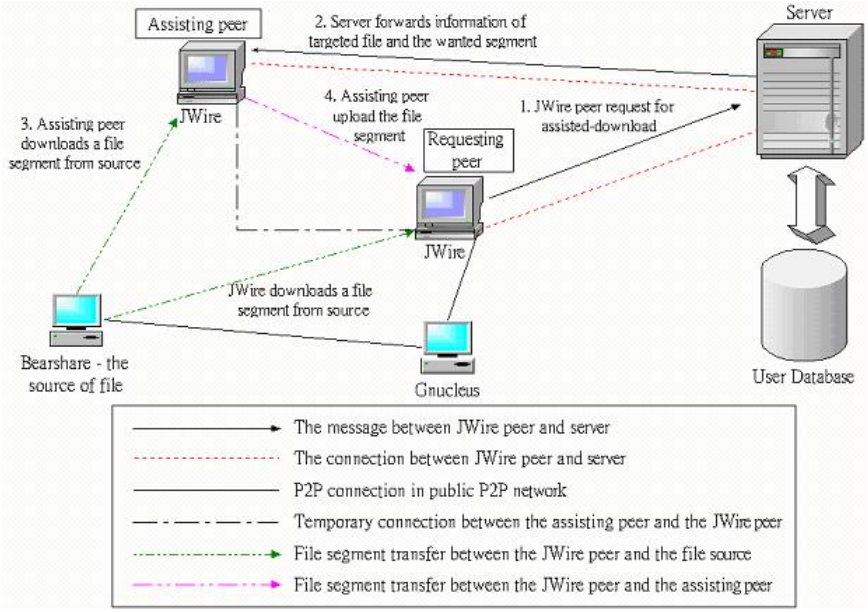


Fig. 3. Illustration of assisted-download.

5. PERFORMANCE EVALUATION

5.1 Assisted-Search

An experiment was conducted to compare the effectiveness of standalone searching and assisted-search. In the experiment, a JWire peer and 2 assisting peers connected to different nodes in the network. Although the network segments of the three involved peers might overlap, each of them must connect to network subsegments to which the other two did not connect. Our focus in this experiment was whether search coverage, measured by the coverage of distinct peers with distinct IP addresses, was increased by performing assisted-search, compared with the search performed solely by a JWire peer. We selected IP address as the parameter to measure the improvement in search coverage rather than the intuitive parameter, or the number of matched files, because the number of matched files returned depends heavily on techniques for matching search criteria with file indices inside a peer. Two peers with identical shared files but distinct file indices matching could generate different query hits in response to the same search query.

We measured the total number of distinct hosts with distinct IP addresses tagged on query results that were received in a 10-minute period, first by searching performed by a JWire peer solely and second by a JWire peer with the assistance of 2 assisting peers. The term mp3 was chosen as the search criterion in this experiment because it is a popular query term in P2P networks, by which a query hit from every accessible peer is guaranteed. The searching effectiveness was evaluated by the mean number of distinct hosts with distinct IP address counted in a series of 10-minute intervals.

We completed 10 trials in total and the results are presented in Table 2. The mean coverage of distinct hosts by means of assisted-search (involving 2 assisting peers) was 395.5 while the mean coverage by a sole peers search was 92.20. On average, 289.91% gain in the search coverage was resulted from making use of assisted-search.

5.2 Assisted-Download

We conducted another experiment to measure the effectiveness of file downloading between traditional download with a single JWire peer and assisted-download with the assistance of two assisting peers.

The experiment was conducted by setting up two PCs at the Hong Kong University of Science and Technology campus in which both PCs belonged to the same network and subnet, labeled network A, and a PC using another ISP, i-Cable, belonging to another network labeled network B. There were two tests in the experiment where three PCs played different roles as a peer having the source file (source peer), a peer requesting for file (JWire peer) and a peer participating in the assisted-download (assisting peer) respectively.

We measured the improvement in file downloading by comparing the (average) speed between download solely by a JWire peer from a source of file and download collaboratively by a JWire peer with the help of an assisting-peer. In the collaborative download, the JWire peer and assisting peer each downloaded an equal portion of the source file. The total time for file download was recorded and the (average) speed is calculated.

Table II. Summary of experiments on effectiveness of assisted-search and assisted-download

Assisted-Download			**Assisted-Search**	
	Test 1	Test 2		
Mean gain in speed (of 100 trials)	38.11%	34.84%	Mean gain in search coverage (of 10 trials)	289.91%

In Test 1, the PC in network B was chosen as the JWire peer while the two PCs at network A were set up as the source peer and assisting peer respectively. The objective of this test was to measure the increment in downloading speed under the situation that the assisting peer was within the same network as the source peer. The gain in speed was calculated by subtracting the speed in downloading solely by a JWire from the speed in downloading by assisted-download. After repeating the trials for 100 times, the mean gain in speed by making use of assisted-download was found to be 38.11%.

Test 2 probably illustrates the most common circumstance of using our server-mediated system. The JWire and assisting peers belonged to the same network, network A, and the source peer resided in another network, network B. Such set up closely resembles the circumstances that a group of inter-trust peers within the same department of an organization perform download collaboratively from a remote file source. In Test 2, 100 times of trials were repeated and the mean gain in speed was found to be 34.84%. The summary of experiments on effectiveness of assisted-search and assisted-download was illustrated in Table II.

6. COMPARISON OF THE SERVER-MEDIATED SYSTEM WITH CENTRALIZED P2P SYSTEMS

Our proposed server-mediated system is significantly different from that of the traditional centralized system. The role of server in our proposed system is to help peers locate other assisting peers in order to facilitate the coordination among peers, whereas the role of server in the centralized system is to provide file directory service. Our proposed system does not rely on the central server to perform core P2P tasks, such as querying, and the primary role of our server is to provide extra functionality on top on the existing decentralized systems, so peers can still perform ordinary P2P tasks even when the server is down. As a result, the proposed server-mediated system is able to avoid the problem of central point of failure and can be fully compatible with any decentralized P2P systems to improve the performance through peer coordination. Moreover, since the server in our proposed system does not perform file indexing and query processing, the server requires less computational power than the server in the centralized system does.

7. CONCLUSIONS AND FUTURE WORKS

Existing P2P systems lack a mechanism for a group of voluntary and inter-trust peers to utilize available network and computing resources to satisfy the needs of group members in a collaborative sense. The proposed server-mediated P2P system is intended to address the following problems in the Gnutella network: (1) limited search scope, (2) dynamic nature of peers, and (3) lack of collaboration among peers. Our system supplements existing P2P systems to facilitate collaboration

among peers. We illustrated the proposed system on the Gnutella network and introduced the three main components of our system, namely log-in/registration, assisted-search and assisted-download. We conducted experiments to evaluate the effectiveness of assisted-search and assisted-download. An average gain in search coverage by 289.91% was achieved by using assisted-search. By employing assisted-download, the overall speed was improved by 33% in general. In short, our proposed system has successfully improved the performance of P2P systems through collaboration among peers.

Finally, we are going to extend our work on the failure and recovery issues of server and peers in the collaborative activities. Security issue such as malicious peers is also an area we would like to focus on in the future. Above all, we will investigate the total computing and networking resources that are required for the optimal operation of our system.

REFERENCES

1. Adar, E. and Huberman, B.A. Free riding on Gnutella. First Monday, 5 (10).
2. Bearshare. Bearshare, 2003.
3. Damiani, E., De Capitani Di Vimercati, S., Paraboschi, S. and Samarati, P. Managing and sharing servants' reputations in P2P systems. IEEE Transactions on Knowledge & Data Engineering, 15 (4). 840-854.
4. eDonkey. eDonkey2000, 2003.
5. FastTrack. FastTrack, 2003.
6. Gnutella. The Gnutella Protocol Specification v0.4, 2003.
7. Kirk, P. Gnutella Protocol Development - Protocol Specification, 2003.
8. Krishnan, R., Smith, M.D. and Telang, R. The Economics of Peer-To-Peer Networks. The Journal of Information Technology Theory and Application (JITTA), 5 (3). 31-44.
9. Kwok, S.H., Lang, K.R. and Tam, K.Y. Peer-to-peer technology business and service models: risks and opportunities. Electronic Markets, 12 (3). 175-183.
10. Limewire. Limewire, 2003.
11. Lui, S.M., Lang, K.R. and Kwok, S.H., Participation incentive mechanisms in peer-to-peer subscription systems. In Proceedings of the 35th Annual Hawaii International Conference on System Sciences., (Hawaii, 2002), IEEE Comput. Soc., 3925-3931.
12. Lv, Q., Cao, P., Cohen, E., Li, K. and Shenker, S., Search and replication in unstructured peer-to-peer networks. In Conference Proceedings of the 2002 International Conference on SUPERCOMPUTING. ACM. 2002, (2002), 84-95.
13. McCrary, K. and Waters, B. JTella.
14. Menasce, D.A. Scalable P2P search. IEEE Internet Computing, 7 (2). 83-87.
15. Mischke, J. and Stiller, B., Peer-to-peer overlay network management through AGILE. In Integrated Network Management VIII. Managing It All. IFIP/IEEE Eighth International Symposium on Integrated Network Management (IM 2003) (IEEE Cat. No.03EX700). (2003), Kluwer Academic Publishers, 337-350.
16. Napster. Napster, 2003.
17. Ratnasamy, S., Francis, P., Handley, M., Karp, R. and Shenker, S. A scalable content-addressable network. ACM. Computer Communication Review, 31 (4). 161-172.
18. Rowstron, A. and Druschel, P. Storage management and caching in PAST, a large-scale, persistent peer-to-peer storage utility. ACM. Operating Systems Review (Acm), 35 (5). 188-201.
19. Saroiu, S., Gummadi, P.K. and Gribble, S.D., Measurement study of peer-to-peer file sharing systems. In SPIE - the International Society for Optical Engineering, (2001), 156-170.
20. Stoica, I., Morris, R., Karger, D., Frans Kaashoek, M. and Balakrishnan, H. Chord: a scalable peer-to-peer lookup service for Internet applications. ACM. Computer Communication Review, 31 (4). 149-160.