Comparative Analysis of Unstructured P2P File Sharing Networks

Mehr-un-Nisa, Fasiha Ashraf, Ateeqa Naseer, Shaukat Iqbal
University of Management and Technology, Lahore
mehr.nisa@umt.edu.pk; fasiha.ashraf@umt.edu.pk; ateeqa.naseer@umt.edu.pk; shaukat.iqbal@umt.edu.pk

ABSTRACT

Numerous P2P networks are available covering different areas of P2P technologies. Since P2P network overlays are good for creating large scale content distribution and data sharing applications, an ample amount of effort has been made to improve the performance of these networks. Moreover, rapid development of P2P technologies has left older technologies far behind. The goal of this paper is to study and present a comprehensive overview of some of these networks. Furthermore, a comparison of Unstructured P2P file sharing wired networks with their distinct characteristics has been presented. In this paper, various features of P2P networks have been studied that results in understanding the behavior of these protocols under different circumstances. A critical analysis of the state of art P2P networks has been made taking care of various significant features for both wired. The analysis presents the effectiveness of different systems in different circumstances. This research aims to provide a comprehensive comparative analysis of P2P file sharing protocols of wired networks.

CCS Concepts

Networks → Network performance analysis

Keywords

Peer-to-Peer Unstructured Protocols, BitTorrent, Freenet.

1. INTRODUCTION

PEER-TO-PEER is a distributed technology that divides the work load between peers. All peers are given same privilege working both as "clients" and "servers" in the overlaying network. P2P file sharing networks helps users to share different files like movies, books, pirated software or games. P2P software finds the location of desired files in the network and start downloading it. It is made available to the user through internet.

File sharing networks are divided into two categories: Centralized (client-server) and Decentralized (P2P). A hierarchical view of file sharing networks is shown in Figure 1: A hierarchical view of File Sharing Networks.. Decentralized networks are further classified into Structured and Unstructured. Structured Networks are those networks which strictly follows some topology, where peers are arranged in some specific order, which help in efficient searching.

ICISDM 2019, April 6–8, 2019, Houston, TX, USA

ACM ISBN 978-1-4503-6635-9/19/04…$15.00
https://doi.org/10.1145/3325917.3325952

On the other hand, Unstructured Networks organize peers randomly to form a network. Queries are passed on maximum possible peers to check the content stored on peers. The results against queries are more accurate in case of structured networks. Because in Unstructured Networks, there is no correlation between content and peer storing it.

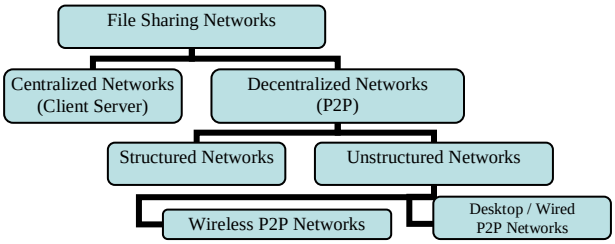

Figure 1: A hierarchical view of File Sharing Networks.

In a Structured P2P Network, a peer can locate rare content efficiently (because keys and nodes are correlated), while popular content increases overhead in structured systems. In Unstructured P2P networks, queries for rare contents are not guaranteed (due to uncertainty of the node having data), but popular content is mostly available in nearby network. One problem faced by Unstructured Networks is that peers become readily overloaded with queries when a popular content is available. Three Unstructured P2P file sharing protocols has been selected for detailed study: Fast Track, BitTorrent and Freenet. They are the most popular networks possessing some similarities and differences among them. An extensive study has been conducted to gather the features of all these networks in this paper [2] [1]. The comparison of the unstructured wired networks has been made while considering the following points:

- *Decentralization/Architecture* — examines the nature of distribution used in the overlay system, and describes architecture with respect to its operation.
- *Lookup Method* — the method and the required system parameters adopted by the overlay system for query processing.
- *Routing Performance* — the behavior of an overlay network when a query is initiated.
- *Download/Upload Performance* — capacity of an overlay network to use its bandwidth for downloading or uploading data.
- *Content Lifetime* — how long the peers can keep the content in their database
- *Flash Crowd performance* — examines how protocols can handle the crowd on the network for popular content.

- *Scalability* — examines the ability of a network to handle growing amount of work.
- *Availability* — examines the ability of a system to be operational for a long time.
- *Security/Pollution* — examines the threats and vulnerabilities of a P2P network and content.
- *Anonymity* — examines how the overlay network hides the physical location of peers from each other.

In section II basic understanding of the protocols is presented. Section III will elaborate detailed features of Unstructured P2P Networks. In section IV conclusion has been made, on the bases of the literature review done on above mentioned features. Not only positive but also negative points of these protocols have been investigated.

2. UNSTRUCTURED P2P FILE SHARING PROTOCOLS

An unstructured p2p network does not follow a particular topology; rather peers participating in overlay network are connected randomly. Such networks are easy to build and peers can enter or leave the network without disturbing the overall structure. However, the lack of structure in the network poses some limitations. Whenever a search is initiated, the query spreads in the network using flooding, random walks or expanding-ring Time-To-Live search. Such methods cause high signaling traffic, require each peer to process queries, thereby using more CPU and Memory, and does not guarantee results [1]. Availability of popular content is high as compared to the content which is rarely available on few peers. In this section, basic structure and working of some famous Unstructured P2P networks is introduced.

2.1 Fast Track/KaZaA

Fast track is a hybrid proprietary protocol that has been extensively used in file sharing. Fast track exploits the heterogeneity of the nodes by using two tier hierarchy of peers; Super Nodes (SNs) and Ordinary Nodes (ONs). The bandwidth, availability, CPU power, and NATed access of SNs are more powerful than ordinary nodes [1].

Whenever a user activates Kazaa, a client of Fast track first establishes a connection with a boot strap node which configures it as an Ordinary or a Super node (on the basis of bandwidth, availability, CPU power, and NATed access). Boot strap node is not a part of an overlay network but it is assumed to be available always. After configuring a peer as super node or ordinary node, bootstrap replies with a list of active super nodes. Kazaa client also have a list of 200 super nodes in windows registry.

Each ON has to connect itself with a SN to become a part of a network. It selects typically 5 SNs from the list (on the basis of work load) and sends UDP request to check if they are available to become parent. ON then creates a TCP connection with all the SNs which replied to the UDP request. For each TCP connection, SN and ON will exchange encrypted keys. ON send peer information (local IP address, username and service port number) to SN and gets refreshed list of available SNs [3]. ON selects one SN as its parent and disconnects other SNs. ON then uploads meta-data of files, to its parent SN, which includes file name, size, ContentHash, and key words. SN maintains a file index (DBB of files at its children) and ON index (list of children) [4].

Figure 2 describes file sharing process used in Kazaa. Peer 2 and peer 3 upload objects to their parent SN. Peer 1 will send a query, with keywords object 2, to its parent SN. If SN finds the file in its DBB it will reply peer 1 with IP address and meta-data of corresponding match. Otherwise, the query will be forwarded to other SNs connected in the network. Once the peer 1 receives result of a query; it gets IP addresses of the peer 2 having object 2. It may happen that some other peers in the network also have object 2. Peer 1 can start downloading the file from peers 2 (and other peers if any) using segmented file transfer.

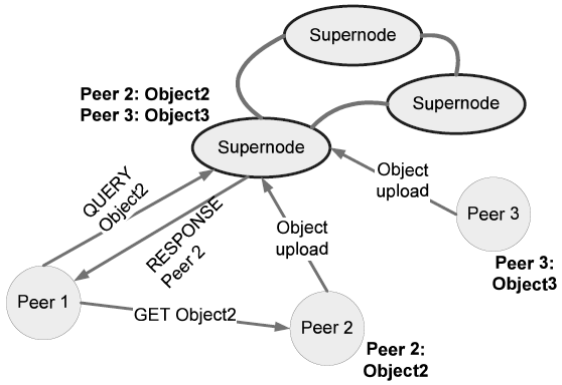

Figure 2: FastTrack - Search routing through SNs and downloads from ON peers

Fast Track exchange mostly two type of TCP traffic among peers: signaling traffic (includes handshaking traffic, metadata uploading from ONs to SNs, exchanging SN lists; and queries and replies. All signaling traffic is encrypted) and file transfer traffic (file is downloaded directly from peers without passing through intermediate SNs. File transfers are not encrypted and are sent within HTTP messages) [4].

2.2 BitTorrent

BitTorrent is a centralized, unstructured, peer-to-peer file sharing protocol. This protocol was proposed by Cohen in 2003 [6]. In BitTorrent multiple downloads of a file can happen concurrently. The basic architecture of end host network mainly consists of a tracker server, metainfo (.torrent) file, web server, original content provider, data content and end hosts. Since 2005, tracker-less BitTorrent model is also available. It uses distributed hash tables (DHT) as an alternative and instead of having one particular tracker-server every peer act as a tracker. Whereas, the basic mechanism remains same for both the systems so we have focused only on tracker-based system for comparison purposes [7]. Some famous clients are BitTorrent, µTorrent, BitComet etc. When a user wants to upload a file, it needs to create a torrent for it. While creating a torrent, piece size is mentioned in the BitTorrent client usually ranging from 32KB to 16MB. Data content is divided into pieces, to distribute it across the network efficiently, for frequent downloads to all interested peers.

Peer must obtain a *.torrent file* in order to download the specific data file. Torrent files are published on dedicated websites such as pirate bay, torrentz, and isoHunt etc. Torrent file contains information about the file, its name and length, along with URL of tracker and SHA-1 hash of all pieces of this file which is used to check data integrity of each downloaded piece. The torrent file runs

in BitTorrent client software to establish connection with other peers and continue download process [9].

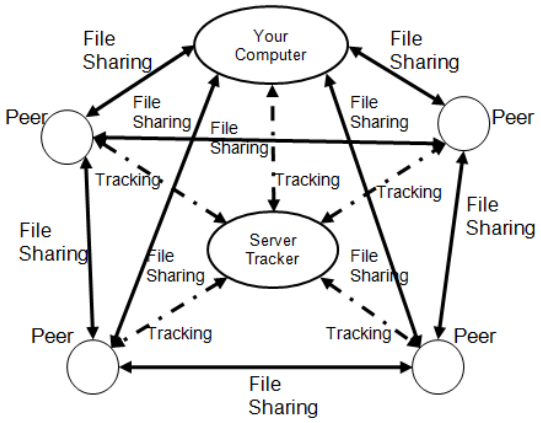

Figure 3: File sharing in BitTorrent using central tracker and peers

End user interested in downloading a file gets connected to the tracker which further contains list of peers currently having the required pieces of data file and how many copies of each piece is available as shown in Figure 3. All peers communicate with each other through a TCP connection. BitTorrent has implemented segmented file transfer (non-sequential) among peers and hence, uses a few algorithms for piece selection criteria. Generally, it applies rarest-piece first approach to avoid the loss of piece with minimum number of copies. A download is initialized with randomly selected piece as a first piece however; towards the end it prioritizes the missing pieces. BitTorrent discourages freeloaders (peers that only download and do not upload as much). To maintain consistent download rates BitTorrent applies a choking algorithm based on tit-for-tat approach [10].

2.3 Freenet

Freenet is a decentralized and fully distributed peer-to-peer protocol for storing and retrieving data. This protocol was proposed by Ian Clarke in 2000 [11]. The basic architecture of Freenet consists of peers, data store and routing tables. Every peer maintains its local data store and routing table. Routing table contains ids and keys of its neighboring peers. Peers communicate with one another by passing queries for storing and retrieving data. Every query contains a unique identifier (to avoid loops), hop to live (to specify maximum number of peers to visit), depth and key (to store or to search).

Peer must obtain a key in order to retrieve the specific data file. Keys can be searched by running a hypertext spider, published on dedicated websites. A peer *A* gets a key and checks its local data store to find data firstly as shown in Figure 4. If data is not found, routing table is consulted to find closing match for key and request is forwarded to the peer B along with time to live (decremented at every hop). This process continues until data is found or the TTL expires.

Any peer in the network interested in publishing a file needs to provide short text description of the file (keywords). The short text description is hashed using SHA-1 hash function resulting in pair of public and private key. Public key becomes the identifier of the file, and private key is applied as file signature for authentication, integrity check and any modification in file contents [12].The process of publishing file is similar to searching where key generation and hash can cause collision. On encountering collision, peer act as if a request for search has been made and reply along with data file. Otherwise every peer propagates file along the path, caching it and modifying the routing table entry, until time to live exceed.

Freenet provides anonymity of files, their contents and both the source and destination of file. The architecture uses flat namespace, stores only keys for files rather than their storage location. Large files are split into multiple hashed parts every part pointing to other part. Keyword-Signed Keys (KSK), Signed-Subspace Keys (SSK), Content-Hash Keys (CHK) are three methods used in Freenet for generating keys and hashing contents depending on the context [13].

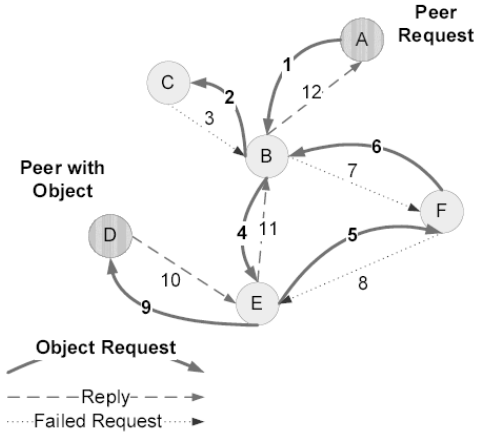

Figure 4: A typical request sequence in Freenet Characteristics of P2P systems.

3. Discussion Based on Evaluations Metrics

In this paper, an extensive study of each network under consideration, has been made. Some evaluation metrics has been discussed in Section I. In this section, an elaborative discussion and critical analysis has been given, while taking care of all the previously discussed qualitative measures.

3.1 Decentralization/Architecture

Fast track uses hybrid network in which sub network shows SNs are acting as central server to ONs attached to it. All SNs are attached to other SNs randomly. They change their connections frequently and do not follow any particular topology. BitTorrent protocol uses centralized network where all peers connect to a central tracker server in order to download the file. Freenet uses fully decentralized network in which all peers are at same level having loosely DHT functionality [14]. Each peer has a routing table in which keys are searched.

3.2 Lookup Method

In Fast Track SNs are responsible for processing queries, they communicate with other SNs to exchange information about queries. When search process ends, ONs communicate directly with peer to download file. BitTorrent uses *meta-info* file which is

called *.torrent file* containing URL of the tracker server. Tracker server has a list of all those peers that have the required file whether in pieces or complete file. Freenet is more of a file storage and retrieval system rather than file sharing. Publication and retrieval of data involves generating a "key" or using descriptive strings which are passed from one peer to another. Local Routing table information based on closest key match is used by peers to locate keys as Freenet only store file keys instead of files storage location. Peer to peer communication is packet oriented and carried out by UDP [13].

3.3 Routing Performance

In BitTorrent if a user is able to get the torrent for desired file, it is guaranteed to find a peer having required data. Because torrent have tracker information that helps to correctly locate peers. Torrents files are usually published on public torrent-hosting websites e.g., The Pirate Bay. In Freenet, no correlation between keys and content is present due to which blocks related to a file are scattered in different locations by which latencies occur in search. Freenet only guarantee to locate data (using Key search) in the network range of 7 Hops-To-Live (HTL) [16]. Routing performance seems good for popular content because it is expected to be on maximum possible peers. No particular algorithm exists for replication due to which rare files may become impossible to find. Fast Track gives some degree of guarantee to locate data, since queries are routed to the Super-Peers which has a better scaling. Good performance for popular content as compared to rare content. Fast Track is better than Freenet because we only need to perform search in SNs but Freenet search in each peer.

3.4 Download/Upload Performance

Fast Track uses segmented file downloading scheme which helps in increasing the downloading speed. But it does not share partially downloaded file with other peers. BitTorrent provides simultaneous downloading from multiple peers and shares partially downloaded files for better download performance [15]. If a file is found in Freenet, then we need to cache and backtrack towards the requested peer. That is why retrieval of data is very slow due to caching of file at every node along the path. It can take a very long time even days to retrieve files from it.

3.5 Content Lifetime

Fast Track allows data to be located as long as the peers holding the content stays connected. Data is lost only if peer leaves the network. In BitTorrent peers share/upload the content that is being downloaded at the moment. Hence, the content lifetime is short. Lifetime often depends on the popularity of the content. Freenet drop out infrequently requested data from caches, and no attempt is made to prevent this. Hence content lifetime is long for popular data but short for unpopular data.

3.6 Flash Crowd performance

Protocols that use swarming can handle flash crowd better and hence, BitTorrent can effectively handle flash crowds. BitTorrent caters high flash crowds by creating new seeds quickly. Flash crowds can also affect to overlay network and its functionality. Fast Track network can dynamically distribute SNs' tasks to new participating peers and thus be scalable in the face of large flash crowds [15]. In Freenet, Peer can leave in no time but joining of peer within the network can take days to handshake with other peers. Latency depends upon bandwidth and internet connectivity of peers participating in the network. Route lengths grow logarithmically with the number of users.

3.7 Scalability

Fast Track networks have the concept of Super Node to improve scalability. But its design has not been analyzed. As long as the work load of SNs is fairly distributed, there shouldn't be seen any major bottlenecks. On the other hand, Fast Track clients can refuse to be a SN may hinder scalability. BitTorrent is a swarm supporting protocol and swarming improves scalability. More Freenet users mean less bandwidth per user, as compared to HTTP and FTP. The better their Freenet caches are, the fewer infrastructures they need. Scalability can be improved by splitting big documents and combining the tiny ones. Scalability of Freenet can also be improved if caching of data at every node is ignored encouraging controlled replication.

3.8 Availability

Fast Track has good availability because of the distributed index and search mechanism with Super Nodes. The availability of components in BitTorrent protocol is unpredictable. If torrent file web servers are not available no download can be started. Similarly, if tracker server is down the file transfer is blocked. The availability of complete file depends on its popularity Freenet do not guarantee data availability or life-time. If data is not popular it will fade out and eventually all cached copies of data will disappear. Data availability is long for popular data but short for unpopular data.[17]

3.9 Reliability

In Fast Track if an ON fails, it will not affect the network. If a SN fails then ONs attached to it can connect to other SNs present in the list. In BitTorrent tracker is central and keeps track of peers having required pieces of file or complete file. As long as tracker server is online, file transfer is reliable. Its failure can stop overall download process. In Freenet, no hierarchy or central point of failure exists because all peers are similar. Therefore, the failure of peers does not cause network-wide failure, which gives good reliability and fault resiliency.

3.10 Security

Fast Track users may have to download whole files over and over again, because these protocols don't recognize corrupted chunks while downloading from multiple sources. File integrity check can be done only at the end of the download and one corrupted chunk ruins the whole file. Moreover, spyware fake content and viruses are found a lot in Fast Track [17]. BitTorrent ensures secure file transfer by generating SHA-1 hash for every piece. These hashes are stored in torrent file, when a piece is downloaded its hash is compared to check data integrity. Freenet do provide security measurements for denial of service attack by hashing and encrypting messages between nodes but still suffers from security problems such as man-in-middle, Trojan attacks, and needs to provide security mechanisms against malicious peers. A data block might pass through multiple nodes which will cache it on the way to its destination. The network protocol is designed such that an observer cannot tell the difference between a block being transferred as part of an upload and a block being transferred as part of a download. They cannot tell from observing the network, but if the attacker knows what the encrypted version looks like they can inspect client computer's Freenet cache to see if it contains a copy of it. Files are moved transparently, replicated and deleted without centralized location indexes.

Table 1 : COMPARISON OF VARIOUS UNSTRUCTURED P2P WIRED NETWORKS

FEATURES	FAST TRACK	BIT TORRENT	FREENET
Decentralization/ Architecture	Hybrid. Two tier hierarchy of SN and ON. No explicit central server.	Centralized model. Tracker keeps track of Peers and peers request info from tracker.	Fully decentralized. Routing tables to search string from peer to peer.
Lookup Method	Keywords are used for searching files. Super nodes pass keywords among themselves.	.torrent file is needed to download file by requesting piece information from tracker.	Keys and descriptive text strings are used for searching. Every peer participates in search.
Routing performance	Some degree of guarantee to locate data, better than Freenet. Good for popular content.	Full guarantee to locate data, good performance for popular content	Guarantee to locate data only in the network range of 7 Hops-To-Live. Good for popular content.
Download/ Upload Performance	Better downloading; do not share partially downloaded files.	Leader due to its Advanced Download Distribution Protocol.	Very slow due to caching at every peer.
Block Size	64 KB (for downloading only)	32KB - 16MB (for both downloading and uploading)	Latest version uses 16 KB (for both downloading and uploading)
Content Lifetime	As long as the peer holding the content stays connected.	Short. Lifetime often depends on the popularity of torrent; old torrents are often useless.	Long for popular data but short for unpopular data.
Flash Crowd performance	Medium.	High.	Low.
Scalability	Medium. Super Nodes may become a bottle neck in local network.	Good due to swarming.	Difficult to test. Improves if caching of data is ignored.
Availability	Good availability because of distributed index and search using super nodes.	Block new downloads if no torrent file server available. If tracker goes down OR seed becomes unavailable, it will hinder download performance of peers.	Do not guarantee data availability or life-time; if data is not popular it will fade out & eventually all cached copies of data will disappear.
Reliability	Good. If a super node fails the ordinary peers are reassigned to other Super Peers.	If tracker goes down OR seed becomes unavailable, it will hinder download performance of peers.	Good. No hierarchy or central point of failure exists.
Security	Low; Threats: flooding, fake content, viruses, etc. Spywares monitor the activities of peers in the background.	High; Centralized Tracker manage file transfer and allows more control which makes it much harder faking IP addresses, port numbers, etc.	Low; Suffers from man-in-middle and Trojan attacks.
Anonymity	Users are not fully anonymous in the Fast Track network.	Not anonymous.	Fully anonymous.

3.11 Anonymity

In Fast Track users are not fully anonymous, because during downloading a peer connects to source peers using IP addresses. But peers visited during search process are anonymous. On the other hand, there is no anonymity of peers in BitTorrent because torrent file has IP of associated tracker and tracker has addresses of peers containing the specified file. Freenet protocol is considered to be fully anonymous. It does provide the anonymity of clients and file content. Clients add a file to the network but do not know the actual storage location. Only file keys are exposed to user. Information about file content and storage is kept private by employing various levels of encryption as the data traverses through the network. But Freenet does not provide anonymity for network usage. A statistical analysis of network traffic can help the attacker in determining the source of data [18]. Depth value may also help in determining the source of file.

4. CHALLENGES

Although different applications are offering cutting edge features for file sharing. Still content storage, content search and life time are some problems of serious concerns. Security, privacy and data integrity also need to be addressed.

Unpopular Content search is also a big problem in unstructured P2P networks. Searching data, which is available on small number of peers becomes almost unavailable. Reason behind this is no current application is actually guarantying such deep search. Additionally, lack in content replication algorithms is another reason for rare files to become unreachable.

5. CONCLUSION

This research presents a comprehensive comparative analysis of various techniques in unstructured P2P for wired systems based on various qualitative measures. P2P exploits decentralization which provides better scalability, reliability and anonymity but it makes metadata more vulnerable to attacks. Unpopular Content search is

also a big problem in Unstructured P2P networks. It becomes highly unlikely to get data if it is present on few peers only. Ideally Freenet should find data in $[\log(n)]^2$ but it does not guarantee. Moreover, lack in content replication algorithms causes rare files to fade out. Decentralization used in Fast Track has no data availability problems but it lags data integrity which results in 50% of data pollution in its network. In BitTorrent User may spend time in finding *.torrent file* but searching peers takes constant time and depends upon connectivity and bandwidth of user. Searching in Fast Track is better than Freenet due to hierarchy of SNs and ONs. Major challenges for P2P networks are content storage, content search and life time. Security, privacy and data integrity is also a challenge. Therefore, researchers are working to improve performance of P2P networks.

This research is focused on the features and open challenges of unstructured P2P systems in wired environment. Characteristics and challenges of File sharing on wireless systems is another active area of research which can be addressed in future.

6. REFERENCES

[1] A. Jantunen, S. P. (2006). Peer-to-Peer Analysis: State-of-the-art. Tampere University of Technology.

[2] E. K. Lua, J. C. (2004, March). A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. IEEE Communications Survey and Tutorial.

[3] J. Liang, R. K. (2004). Understanding Kaaza. Polytechnic University Brooklyn.

[4] J. Liang, R. K. (2004). The KaZaA Overlay: A Measurement Study (2005). Polytechnic University Brooklyn.

[5] Shi, J., Liang, J., & You, J. (2005). Measurements and understanding of the KaZaA P2P network. In Current Trends in High Performance Computing and Its Applications (pp. 425-429). Springer, Berlin, Heidelberg.

[6] Cohen, B. (2003). Incentives Build Robustness in BitTorrent. Proceedings of P2P Economics Workshop. Berkley, CA, USA.

[7] Christian Luthold, M. W. (Feb, 2014). A Success and Failure Factor Study of Peer-to-Peer File Sharing Systems. University of Zurich, Department of Informatics. Internet Economics VIII.

[8] Wikimedia Foundation, I. (2015, May 11). BitTorrent. Retrieved May 15, 2015, from Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/BitTorrent

[9] H. Park, R. R. (2010). Peer-to-peer network protocols, cooperation and competition. Streaming Media Architectures, Techniques, and Applications: Recent Advances.

[10] Cohen, B. (2013, Oct 11). BitTorrent Protocol Specifications. Retrieved May 19, 2015, from BitTorrent.org: http://www.bittorrent.org/beps/bep_0003.html

[11] Clarke, I. (2001). Distributed anonymous information storage and retrieval system. Designing Privacy Enhancing Technologies. Springer Berlin Heidelberg.

[12] Clarke, I. e. (2002). Protecting free expression online with Freenet. IEEE Internet Computing .

[13] Freenetproject.org. (2015). The Freenet Project - /faq. Retrieved May 16, 2015, from The Freenet Project: https://freenetproject.org/faq.html

[14] Moujane, A., Chiadmi, D., Benhlima, L., & Wadjinny, F. (2009, May). A study in the P2P data integration process. In 2009 IEEE/ACS International Conference on Computer Systems and Applications (pp. 57-58). IEEE.

[15] J. A. Pouwelse, P. G. (2005). The Bittorrent P2P File-Sharing System: Measurements and Analysis. Peer-to-Peer Systems IV. Springer.

[16] CH. Ding, S. N. (2003). P2P Networks for Content Sharing. Grid Computing and Distributed Systems Laboratory. Australia: University of Melbourne.

[17] Shakimov, A., Varshavsky, A., Cox, L. P., & Cáceres, R. (2009, August). Privacy, cost, and availability tradeoffs in decentralized osns. In Proceedings of the 2nd ACM workshop on Online social networks (pp. 13-18). ACM.

[18] Zhang, J., Duan, H., Liu, W., & Wu, J. (2011). Anonymity analysis of P2P anonymous communication systems. Computer Communications, 34(3), 358-366.