Engineering Sciences International Research Journal
ISSN (Print) : 2330 – 4338 ; Volume 1 Issue 1 ‖ February. 2013 ‖

A Survey on File Sharing in Peer-to-Peer Systems

Phillip Kisembe[1], Jeyasekar Anthony[2]

Abstract: Internet today is based on strict client– server communication. This has made the communication channels to and from some central servers to become a bottleneck in the system. Distributed computer architectures are designed for the sharing of computer resources by direct exchange. The essence of the peer-to-peer design philosophy is to design protocols for end hosts or peers to work in collaboration to achieve a certain design objective such as the sharing of a large file. Content distribution is an important peer to peer application on the internet that has received considerable research attention. Content distribution allows computers to function in a coordinated manner as a distributed storage medium by contributing, searching, and obtaining digital content. We survey various schemes of content distribution for large files that are based on P2P computing, with focus on what led to its popularity and success. At the end of this paper we discuss the use of new concept as a solution to the underlying schemes. In order to provide a solution for large file sharing, we propose the use of network coding concept in this paper. Using network coding, nodes on the distribution network are able to generate and transmit encoded information providing optimal utilization of resources of a network topology. It also maximizes the bandwidth and improves the time it takes to send data blocks between nodes on a network.

Keywords: Peer and Network Coding

1. INTRODUCTION

Peer-to-peer is one of the bases of operation of file sharing systems i.e. Gnutella, Napster, Bit Torrent, and many others. These systems are generally characterized by the direct sharing of computer resources (CPU cycles, storage, content) rather than using the help of a centralized server [1].

Basing applications on peer-to-peer systems derives to a large extent from their ability to function, scale, and organize in the presence of a population of nodes, network, and computer failures, without the need of a centralized server and the overhead of its administration. Administration, maintenance, responsibility for the operation, and even the notion of ownership of peer-to-peer systems are also distributed among the users, instead of being handled by a single company, institution or person. Finally, peer-to-peer systems have the potential to speed up communication processes and reduce collaboration costs through the ad-hoc administration of working groups [2].

We carry out a survey on peer-to-peer content distribution of large files that are based on P2P computing technologies, aiming to provide a comprehensive account of applications, features, and implementation techniques. Peer-to-peer systems are classified into three categories (communication and collaboration, distributed computation, and content distribution). We discuss content distribution systems further and provide a solution for large file sharing, we propose the use of network coding concept in this paper. Using network coding, nodes on the distribution network are able to generate and transmit encoded information providing optimal utilization of resources of a network topology. It also maximizes the bandwidth and improves the time it takes to send data blocks between nodes on a network.

2. NETWORK CODING

We propose a new scheme of content distribution of large files that is based on network coding. Network coding is coding at a node in a packet network (where data is divided into packets and network coding is applied to the contents of packets), or more generally, coding above the physical layer. Using network coding, nodes on the distribution network are able to generate and transmit encoded information providing optimal utilization of resources of a network topology. It also maximizes the bandwidth and improves the time it takes to send data blocks between nodes on a network. The principle behind network coding is to allow intermediate nodes to encode packets. Network coding makes optimal use of the available network resources and, moreover, computing a scheduling scheme that achieves such rate is computationally easy. Using ideas borrowed from network coding, we propose an end-system content distribution solution which optimally uses the resources of the network. Every time a client needs to send a packet to another client, it generates and sends a linear combination of all the information available to it (similarly to XORing multiple packets). After a client receives enough linearly independent combinations of packets, the client can reconstruct the original information.

3. CLASIFICATION OF P2P APPLICATIONS

Peer-to-peer systems are used because of various applications such as:

3.1 Communication and Collaboration

They provide direct, real-time, communication and collaboration between peer computers. Examples include chat and instant messaging applications

3.2 Distributed Computation

The systems make use of the available peer processing power

3.3 Content Distribution

Current peer-to-peer systems fall within this bracket of content distribution, which includes systems and infrastructures designed for the sharing of digital media and other data between users. Peer-to-peer content distribution systems range from relatively simple direct file sharing applications, to more sophisticated systems that create a distributed storage medium for securely and efficiently publishing, organizing, indexing, searching, updating, and retrieving data. There are numerous such systems and infrastructures.

Some examples are: the late Napster, Publius [3], and Gnutella [4]. The focus will be on content distribution, one of the most prominent application areas of peer-to-peer systems

4. P2P CONTENT DISTRIBUTION

Peer-to-peer content distribution systems create a distributed storage medium that allows for the publishing, searching, and retrieval of files by members of its network. When we examine current peer-to-peer technologies, they can be categorized as follows [5]: in this context suggests that they can be grouped as follows:

4.1 P2P Applications

This group involves content distribution systems that are based on P2P technology. They are further subdivided into the following two groups, based on their application goals and supposed complexity:

4.1.1 P2P file exchange systems

These systems are targeted towards simple file exchanges between peers. They are used for setting up a network of peers and providing facilities for searching and transferring files between them. These are typically trivial applications that adopt a best-effort approach without addressing security or availability.

4.1.2 P2P Content publishing and storage systems

These systems are targeted towards creating a distributed storage medium in and through which users will be able to publish, store, and distribute content in a secure and persistent manner. Such content is meant to be accessible in a controlled manner by peers with appropriate privileges.

The main focus of such systems is security and persistence, and often the aim is to incorporate provisions of accountability, and anonymity, as well as persistent content management facilities.

4.2 P2P Infrastructures

This category includes peer-to-peer based infrastructures that do not constitute working applications, but provide peer-to-peer based services and application frameworks. The following infrastructure services are identified:

4.2.1 Routing and Location

Any peer-to-peer content distribution system relies on a network of peers within which requests and messages must be routed with efficiency and fault tolerance, and through which peers and content can be efficiently located. Different infrastructures and algorithms have been developed to provide such services.

4.2.2 Anonymity

Peer-to-peer based infrastructure systems have been designed with the explicit aim of providing user anonymity.

4.2.3 Reputation Management

In a peer-to-peer network, there is no central organization to maintain reputation information for users and their behavior. Reputation information is, therefore, hosted in the various network nodes. In order for such reputation information to be kept secure, up-to-date, and available throughout the network, complex reputation management infrastructures need to be employed.

5. ARCHITECTURE OF EXISTING P2P FILE SHARING SYSTEM

5.1 Hybrid Centralized P2P File Sharing

Server facilitates the interaction between peers by maintaining directories of the shared files stored on the respective nodes of registered users of the network. An example is Napster. Napster was the application that made people aware of the possibilities with a modern P2P architecture. It was designed to let people exchange music files, MP3-files, with each other over some network. It provides a service that does not require any encryption or strict security.

A central directory server as shown in Fig. 1 maintain index on the metadata of all the files in the network. The metadata might include file names and creation dates. The server also maintains a table of user connection information including user's IP address.

A file query is sent to the server first. A query consists of a list of desired words. When the server receives a query, it searches for matches in its index.

The query results including a list of users who hold the file are sent back to the user who initiated the query. The user then opens a direct connection with the peer that has the requested file for downloading.

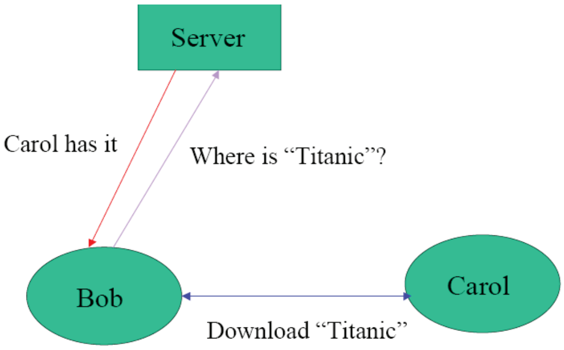

Fig. 1. A query in Napster

5.2 Pure Decentralized P2P File Sharing System

Peers have same capability and responsibility. The communication between peers is symmetric. There is no central directory server Index on the metadata of shared files as shown in Fig. 2 is stored locally among all peers. Examples are Gnutella and Freenet [6].

Each user acts independently. There is no central directory server. Peers directly change file information among each other in a query. Purely decentralized systems have open architectures and a self organizing structure.

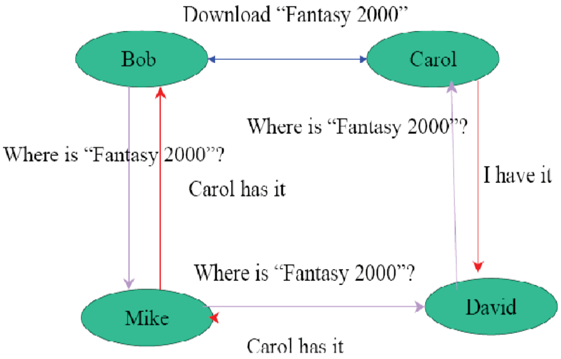

Fig. 2. A query in a decentralized file sharing system

Like most peer to peer systems, Gnutella builds a virtual overlay network with its own routing mechanisms allowing its users to share files with other peers. There is no central coordination of the activities in the network and users connect to each other directly through a software application that functions both as a client and a server.

5.3 BitTorrent

BitTorrent is an application layer network protocol used to distribute files. It uses a peer-to-peer (P2P) network architecture where many peers act as a client and a server by downloading from peers at the same time they are uploading to others. The serving capacity increases as the number of downloaders increases making the system self-scaling [7]. It also uses a client-server architecture where peers contact the server to find other peers that they may connect to.

A peer must be in one of two states. It is in the leecher state when it is still downloading the file while uploading pieces it has to other leechers. A peer is in the seed state if it has the complete file and is uploading to leechers. There needs to be at least one seeder in order for the torrent to be alive otherwise the leechers will not be able to finish. A file is split into equal size pieces, which are further divided into smaller blocks.

Blocks are the transmission unit of the network, but the protocol keeps track of what pieces have been downloaded. Each peer must maintain a list of peers it is connected to, which is called the peer set. Also a peer can only upload to a subset of this peer set called the active peer set. Peers also need to know what pieces of the content each peer in its peer set has [8]. By knowing what peers a peer can upload to and by knowing what pieces all of its connected peers have, BitTorrent can use this information in order to deliver content efficiently[7].

6. NETWORK CODING

Network coding is a rising topic of research in P2P systems. It was first proposed in the information theory community in 2000 [9], recognized the ability to code at intermediate network nodes in a communication session, in addition to the ability to forward and to replicate incoming packets. The concept of random network coding is proposed, where a network node transmits on each of its outgoing packets over a finite field, with randomly chosen coding coefficients. We propose the use of the concept of random network coding to improve the performance of the existing BitTorrent system.

6.1 Network coding and BitTorrent

Gkantsidis, [10] was the first to consider random network coding as a substitute for peer-to-peer file sharing based on exchanges of individual blocks (e.g BitTorrent). It is argued in [11] that, if peers can linearly combine all the blocks it has already received so far using exclusive OR (XOR) or randomly generated coding coefficients, and then transmit such coded blocks to other peers, the amount of time required to distribute a large file to all the peers in the network may be reduced.

Below is the proposed network coding file sharing system as shown in Fig. 3 where B has received blocks B1 and B2, and D has received block B2.

Although both B and D can serve E at this point, they may end up transmitting the same block B2 to E without network coding, and without explicit coordination via information exchanges, as proposed in BitTorrent. In this case, the upload bandwidth of B may be wasted. With the use of random network coding, however, B can transmit to E a coded block B1+B2, using the concept of XOR. E can then solve for B1 using the received blocks B2 and B1 + B2.

Similarly, with neither network coding nor periodic information exchanges, B may transmit block B1 to F, who has it already. With random network coding, B can instead transmit to F another randomly encoded block B1 + B2. Indeed, in [11], random network coding was to be used for file sharing, replacing BitTorrent. Gkantsidis et al. has made the claim that “the performance benefits provided by network coding in terms of throughput can be more than 2-3 times better compared to transmitting unencoded blocks.” If their statement is to be believed, one may conclude that network coding does offer significant advantages as compared to BitTorrent.

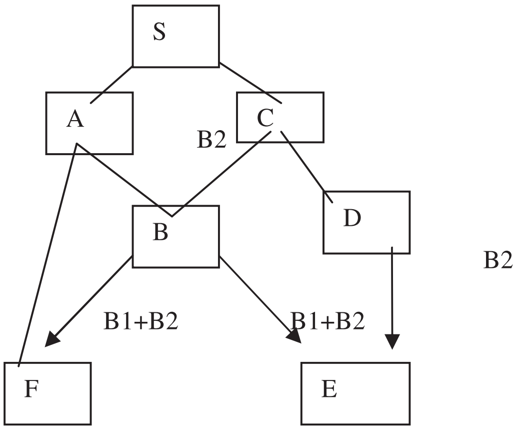

Fig. 3. An illustration of distributing two blocks B1 and B2 with network coding. S is the source peer.

7. CONCLUSION

Peer to peer systems are distributed systems consisting of interconnected nodes, able to self organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage, and bandwidth. Content distribution a prominent application area of peer to peer systems, is based on systems and infrastructures designed for sharing digital media and other data between users. Peer to peer content distribution systems range from relatively simple direct file sharing applications, to more sophisticated systems that create a distributed storage infrastructure for securely and efficiently publishing, organizing, indexing, searching, updating, and retrieving data.

The essence of P2P systems is to take full advantage of both storage and bandwidth resources on end hosts. P2P file sharing on the other hand utilized peer upload bandwidth to improve the downloading performance. The main challenge of P2P systems is its lack of robustness against peer dynamics, such as a flash crowd scenario in which a large number of peers join around the same time, or an alarmingly high attrition rate in which a large number of peers leave the system. By dividing the files into smaller blocks and serving them from multiple peers, BitTorrent was able to mitigate the negative impact of peer dynamics, but was still not able to provide any performance guarantees that are routine if dedicated servers are to be used.

Network coding, a simple yet far reaching idea, has been touted since 2005 to be the savior in P2P systems. We concur that P2P systems may be the most promising context in which network coding is to be applied, only because end hosts are able to afford the increased computational complexity introduced by network coding.

8. REFERENCES

[1] Hurvicz, M. 2001 “Groove networks: think globally, store locally”, Network Magazine, vol. 16, no.5, Miller, Freeman, pp 82 – 84.

[2] Schoder, D. and Fischbach, K. 2003 “Peer-to-peer prospects”, Communication ACM vol. 46, no. 2, pp 27–29.

[3] Waldman, M., AD, R., and LF, C. 2000. “Publius: A robust, tamper-evident, censorship-resistant web publishing system”, In Proceedings of the 9th USENIX Security Symposium.

[4] Clarke, L.; Sandberg, O.; wiley, B. Hong, T.W. 2000 ”Freenet: a distributed anonymous information storage and retrieval system. Design Privacy Enhancing Technologies”, In the proceedings of International Workshop on Design Issues in Anonymity and Unobservability, Berkeley, CA, USA, July 2000.

[5] Parameswarn, M.; Susarla, A. Whinston, A. 2001 “P2P Networking: an Information-Sharing Alternative”. Computer, vol. 34, no. 7, pp. 31-38

[6] Smart Decentralized Peer-to-Peer Sharing. http://www.ohaha.com/design.html..

[7] Legout, Arnaud, “Understanding BitTorrent: An Experimental Perspective,” in INRIA- 00000156, Version 3 – 9 November 2005.

http://hal.inria.fr/docs/00/04/31/40/PDF/bt_experiments_techRepINRIA 00000156_VERSION3_9NOVEMBER2005.pdf

[8] Bharambe, Ashwin R. “Analyzing and Improving BitTorrent Performance,” in MSR-TR- 2005-03.http://hal.inria.fr/docs/00/04/31/40/PDF/bt_experiments_techRepINRIA-00000156_VERSION3_9NOVEMBER2005.pdf

[9] Bharambe, Ashwin R. “Analyzing and Improving BitTorrent Performance,” in MSR-TR- 2005-03.http://hal.inria.fr/docs/00/04/31/40/PDF/bt_experiments_techRepINRIA 00000156_VERSION3_9NOVEMBER2005.pdf

[10] C. Gkantsidis, J. Miller, and P. Rodriguez, “Anatomy of a P2P Content Distribution System with Network Coding,” in Proc. 5th International Workshop on Peer-to-Peer Systems (IPTPS 2006), 2006.

[11] [109] ——, “Comprehensive View of a Live Network Coding P2P System,” in Proc. 6th ACM Internet Measurement Conference (IMC), 2006, pp. 177–188.

* * *

[1]*Phillip Kisembe, PO Box 1904KF Ghana/Student/SRM University/phillipkisembe@yahoo.com*
[2]*Jeyasekar Anthony, 603203-Kattankulathur/ Assistant Prof, Dept of CSE/SRM University/ajeyasenkar@yahoo.com*