Public File Sharing System Using Google Drive

Omkar A.Kulkarni
omkar.k@somaiya.edu

Vaibhavi A. Shah
vaibhavi.shah@somaiya.edu

Krishana B. Paste
krishana.p@somaiya.edu

Monic V. Bhanushali
monic.b@somaiya.edu

Guide: Prof. Swati Mali
swatimail@somaiya.edu

K. J. Somaiya College of Engineering, Mumbai University, Maharashtra, India

Abstract-- Peer-to-peer computing popularly known as P2P, is the sharing of computer resources and services by direct exchange of files between systems. We are proposing a new architecture which overcomes the disadvantages of the existing file sharing systems like DC, Torrent and so on. The proposed system tries to minimize resource consumptionat the user's side in terms of bandwidth and storage.This paper suggests an approach that improves the limitations of the traditional file sharing approach, it shows us the need of a new system and the risks attached with its implementation.

Keywords—*P2P, file sharing, categorization, Google Drive, Google Scripts*

*****

I. INTRODUCTION

Peer-to-peer computing popularly known as P2P, is the sharing of computer resources and services by direct exchange of files between systems. Information is distributed among the member nodes instead of concentrated at a single server. The P2P paradigm has proved to be a very effective approach for designing scalable and robust networking applications. This approach overcomes the scalability limitations of the traditional client/server approach.

Implementing a search engine isn't an easy task. There are many issues which need to be considered while designing and implementing the same. The downloading and uploading speed issues have to be handled. It also needs to prioritize its downloads. One can't ignore the fact that by the time the user is downloading files from the system, it is very likely that new files have been added or that the files have already been deleted.

The paper starts with the introduction to the topic and is followed by literature review in section 2. Section 3 details the system architecture. The pros and cons of the system arelisted in section 4. The paper ends with conclusion and future scope given in the last section, section 5& 6.This paper focuses on the basics of peer-to-peer systems and their working. It also discusses ways to share files efficiently on the network.

II. LITERATURE SURVEY

Peer to peer file sharing systems have been under research n development since the 19th century. The concept was popularized by file sharing systems such as the music-sharing application Napster (originally released in 1999)[2]. These are the existing systems of p2p sharing protocol with their features and their comparison with each other:

*A. BitTorrent:*BitTorrent is a protocol supporting decentralized P2P distribution of content over the Internet. BitTorrent is a centralized unstructured peer-to-peer network for file sharing. A central server called tracker keeps track of all peers who have the file[1].

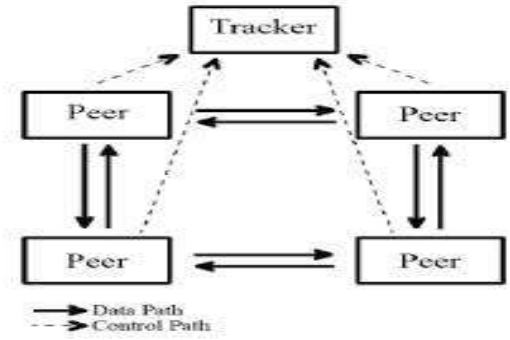

Fig. 1. BitTorrent

*B. Gnutella:*Gnutella is a decentralized unstructured peer-to-peer network. There is no constraint on the network topology. Gnutella approach has one big advantage: Gnutella works all the time as long as you can get to at least one other machine running Gnutella software, you are able to query the network[6].

Fig. 2. Gnutella

C. DC: Direct connect clients connect to a central hub and can download files directly from one another. Hubs feature a list of clients or users connected to them. Users can search for files and download them from other clients, as well as chat with other users. It is a text-based computer protocol, in which commands and their information are sent in clear text[5]. As clients connect to a central source of distribution (the hub) of information, the hub is required to have a substantial amount of upload bandwidth available.

D. Comparisons of systems:

Protocol	Algorithms	Advantages	Disadvantages
BitTorrent [1]	Rarest first	Guarantees high diversity of pieces in peer sets.	Problems may arise due to local variations.
	Choke algorithm	Encourages peers to offer higher upload rates to obtain better download rates.[3]	Difficulty in unchoking the peers.
	Endgame mode	Guarantees the complete download of a piece. Avoids potential delays at the end of content download.	Increases traffic in network.
	Anti-snubbing	Download rates recover much faster when they slowdown.	Poor download rates until the optimistic unchoke finds better peers.
	Strict priority	Completes the download of a piece as fast as possible.	Does not focus on other options for download.
Gnutella [6]	Query flooding		• Creates a bottle-neck in the network. • Large bandwidth is used. • Query response time is more.
	Query routing	Guarantees the download of file.	Dynamic querying is more efficient.
	Dynamic querying	• Bandwidth used in searching is very less. • System using it is scalable. • Flow control mechanism; so the packets will not be lost.	It alleviates problem by sending searches for widely distributed content to far fewer hosts while sending searches for rare content to as many or more hosts than traditional Gnutella searches.

E. Need Of System:

Gnutella system consumes a lot of network bandwidth. It also does not have a flow control mechanism. Gnutella systems are not scalable. BitTorrenton the other hand, uses a central server to store all the information about the file and the peers downloading the file, it suffers from so called "single point of failure". So both the systems are not fully efficient. Virus attacks arefrequent in all the above systems.DC systems are also not scalable. A query duplication problem in the DC protocol drains much of the hubs CPU and bandwidth resources. To adopt the advantages of all these three kinds of systems and to cover the disadvantages of them, a new system needs to be adaopted.

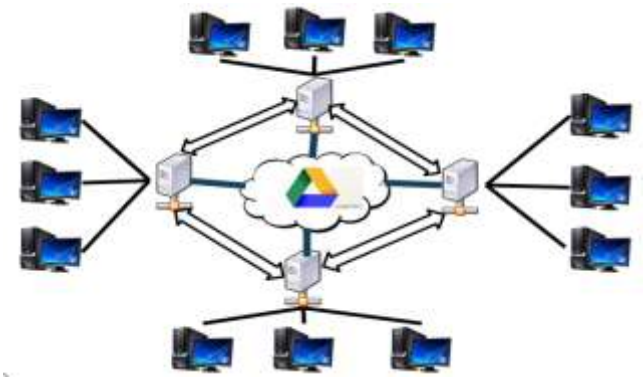

Fig. 3. Generic Architecture

III. System Architecture

*Problem Statement***:**

The proposed system will enable users to share files publically by just uploading it on their Google drive. A user with Google account, modern web browser and active internet connection can use this system. System will allow users to upload files for sharing, search and browse shared files, download shared files, get shared files on his/her drive along with categorization of files. System will maintain activity log. It will accept feedback, tags and ratings for file from users. Based on this system will be able to generate suggestions for users.

The system will consist of multiple servers. Each user will connect to its nearest server and need to register with it to be part of the system. Server will maintain list and the details of files shared by users connected to that system. This list will be generated automatically using Google API and will have metadata of all files and not the actual file.

*Registration:*User needs to create account using Google Id. User should create and share a specifically named folder with server's account id provided by us with owner permissions provided to server account.

Database files:This will include user list, files list and activity log of the users.

Searching & Downloading Files:File searching query will be responded by list of relevant files. Result will be generated by query flooding kind of communication between servers. To download files user will be provided with download link with metadata of the file. If user wants the file on his Google drive then that can also be done by using API and sharing that file in his folder on the drive.

*Categorization:*The categorization of the files will be done based on the hash tags provided by the users that will be associated with the files.

*Feedback & rating:*According to the number of downloads, amount of hits on a particular file i.e. on the feedback and rating of the files done by the user, the efficient searching will be provided by the suggestions depending on the ratings i.e. higher the file rating, it will considered first.

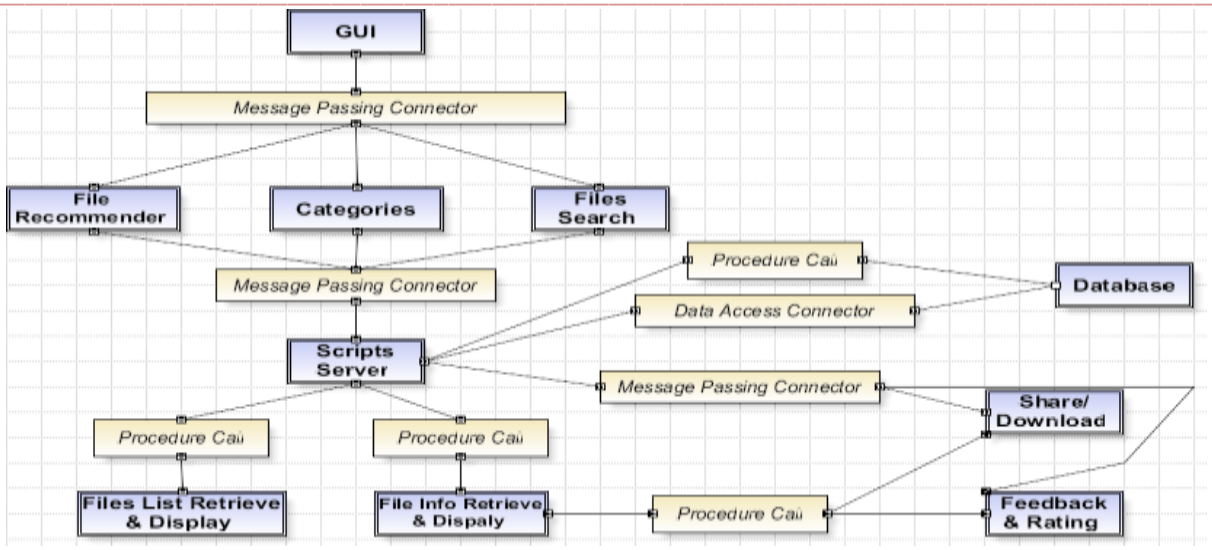

Fig. 4. System Architechture

IV. PROS AND CONS OF THE SYSTEM

Software Product Features:

1) Searching and browsing files (essential):-

- File can be searched from list of files based on some parameters. Files can be browsed based on various categories.
- Each search query will filter out list of files from files database and display result.
- Browsing files of particular category by selecting particular category will be possible.

2) Downloading or sharing file (essential):-

- User can download file of his interest. Along with file user browses he will be provided its download link from metadata of file.
- User can also just get that file shared with him if he doesn't want to download it.

3) Sign Up / Sign In with Google account:-

- System will authorize user based on his Google account credentials. For this at sign up user will have to authorize system to access his account's basic info.

4) Categorization of files:-

- Every File will have characteristic attribute information to help the further categorization.
- The files could be divided into multiple categories by using tags provided by the user.

5) Improved search results:-

- Search results should be improved based upon hits of file, search query, files age etc.
- Some good searching algorithm should be used to for this. Algorithm shouldn't cause much computational overhead to allow generation of faster search results.

6) Subscribing to categories & Generating suggestion mails:-

- User can subscribe to various categories of his interest for getting suggestions.
- User should get mails suggesting files that might be of his interest on periodic basis.
- User's subscription to various categories will be used for this along with hit count of files.

7) Most Downloaded and Recently added files:-

- Most downloaded files of each category can be obtained from hit count based sorting.
- On file addition to files list recently added files list shall be updated for this purpose.

V. CONCLUSION

The proposed system removes the disadvantages of existing p2p systems and provides 24*7 availability of the data, more security and easy access to the data. This paper states the process model to be implemented with the system, the management plan and the design documentations. We will have a complete system and its working on the drive and no memory will be wasted on the shared data. Thus it will be a more secure and easy to access system for any kind of user.

VI. FUTURE SCOPE

We can look upon the security issues if any, streaming can be made more and more efficient, different methods of file categorization and searching using user preferences and most shared, downloaded files. We can also improve the methods by the suggestions given by the users using our system.

REFERENCES

[1] Sharma Parul ;BhakuniAnuja ; KaushalRishabh " Performance analysis of BitTorrent protocol" Published in:Communications (NCC), 2013 National Conference on ,15-17 Feb. 2013 E-ISBN :978-1-4673-5951-1

[2] Arnaud Legout , G. Urvoy-Keller , P. Michiardi "Rarest first and choke algorithms are enough" Published in:Proceedings of the 6th ACM SIGCOMM , 2006

[3] Xia R.L. Mulappa J.K. " A Survey of BitTorrent Performance" Published in:Communications Surveys & Tutorials, IEEE , 19-April 2010
[4] Jenq-ShiouLeu ; Ming-Hung Huang " Piece-based file sharing over a Peer-to-Peer network in a heterogeneous network environment" Published in: Consumer Electronics, 2009. ISCE '09. IEEE 13th International Symposium , 25-28 May 2009
[5] GurvichP ,Koenigstein N , Shavitt Y "Analyzing the DC File Sharing Network" in Peer to Peer Computing(P2P) , 2010 IEEE 10th international conference on 25-27 August 2010
[6] YatinChawathe ; Sylvia Ratnasamy ; Lee Breslau ; Nick Lanha; Scott Shenker "Making Gnutella like P2P Systems Scalable" , 2003
[7] Google apps script:https://developers.google.com/apps-script/reference/(APIreferences) https://developers.google.com/apps-script/reference/drive/
[8] From Google scholar:http://cs.brown.edu/courses/cs295-11/2006/gfs.pdf http://static.googleusercontent.com/media/research.google.com/en/university/relations/facultysummit2010/storage_architecture_an_challenges.pdf