International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-5 Issue-1, March 2016

Efficient Peer to Peer Content Sharing using Wi-Fi

Sagar Vairagar, Shubhangi Vairagar, Shyam Sunder Gupta

Abstract*—As mobile environment is becoming popular; it becomes very effective and quick way of sharing user contents between users. This content sharing using Peer to peer scheme is limited due to the lesser distance factor and continuous availability of communication medium. Whenever large file size transfer is required, it requires more time and limitation on radius. Additionally content sharing is in plain format without any security considerations. Peer-to-peer networks offer several advantages over traditional client-server networking models like lack of connectivity to reliable hosts or servers and the use of inexpensive communication channel. Peer-to-peer network model has becoming popular in the wired broadband environment but not yet been effectively adapted to the mobile network environment. Many researchers are currently proposing and developing new P2P schemes for mobile environments with different wireless communication protocols. They are significantly adapted into applications such as the sharing of large files like multimedia and DB files between mobile devices. The peer-to-peer model based on mobile environment having several challenges like the limited on device processing power, limited memory, wireless data bandwidth, and available battery energy. With the proposed a peer-to-peer protocol, these specific constraints are addressed. This system also investigates the feasibility of a practical implementation of a peer-to-peer file sharing model of android smart phones using Wi-Fi technology. This includes an analysis of impacted performance by various variables that can be dynamically controlled in the protocol. Proposed work is done on leading Android platforms and found various optimal strategies which include minimizing the upload and download ratio to conserve battery life effectively, using flexible file segments to increase throughput, and decrease memory overhead.

***Index Terms*— Bluetooth, Wi-Fi, Peer-to-peer communication, wireless communication.**

I. INTRODUCTION

Recent market survey shows wide spectrum of smart phone devices apart from its traditional and regular uses. Smart phone users are now increasing with the time and continuously go on in increasing order. Improvements in smart phone version releases and effective wireless communications have made possible the sharing of device contents. In addition to smart phone devices, other electronic devices such as digital frame, digital camera has capability to communicate with smart phone devices. This causes to create peer-to-peer network model in which peers can either communicate with other nodes or perform interesting task with data.New mobile P2P systems seem encouraging in a new domain of applications based on physical place and framework, together with the possibility of using a wide variation of wireless radio access technologies. A promising solution is to use peer-to-peer sharing among smart phones, in order to consume free peer-to-peer wireless links versus expensive packet data networks for file sharing purposes. Although peer-to-peer concept for communication is more common in desktop environment, it is less popular in wireless environment. However since the performance of modern smartphones has improved greatly in such way that they now run complex and bandwidth-intensive protocols that earlier used in desktop environment.

Proposed contributions include the proposal of a peer-to-peer system for efficiently sharing large content such as multimedia among smart phones in a way that is transparent to the user and at minimal cost from a hardware and economic point of view. Userscan adapt peer-to-peer sharing from a traditional desktop to a mobile device environment, adapting to unique restrictions on energy and bandwidth usage.

Defined peer-to-peer or P2P computer network is one in which each peer node in the network can act as a client or server for the other remaining peer nodes in the network.This allows shared access to various resources such as files, peripherals, and sensors without the requirement for a central server. Generally, P2P networks can be set up within the home, a business, or over the Internet. Each network type requires all nodes in the network to use the same or a compatible program to connect to each other. Once connectedthrough program then tries to access files and other resources found on the other nodes. Thus P2P networks can be used for effectively sharing content such as audio, video, data, or anything in digital format.

II. LITERATURE SURVEY

A. System Model

In current system model, consider a distributed system consisting of mobile smartphones acting as peer nodes. These peer node are communicating with each other via ad hoc connections. Each peer effectively acts as a server or host depending on the nature of the transaction, and thus performs dual functions. Each mobile node needs to obtain a full copy of a file identified by its file name, time stamp, and possibly other required characteristics. The file itself is segmented with appropriate chunk size and each segment is identified by a hash value. Now all these individual segments/chunks may exist on different mobile nodes at any time. Each node may have a partial or complete copy of the file/s at any random time.

Revised Manuscript Received on 30 March 2016.
* Correspondence Author

Mr. Sagar Vairagar*, Department of Computer Engineering, SPPU University/ Siddhant College of Engineering, Pune, India.

Mrs. Shubhangi Vairagar, Department of Computer Engineering, SPPU University/ Siddhant College of Engineering, Pune, India.

Prof. Shyam Sunder Gupta, Department of Computer Engineering., SPPU University/ Siddhant College of Engineering, Pune, India.

The advantage of such peer-to-peer model is that a node may obtain the required segments from various nodes within network that it can actively connect to at any given time and then sequence them and finally merge them. This concept lends itself well to a mobile environment, where nodes are subject to high mobility and short-range connections are short-lived. This is in-contrast to a Mobile Ad Hoc Network (MANET), because devices are not responsible for forwarding and routing traffic unrelated to their own. There is only point-to-point communication between the nodes; a peer in current system model only learns of another if a direct pairing occurs. A Personal Area Network (PAN) modeled using Bluetooth, typically called a piconet. The IEEE 802.15 working group specializes in Wireless PANs (WPANs). When two devices come into proximity of each other, they plug in and start communicating as if in a wired network, with lockout of unauthorized peers being possible. One difference in current model is that there is no dedicated master or slave role as each device has the capacity to upload and download. As each party downloads a file, it becomes a candidate for sharing it with others, and thus contributes to a so-calledswarm that eases distribution of the file among participants. Thegreater the contributors that are within range of a peer, the quicker the file contents may be downloaded.

For peer-to-peer connectivity, generally considers only low-costshort-range connections such as Bluetooth that are standard and ensure interoperability in a heterogeneous mobile device environment. The nodes that described are similar to transient nodes on Tier 3 in Gnutella that only join the network for a short period of time and connect to a very limited number of other nodes. Thus, the network is not democratic; leeching is effectively limited by the presence of few nearby peers.

A nodethat has more connectivity options such as 3G, 4G may download content directly from a server/internet to obtain an initial seed or act as desktop computer with peer-to-peer connectivity itself. This so-called initial seed otsuper-node will then share its file content with other peers using a peer-to-peer network. Note that a single peer may draw segments from multiple sources and assemble them locally. Single peer may create multiple simultaneousconnections or transfer its content using one connection at a time depending type of requirement.

The user initially subscribes or registers to a service of interest, such as an OS update or interested video sharing service. Peers discover other nodes on the network through the process of scanning. Compatible and identical services are identified, and content that is assigned to a service is indexed and may be requested to be placed on a transfer queue. All individual segments are transferred and concatenated at the receiver node. After all segments arrived, they are assembled, and the content is registered with application handlers so that it becomes visible to the user.

The proposed model differs from a simple point-to-point pairing approach commonly found in a Bluetooth file transfer between two mobile phones. Generally Bluetooth file transfer simple scenario, there is no content subscription or registration, requires manual peer discovery, and no simultaneous or time-shared file transfer involving multiple phones in vicinity. The user is responsible for micro-managing all aspects.

B. Desktop Torrents

A comparison with peer-to-peer file sharing on desktop systems iswarranted. The popular BitTorrent protocol provides centralized tracking, whereby clients download torrent files that contain hashes corresponding to file segments, as well as the location ofa tracker that lists other clients to connect to. It is not plausible to translate a torrent implementation directly to the mobile environment, and in fact the only mobile applicationswidely available perform remote control of desktop torrents asopposed to native peer-to-peer transfers. The general problem is that BitTorrent does not recognize the sporadic connectivity inherent in mobile devices. BitTorrent utilizes a rarest-first 1446 algorithm, in which each peer maintains knowledge of the number of copies of each file segment inits peer set. It then determines which segments are rarest, and downloads those first while it still has the opportunity to do so. In current model, this stratagem does not confer an advantage as it is not foreseen that a large number of peers would be within the very short range permitted by a peer-to-peer protocol such as Bluetooth. BitTorrent also prioritizes the download of remaining blocks of a segment so that it can start sharing a complete segment once it assembles a copy of it. In the case of a peer-to-peer wireless protocol, it is assumed that packet loss in the transport would be relatively rare, and so no prioritization of packets is necessary.

The concept of super-seeding in BitTorrent is also superfluous.Here, the seeder initially advertises that is has no pieces,then un-chokes a peer by degree only if it has received confirmation that the segments made available earlier have already been distributed to other clients. Although this has the benefit of restricting uploads which are a big battery drain,it can result in stalling when interacting with only one other peer. It also requires feedback from other peers which is deemedexpensive in a mobile environment.

An analysis of BitTorrent claims that the average per peer download throughput improves with increasing file size, as service capacity improves with parallel multisegment downloading, but it is not shown whether the desktop results apply to a mobile scenario.

C. Other Related Work

It is argued that file sharing on mobile devices is in-feasible for the following reasons: limited bandwidth and high cost to the user, intermittent disconnections, and IP address changes due to network migration. Short-range networking technologies such as Bluetooth entail no cost to the user and are sufficiently fast to make file transfers very practical. In addition, the authors claim that the periodic sending and receiving of heartbeat messages from peers, as in the Gnutella network, would drain power. The power consumption of Bluetooth is so minimal that this is not a limitation. In this approach, nodes subscribe to content and retrieve it from adjacent neighbors only. Do not model a constantly changing topology of interconnected clusters of nodes, and so this avoids the problem of flooding of control messages or routing information in the network. Proposed model involves direct connectivity between mobile phones that are in close proximity for a short duration; yet, the transfers are organized through automatic content subscription, peer discovery, and file transfer involving multiple nearby peers; this is significant improvement over a one-time point-to-point single file transfer between a pair of phones that is managed by a user.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

A peer-to-peer architecture is proposed with a receiver-drivendiscovery algorithm to obtain the fresh status of peers that share files.
It relies on the ability to divide a network into multiple network-aware clusters, with all files within the samecluster being searched, first. This approach relies on searching only nodes within range and those that have been authenticated orsupport a common subscribed content type.

III. Proposed System

The improved or enhanced system makes changes to existing infrastructure and its working. Instead of using Bluetooth connectivity and window based smart phone devices; enhanced system uses Wi-Fi protocol for content sharing. Additionally, modified system supports different file formats and types. Modified system considers flexible chunk sizes, check pointing and downtime of failure node/s. Updated architecture diagram is shown in figure.

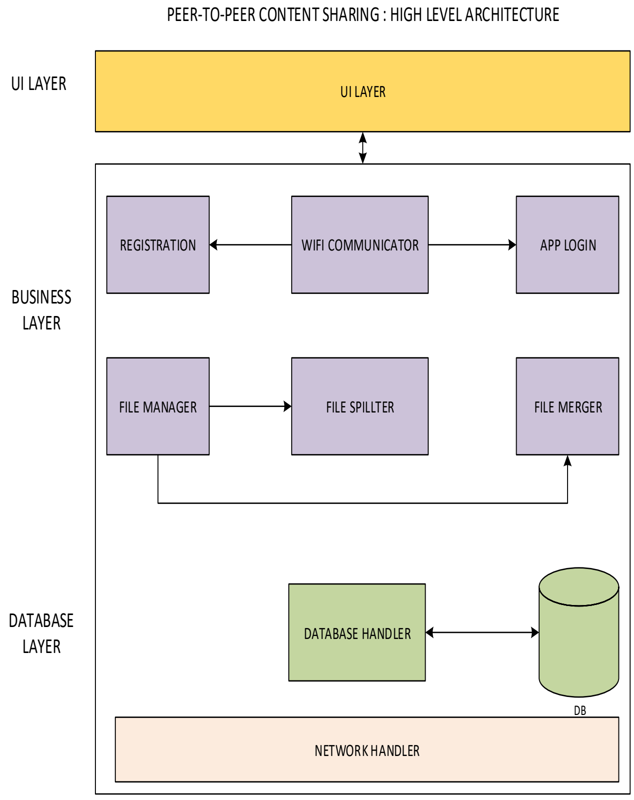

Figure 1

A. REGISTRATION/LOGIN

All device syncing process get done when application get launched. After installing application .apk user needs to perform app registration as initial mandatory activity which sync its IP address to backend. As part of registration process user has to enter unique credentials and other optional parameters. Registration inputs are get updated into database. Additionally, user will specify the type of current device node i.e. whether its super node or normal node. Once successful registration done, user has to use these credentials to login into application.

B. Wi-Fi Communicator

This module searches all active Wi-Fi within range using Android SDK in-built classes. On successful search, Wi-Fi communicator retrieves entire list of all active registered devices by active Wi-Fi. Wi-Fi communicator is vital component to perform content sharing since it is only responsible element to discover all active/inactive nodes within Peer-to-Peer network created through Wi-Fi. In base system each time it requires to perform connection pairing and service discovery features before starting content sharing by seed node. In enhanced system both of these functionality improved through communicator module.

C. File Managers

File manager reads devices internal memory and required supported format files. File manager acts both sides of Peer-to-Peer model i.e. client and server. In case of super node/initial seed file manager reads predefined file directory where all public files are copied and kept on device. Super node either reads or stores device file or downloaded internet files in this folder. While sharing file manager creates multiple chunks of files and stores in internal memory. Each chunk is unique represents of file portion of file and file manager is dump reader of these chunks treating each chunk as separate file.

File manager on normal peer/seed side acts as chunk requester. For selected file it requests absent chunks from different peer nodes in parallel and on arriving all chunks it merges all chunks into file.

D. File Splitter

When file need to share across all active nodes in Peer-to-Peer network, the file is divided into multiple chunks. Each chunk is representation of unique file portion of whole file. Also, each chunk has multiple attributes such as its unique checksum, contents, size, number, file checksum etc. Dynamic file chunk size is selected as per total size of file. Chunk size is directly affecting throughput of the file transfer. File splitter is invoked by Super Node only.

E. File Merger

Normal node can get list of all files and allow user to select file to open / play. On request, file merger request for all chunks from peers nodes in same network. Chunks are same as splitter by File splitter so merger starts reading chunk metadata. Once all chunks, its metadata read out File merger orders sequentially all chunks. Once all chunksareinorder,filemergingstartsandoncompletionfile either getplayedoropen. File splitter in invoked by super node andFile Merger generally invoked by peernodes.

F. Chunking Algorithm

Chunking algorithm read file attributes and split whole file into appropriate chunks. Created chunks are stored on device internal memory.Chunking algorithm read file attributes and split file into appropriate chunks. Created chunks are stored on device internal memory.

Steps to split file:-

1. Get filename and verify file is exists on super node folder along with required permission on file.
2. Read selected file metadata such as file format, type, size, header details etc. Once all meta data read out, select best suitable chunk size.
3. Calculate number of chunks of split out file using formula

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Number of chunks = Total Size of File / Selected Chunk Size

4. Create File Input Stream and divide whole file into number of chunks.
5. Create individual file named as file chunk and copy under /VS_Videos/ directory.
6. Return.

Steps to merge file:-

1. Get filename and verify required permission on file.
2. Get metadata of selected file from Super Node.
3. Calculate all active nodes and absent chunk files.
4. Retrieve all absent chunks from respective nodes through socket.
5. Order all chunks in sequence under /VS_Videos/ folder in internal memory.
6. Start merging all chunks until becoming flat file of whole size.
7. Return.

IV. RESULT

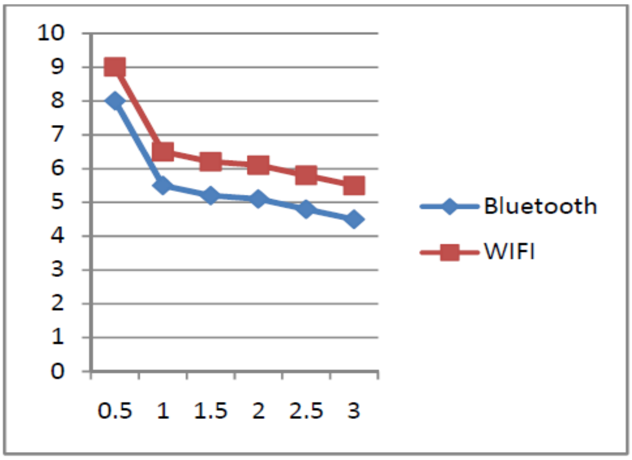

Figure 2

The plotted graph above shows throughput vs time comparison of base system and proposed system. This indicates transferring same file over Wi-Fi accomplish quickly without any burden. From the output graph it seems proposed system works well for different comparison factors such as throughput, distance range etc. Additionally chunk size, peer discovery works more dynamic nature which provides further benefits over base system.

V. CONCLUSION

Withproposed system, tried to improve many aspects of content sharing which caused limitation in base system. Thereby, the below conclusion is put forward in accordance with their original system as follows:

- Peer-to-Peer model of content sharing is tried to understood, done with analysis. After literature survey, basic system's pitfalls identified.
- In improved version, tried to figure out every aspect of challenges faced by basic system right from authentication, registration to flexible chunk sizes. In enhanced system improved both hardware and software dependent design factors.
- A sincere attempt has been made to borrow from the literature and conceived idea of uplifting the base system for range, availability, downtime attributes as it faces limitation on base system. Previous working is well understood and replaced with more appropriate and accurate system.
- With achieved content sharing and in comparison with older system, chunk splitting algorithm works much better than existing system.

REFERENCES

1. Piotr K. Tysowski, Pengxiang Zhao, KshirasagarNaik, "Peer to Peer Content Sharing on Ad Hoc Networks of Smartphones",In proceeding of: Proceedings of the 7th International Wireless Communications and Mobile Computing Conference, IWCMC 2011, Istanbul.
2. M. G. Williams, "Directions in Media Independent Handover", IEICE Trans. Fundam. Electron. Commun. Comput. Sci., vol. E88-A, no. 7, pp. 1772–1776, 2005.
3. R. Winter, T. Zahn, and J. Schiller, "Dynamo: A topology-aware p2p overlay network for dynamic, mobile ad-hoc environments", Telecommunication Systems, vol. 27, no. 2, p. 321, 2004.
4. A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems", Lecture Notes in Computer Science, vol. 2218, 2001.
5. A. Datta, "Mobigrid: P2P overlay and MANET rendezvous — a data management perspective", in CAiSE 2003 Doctoral Symposium", 2003.
6. K. Aberer, P. Cudr´e-Mauroux, A. Datta, Z. Despotovic, M. Hauswirth, M. Punceva, R. Schmidt, and J.Wu, "Advanced peer-to-peer networking: The p-grid system and its applications", PIK Journal - Praxis der Information sverarbeitung und Kommunikation, Special Issue on P2P Systems, 2003.
7. Symbian OS, the mobile operating system. [Online]. Available: www.symbian.com
8. R. Schollmeier, I. Gruber, and F. Niethammer, "Protocol for Peer-to-Peer Networking in Mobile Environments," in Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on, 20-22 Oct. 2003, pp. 121–127.
9. M. Matuszewski, N. Beijar, J. Lehtinen, and T. Hyyrylainen, "Mobile Peer-to-Peer content sharing application", in Consumer Communications and Networking Conference, 2006. CCNC 2006. 2006 3rd IEEE, vol. 2, 8-10 Jan. 2006, pp. 1324–1325.
10. T. Hakkarainen, V. Savikko, and A. Lattunen, "Generic engine for collaborative mobile applications," in Proceedings of the IADIS International Conference WWW/Internet 2005 (ICWI2005), 2005, pp. 243 – 246.
11. M. Bisignano, G. Di Modica, and O. Tomarchio, "JMobiPeer a middleware for mobile Peer-to-Peer computing in MANETs", in Distributed Computing Systems Workshops, 2005. 25th IEEE International Conference on, 6-10 June 2005, pp. 785–791.
12. G. Kortuem, J. Schneider, D. Preuitt, T. Thompson, S. Fickas, and Z. Segall, "When Peer-to-Peer comes face-to-face: collaborative Peer-to-Peer computing in mobile ad-hoc networks," in Peer-to-Peer Computing, 2001. Proceedings. First International Conference on, 27-29 Aug. 2001, pp. 75–91.
13. K. Kant, "An analytic model for peer to peer file sharing networks", in IEEE International Conference on Communications, May 2003, pp. 1801 – 1805 vol. 3.
14. A. Legout, G. Urvoy-Keller, and P. Michiardi, "Rarest First and Choke Algorithms Are Enough", in IMC'06, Rio de Janeiro, Brazil, 2006.
15. Yang, X. et al, "Service capacity of peer to peer networks", in 23rd Joint Conference of IEEE Computer and Communications, 2004.
16. Hu, T. H. et al, "Supporting mobile devices in Gnutella file sharing network with mobile agents", in 8th IEEE Int. Symposium on Computers and Communication, Sep. 2003, pp. 1035 – 1040 vol. 2.
17. Dhurandher, S. K. et al, "A swarm intelligence-based p2p file sharing protocol using bee algorithm", in IEEE/ACS Int. Conf. on Computer Systems and Applications, 2009.
18. Asadi, M. et al, "A scalable lookup service for p2p file sharing in manet," in Proc. of the 2007 Int. Conference on Wireless Comm. and Mobile Computing. New York, NY, USA: ACM, 2007.
19. Huang, C.-M. et al, "A file discovery control scheme for P2P file sharing applications in wireless mobile environments," in Proc. of the 28th Australasian Conference on C.S., 2005.