Peer to Peer File Sharing System

Article *in* International Journal of Scientific and Engineering Research · September 2014

1 author:

Adel Ali Al-zebari
Duhok Polytechnic University
24 PUBLICATIONS **392** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project   e-learning system based on semantic web technologies View project

Peer to Peer File Sharing System

Adel Ali Al-zebari

Peer to Peer file sharing systems are discussed in a lot of academic research. Many Peer to Peer applications are available which work on the computer and mobile, such as Gnutella, Napster, Bittorent, and SymTorrent. File sharing causes a lot of the traffic on the network, thus some of the technology is used to reduce the traffic and find the files easily. Mobile devices are becoming multifunctional, so why not create a peer-to-peer file sharing system between the mobile devices.

In this paper a Peer to Peer file sharing system between mobile devices is designed and implemented using Bluetooth as a communication protocol. The application allows J2ME and MIDP (Mobile Information Device Profile) enabled mobile devices to share and publish the files in the network over Bluetooth, search for specific types of files such as (music, picture, text, and program) and download them onto its local memory. It is possible to develop the system in the future, to add more features and the capacity to work on more than one operating system for mobiles.

The application was implemented and tested successfully between more than two mobile devices using an emulator in Wireless toolkit. The application fulfils the basic requirements for peer-to-peer file sharing. In addition, the application was implemented on two Nokia devices successfully, except the downloading files did not work, which is because the operating system for Nokia devices is Symbian. As a result some libraries did not work. The application allows the user to share and search for any type of file within range of the Bluetooth.

Index Terms: Peer to Peer, File Sharing, J2ME, Bluetooth, Query Routing Algorithms, Caching Policies, Auto Tagging.

——————————— ◆ ———————————

1 INTRODUCTION

Peer-to-peer file sharing systems have become more popular for sharing, exchanging, and transferring files among many users over the internet. In the peer-to-peer network, a central point is not necessary. In addition, a peer-to-peer sharing system is getting more attention in computing. Many peer-to-peer file sharing systems are available between computers, such as Napster, Gnutella, and Freenet over the internet. Most of the internet traffic is due to file sharing systems. Napster uses a server to communicate between the users, and each user should contact the server in order to get the data. Gnutella, however, depends on the client/server approach where the peer sends a query to all peers in the network. The main issue in peer-to-peer file sharing is deciding which protocol to use for finding and indexing information that generates the least amount of internet traffic.

The proposed system allows users with Bluetooth enabled mobile devices to connect to each other and share the files and data free. The application uses the J2ME platform with connected limited device configuration and Mobile Information Device Profile to create an application for mobile devices and PDAs. Any mobile or PDAs devices that support Bluetooth can use the system and share files for free between the nodes in the network. In addition, auto-tagging technologies, query routing algorithms, and caching policies have been studied in this report in order to improve the accuracy and reduce the search time for files in the network.

2 EXISTING SYSTEM

There are many peer-to-peer file sharing applications between computers and mobiles that share files between each other over the network. Some of these applications work with computers and some on mobiles. However, peer-to-peer applications cause heavy traffic on the internet. SymTorrent, BitTorrent, Gnutella, Napster, and eMule are the most popular applications in peer-to-peer systems. Many studies have been performed on the algorithm, modelling, and measurements of different Peer to Peer applications. Some of these are reviewed in this report in order to help the developer to create a good system between mobile devices or other potential devices.

2.1 BitTorrent

BitTorrent is a peer-to-peer file sharing protocol designed by Bram Cohen it has been one of the most popular file

sharing systems during the last few years used for distributing the content of data [1]. As Guo et al. reported, according to last measurement by CacheLogic, BitTorrent traffic on the internet represented 53% in June 2004 [2]. BitTorrent shares the same type of files between peers through the internet unlike the Gnutella, eDonkey, and eMule that share different files. BitTorrent is a peer-to-peer distribution network and this kind of network uses the bandwidth of its users.

2.2 Gnutella

Gnutella is one of the most popular peer-to-peer file sharing systems to date used to exchange and share data between the peers. There is no central point or server in the Gnutella network as opposed to centralized systems such as Napster [3]. Each peer in the Gnutella network acts as a router and rendezvous point where the HTTP used for file transport and TCP for message transport [4]. The main advantage in Gnutella is that there is no central point in the network. It depends on flooding in routing queries. Unlike Napster that depends on the server as a central point if the central point goes down, then the network becomes useless.

2.3 Napster

In 1999, Shawn Fanning created a technology to share music files over the internet that allowed users to share music [5]. The architecture of Napster was based on a central server in the network. Therefore, a peer first joins a network by connecting to a central point known as a broker. A peer passes all the music files that it makes available to the server. The server stores the received information about the music files in the database and the information of all the peers that currently logged into the server is stored in the database [6]. A peer sends a query to the server for a particular file then the server sends back a list of files and a list of peers that have the same file. The advantage of the Napster network is that the query is efficient and finding the files guaranteed. On the other hand, all the participating peers depend on the central server and if this server goes down the network become useless.

2.4 SymTorrent

SymTorrent is the only BitTorrent client for mobile Symbian OS. The target platform for SymTorrent is S60 3rd and 5th edition and it is freely available under the terms of GNU General Public License. SymTorrent was written completely in native C++. Multiple torrents can download at the same time in SymTorrent. In addition, it can continue the download after restarting the application. The UI in SymTorrent is built independently as part of an application in a separate DLL. This means that the porting to different devices is easier. Recently, SymTorrent has been downloaded more than twenty thousand times and the users used for downloading files between each other through GPRS and WLAN.

The performance of SymTorrent and MobTorrent and tested the applications in a real environment. In order to avoid delay it is important to make sure that all peers are live during the download process [7]. As shown in Table 3.1, the performance of J2ME application is similar with Symbian in 3G network. While in WLAN network, the performance of SymTorrent is much better than the MobTorrent. The reason is that the number of the peers in MobTorrent is limited to between 5 and 9, while SymTorrent can download from between 9 and 15 peers.

Table 1: Comparison of real mobile, download speed (kB/sec)

Application	3G	WLAN
SymTorrent N91	50	150
MobTorrent N91	48	79

3 TECHNOLOGY

This section presents the basics of J2ME. There are three platforms defined by the Sun Microsystems:

- Java 2 Micro Edition (J2ME)
- Java 2 Standard Edition (J2SE)
- Java 2 Enterprise Edition (J2EE)

Java 2 Micro Edition is a platform that is designed for small devices, unlike the other Java platforms J2SE and J2EE [8]. J2ME is the typical mobile client platform for small devices like mobile phones and personal digital assistants (PDAs). J2ME has a number of the technologies and specifications that are prepared for different parts of the device market. J2ME is more related with small devices that have a small amount of memory and power consumption. Moreover, it is intended to displace the various JDK 1.1- based products [9].

J2ME is divided into configurations, profiles, and optional packages. The basic sets of libraries and Java virtual machines are defined in configurations [10]. A series of APIs for a narrower range of devices is defined in profiles. Moreover, some technologies like Bluetooth and Multimedia are defined in optional packages. Figure 3.11 shows the relationship between the configurations, profiles, and optional packages in Java ME. Configurations and profiles are the main principles that make J2MEs modular.

4 JXTA

Is an open-source developed by Sun Microsystems, and it is a peer-to-peer platform. JXTA defines the most important protocols for peer-to-peer applications, which are discovery, group organization, and messaging [11]. The JXTA allows the devices in any network to communicate with each other as a peer-to-peer network [12]. There are some basic objectives of the platform, which are:

- Peers should able to discover another peer.
- Peers should self-organize into peer groups.
- Peers should communicate with another.

- Peers should advertise and discover network resource.
- Peers should monitor one another.
- The platform should not require the use of any particular computer language or operating system.
- The platform should not require the use any particular authentication, security, or encryption model.

The main goals of the JXTA are to support peer-to-peer communication between many devices such as personal computer, mobile devices, and PDA [13]. JXTA consists of three different layers: services, applications and platform. These three layers as; the core functionality in the platforms are used by services in order to access to the JXTA protocols, and applications that use these services are developed in order to access the JXTA network . The main technique that a peer uses to communicate with other peers in the network is the JXTA pipe, and there are some protocols which allocates how the JXTA pipe works [14].

5 BLUETOOTH

In 1994, Ericsson Company originated Bluetooth Communication, when the company researched an alternative to connect mobile accessories and omit the cable between mobile devices. Bluetooth is a wireless communication protocol; it is used to communicate between enabled devices. Bluetooth as a technology is an open specification low power consumption, low-cost, and short-range radio technology [15]. In addition, the connection between Bluetooth is not line-of-sight and the transmission is multidirectional.

The Bluetooth technology is like HTTP, FTP, SMTP, and IMAP as a communication protocol [16]. Table 2 below shows the comparison between three short-range wireless standards, However there are many types of wireless technology.

Table 2: Comparison of Wireless Communication

Feature and Function	Bluetooth Communication	Wireless LAN	IrDA
Connection type	Spread spectrum, spherical	Spread spectrum, Spherical	Infrared, narrow beam, line of sight
Spectrum	RF 2.4 GHz	RF 2.4 GHz (5 GHz for 802.11a/n)	Optical 850-900 nm
Transmission power	10-100 mW	100 mW	40-500 mW/Sr
Maximum data rate	3 Mbps	11 Mbps (54 Mbps for 802.11a, 802.11g)	9600 bps–16 Mbps (very rare)
Range	10-100 m	100 m	1 m
Supported devices	8 Active and 200 passive	Connects through an access point	2
Voice channel	Yes	No	No
Addressing	48-bit MAC	48-bit MAC	32-bit physical ID

6 BLUETOOTH NETWORK TOPOLOGY

A Personal Area Network (PAN) is a group of devices that are connected with each other over Bluetooth. A PAN network is known as a Piconet or a Scatternet. A Piconet is small network in ad hoc form containing a master and slave. A Piconet is a connection between two devices and may have about seven slaves. The device that initiates the connection acts as a master and the other devices are slaves. The point-to-point and point-to-multi-point connections are supported by the Bluetooth system. When two or more Piconets are connected together it is known a Scatternet. As shown in Figure 3.16, a slave in one Piconet may be a master in another one [17]. In addition, Bluetooth technology uses Time Division Multiplexing (TDM) in order to support full duplex communication [17].
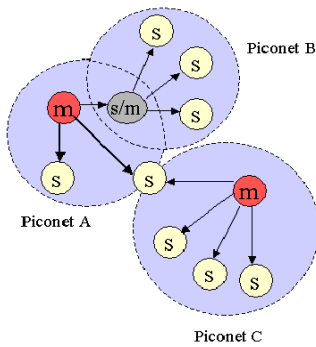
Figure 1: Bluetooth Network Topology

7 SYSTEM DESIGN

7.1 System Flow Chart

Figure 2 below shows the flowchart of the peer-to-peer application. Once the user launches the application, the user must enable Bluetooth in order to open and start the service; otherwise, the application will close automatically. If the user wants to search for files, they must select the search option and select which type of the file they want to search for, such as music, pictures, programmers, or videos. Once the user gets the files and they want to save the files, the user must get permission from the peer that has the file. In order to share the files over the network, the user must select the 'share' or 'publish' option, then access the local memory and then share the files over the network.
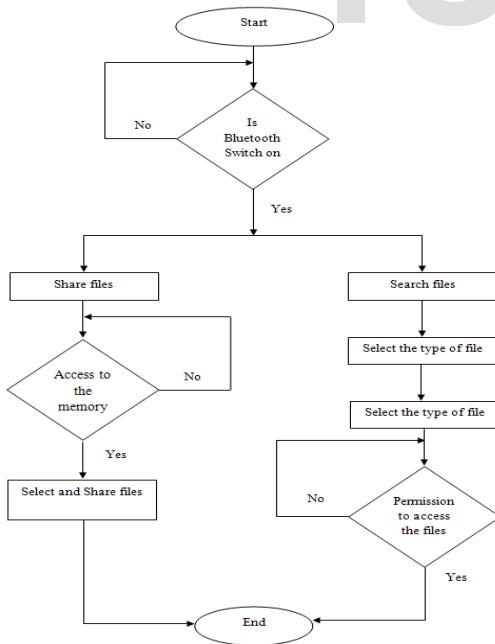

Figure 2: Application flowchart

7.2 System Scenario

The proposed system is that a group of peers join the network in order to share files and information. Peers must enable Bluetooth in order to be able to join the network, because the system is built on Bluetooth as a protocol to connect the peers. First, when the peer wants to share a file in the network they select files from the local memory and share into the network, other users in the network can find the desired files by selecting it by type. Secondly, if the users want to download or save the file, the peer that has the file may accept or not accept the download of the file. In case where the peer refuses to download the file, the request will be sent to another peer that has the same file. A peer that saved the file can publish or share it in the network because the peer in the network can act as server and client at the same time.
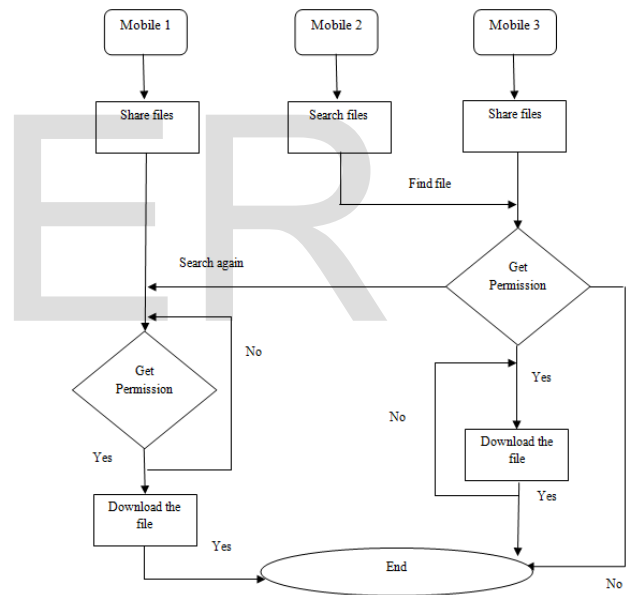

Figure 3: System scenario

8 SYSTEM IMPLEMENTATION

In order to investigate the application the implementation of the software is one of the most important tasks in this project. The main idea of the application in a peer-to-peer network means that one side acts a server and the other as a client and this is how the idea of this system works. The application is built using NetBeans7.1 and wireless toolkit and some libraries are used for the application such as MIDP components, MIDP components PDA and MIDP components, and WMA as resources. The application

consists of six classes and all of these are packaged in SharingSystem as one package.

9 TESTING AND EVALUATION

Testing the code in software development is one of the essential steps in order to make sure all the parts in the application are integrated and function together. System testing is the first step of the life cycle in the MIDlet. It is important to test the application as a whole in order to make sure that it meets the technical and functional specification. The system testing enables the developer to test and verify the architecture of the application. In testing the performance of the application there are some factors that affect the performance of the application, such as delays in the network, the speed of use protocol, and the size of the file. Testing the usability of the system is important in software development and it shows that the application should be easy to use by the user. Testing the user interface of the application is to make sure that the GUI meets requirements.

First, the code of the system for each class should be compiled and debugged to make sure that there are no errors. Secondly, the application should be implemented on the emulator to make sure that the important functions are working properly. The final step is to implement the application on the real device and share the files. Where two Nokia mobile devices were used to test the application it was done successfully, but there was one problem, which was downloading the file.

10 CONCLUSION

To conclude, the system designed allowed users to share and search for data using Bluetooth as a communication protocol. Peer-to-peer applications were studied in order to understand the main concepts of the network and help the developer to design a user-friendly application. Mobile devices and PDAs that were enabled with Java plus supporting CLDC and Bluetooth could use the application and therefore benefit from it. NetBeans was used in this project to help to write, edit, compile, and debug the code, Wireless tool kit was used to test and implement the application in order to learn the limitations and the errors before implementing it on the real device.

Some of the technologies were studied in this report, such as Auto tagging, cache policy and query routing algorithms, in order to increase accuracy and reduce the search time for data. In addition to that, JXTA and JXME were studied as a peer-to-peer protocol in order to develop the system in future. The main idea of Bluetooth has been studied in order to understand the main concepts behind Bluetooth communication. In this application, users can search for specific file types such as (music, picture, text, program, and movie). The user can also download the files

on the local memory of the device but he/she should get the permission from the peer who is responsible for these files.

During the test process, all of the classes were working completely and functioning properly. The errors and limitations were taken into account in order to develop the system and make it better in the future. The implementation of the system showed that the system fulfilled all the requirements. The evaluation of the system showed that it is easy to use and it is easy to search for and publish files that peers within the network have saved to local memory.

REFERENCES

[1] A. Vlavianos, M. Iliofotou, & M. Faloutsos, "BiToS: Enhancing BitTorrent for Supporting Streaming Applications". Proceeding of 25th IEEE International Conference on Computer Communication, pp. 1-6. 2006, doi: 10.1109/INFOCOM.2006.43.

[2] L., Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, & X. Zhang, "A Performance Study of BitTorrent-like Peer-to-Peer System". IEEE Journal on selected areas in communication, vol. 25, no. 1, p. 155, 2007.

[3] M. Sachs, & P. Piccard, Securing IM and P2P Applications for the Enterprise. Rockland: Syngress Inc. p. 246, 2005.

[4] R. Flenner, (Ed.). Java P2P Unleashed. Indianapolis: Sams. p. 72, 2003.

[5] O. C. Ferrell, & M. D. Hartline. Marketing Strategy. Mason USA: Thomson Higher Education, p. 373, 2008.

[6] A. J. Howe, "Napster and Gnutella: a Comparison of two Popular Peer-to-Peer Protocols" Unpublished. University of Victoria. 2002.

[7] I. Kelényi, P. Ekler, & B. Forstner, (n.d.). "Comparison of Mobile Peer-to-peer File-sharing Clients".

[8] M. J. Yuan, Enterprise J2ME: Developing Mobile Java Applications. New Jersey: Pearson Education, Inc. pp. 18-19, 2004.

[9] K. Topley, J2ME in Nutshell. United States of America: O'Reilly & Associates, Inc. p. 10, 2002.

[10] R. Rischpater, Beginning Java ME Platform. United States of America: Dreamtech Press. p. 8, 2009.

[11] E. Halepovic, "Performance Evaluation and Benchmarking of the JXTA Peer-To-Peer Platform" Unpublished. 2004.

[12] A. Akram, & R. Allan, " Comparison of JXTA and WSRF", http://epubs.cclrc.ac.uk/bitstream/1471/CCGrid07.doc. 2006.

[13] D. Brookshire, D. Govoni, & N. Krishnan, JXTA: Java p2p programming. United States of America: Sams, p. 40, 2002.

[14] S. Oaks, B. Traversat, & L. Gong, JXTA in Nutshell. United States of America: O'Reilly & Associates, Inc. p. 8, 2002.

[15] T. J. Thompson, P. J. Kline, & B. Kumar, Bluetooth Application Programming with the Java APIs. United States of America: Denise E. M. Penrose. p. 4, 2008.

[16] B. Hopkins, & R. Antony, Bluetooth for Java. United States of America: Apress. 2003.

[17] Q. H. Mahmoud, "Wireless Application Programming with J2ME and Bluetooth". http://dsc.sun.com/mobility/midp/articles/bluetooth1/ 2003.

Adel Ali Hasan Al-zebari, received his BSc at 2007 in Computer engineering at Mosul University. In 2013 he got a MSc degree in Communication Network Planning and Management at Portsmouth University / United Kingdom. He is a staff Member of Information Technology Department at Akre Technical College- DPU and he is assistant lecture in the IT Dept.