INTERNAL CHATTING SYSTEM FOR FSKKP STUDENTS, UMP ENVIRONMENT

VEETHIA A/P R.PERIASAMY

A thesis submitted in fulfillment of the requirements for the awarded of the degree in Computer Science

Faculty of Computer Systems & Software Engineering
Universiti Malaysia Pahang

May 2010

ABSTRAK

Sistem Chatting adalah alat komunikasi yang sangat umum yang telah digunakan di dalam moden ini dunia teknologi yang mutakhir. Sistem perbualan ini telah menjadi salah satu alat pengantara penting bagi setiap orang untuk berkongsi pengetahuan dan bahan-bahan melalui rangkaian. Jadi Internal Chatting System for FSKKP Students, UMP Environment ini dihasilkan dengan tujuan untuk meningkatkan komunikasi antara para pelajar melalui rangkaian sehingga ini dan ini juga akan memudahkan lagi bagi mereka untuk perbincangan yang berkaitan dengan pengajian. Sistem chatting ini dihasilkan dengan menggunakan konsep peer-to-peer, teknik multicast dan Rapid Application Development (RAD) metodologi. Sistem ini dibina dengan menggunakan perisian Netbeans IDE 6.8. Mesej teks dalam sistem ini menjelaskan konsep peer-to-peer,dan teknik multicast dengan sangat jelas. Hasil daripada sistem ini membolehkan pelajar untuk berbual di antara mereka melalui sistem perbualan dalaman ini.

ABSTRACT

Chatting system is very common communication tools that have been used in human in this modern cutting edge technology world. This chatting system has become one of the important intermediate tools for everyone to share knowledge and materials via network. So this Internal Chatting system for FSKKP Student, UMP Environment is developed in order to enhance the communication among the students via network so that this will be easy for the discussions related to their studies. This chatting system is developed by using peer-t-peer concept and multicast technique and also developed by using the Rapid Application Development (RAD) methodology. This system is built by using Netbeans IDE 6.8 software. The text messaging in this system does explain the peer-peer concept and multicast technique very well. The output from this system enables the student to chat among their course mates via this internal chatting system

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Chatting system is a peer-to-peer system where the users exchange text messages and files between the system's users. The user of the system is defined as client-server. Chatting system is a distributed programming which consist two distributed components, chat server and chat client. Chat client supports for all communication including requesting chat server location information from a location server and display received chat messages. Chat server will conduct chat session and manage all chat clients. Basically chat client starts the chat session by requesting the communication parameter (server name and port number). There are two type of communication between client-server which were control message (used to join and leave chat session, create chat room and switch to chat room) and chat message (supports only public chat message). Two transport protocols are applied for a chatting system that is Transport Control Protocol (TCP) and User Datagram Protocol (UDP). TCP takes place for control message while UDP takes place for chat messaging.

This same application goes for internal chatting system with the definition as a corporate client-server chatting system. This internal chatting system is build specifically for an organization/company on official purpose only.

The peer-to-peer application is widely used in most of the current chatting system. It is because peer-to-peer enables peers to establish direct contact to other peers and inquire about resources. Those systems offers peers the chance to send information through network. Moreover, the client-server both acts as provider and consumer of a service or an application. Decentralization in peer-to-peer makes no central coordinating authority for the organization of the network.

Consequently, in this project was proposed to do an Internal Chatting system for FSKKP Student, UMP Environment which applies peer-to-peer concept. The plus point of this project refers to the contacts which already build-in the system. This means the system will be easier for the students to communicate each other via network by connecting to the server which is ready for them to chat with those who are online. This type of communication tool is very useful for the students as they all engineering students will need to discuss among their course mates about projects and assignments.

1.2 Problem Statement

1.2.1 Existing chatting system's contacts need to add manually by using email addresses or ID name.

1.2.2 Difficult to find friend's contacts as the chat client might not have exact contacts of the user.

1.2.3 Usually the contacts of the system added manually using different web host emails such as yahoo, jarring and others.

1.2.4 If need to add contacts in an internal chatting system for an organization, it might be complicated as the name of the contacts is searched by using email.

1.2.5 Searching contact by name is more preferable than email as the length of the email address is long compared to name and also hard to remember.

1.2.6 In existing chatting system have invoked social problems that lead to unhealthy conversation among the chatters because they do not know their original profile of the personal they have being friend with via the chatting system.

1.3 Objective

The objectives of the system are:

1.3.1 To develop an internal chatting system for FSKKP student, UMP environment with build-in contacts

1.3.2 To develop an internal chatting system that applies peer-to peer concept and applies multicast technique.

1.4 Scope

The scope of the system:

1.4.1 FSKKP students from UMP

1.4.2 The database of the system is includes the information about the build-in contacts.

1.5 Thesis Organization

This thesis is about internal chatting system consist of six chapters. The first chapter contains the introduction on chatting system which gives a brief idea on what is meant by internal chatting system and what is the function of the system. The objectives and scopes of the system also provided in the chapter one. Literature review is given in the second chapter, where discuss about current system and previous system as stated. Different kind of methods and comparison on various software approach related to this system also included in second chapter. Techniques that can be used for chatting system also mentioned. Chapter three consists of methodology. This chapter elaborates on methods that will be used to carry out this internal chatting system. The implementation of the system is explained in detail in the chapter four. The results of the implementation will be discussed later in the chapter five. Finally in the chapter six will be discussing on the summary of the whole system.

CHAPTER 2

LITERATURE REVIEW

2.1 Existing System

2.1.1 Yahoo! Messenger

Yahoo! Messenger is a very common chatting system among people around the world and also known as (YM). Yahoo! Messenger is an advertisement-supported instant messaging client and associated protocol which was provided by Yahoo. It is free and can be downloaded from yahoo website. The ID that used for this system is allowed to access other Yahoo services such as Yahoo Mail. YM provides many functions for the user such as PC-PC, PC-Phone and Phone-t0-PC service, file transferring, webcam hosting, text messaging service and chat rooms in various categories.

The first version of Yahoo! Messenger was launched under the name Yahoo! Pager on March 9, 1998. The developer of this system is Yahoo. Yahoo! Messenger has been created for many type of operating system like Windows, Mac OS X, iPhone, WUI, and UNIX (but no longer supported). Yahoo! Messenger is a type of Voice Over Internet Phone(VoIP) or in other word can be said to be an instant messaging client. The license for this system is proprietary hardware. This software application can be downloaded through the website http://messenger.yahoo.com

Yahoo! Messenger uses TCP/IP transport layer protocol for all communication purpose. Sometimes it also uses peer-to-peer communication directly to the clients. The main purpose of applying peer-to-peer concept because a large amount of data to be transferred directly to the clients. This peer-to-peer application is very useful for the purpose of file transfers, voice conversations, webcams and certain instant messaging environments in yahoo messenger.

Yahoo! Messenger uses TCP/IP packets to allow communication in the system. But there is a slight difference in the packet where the TCP/IP header is extended by using its own application level header format. The Yahoo! Messenger header long is about 20 bytes long and identified as YMSG for the first four bytes. The header YMSG include the information about version, message length, service type, status, and session ID.

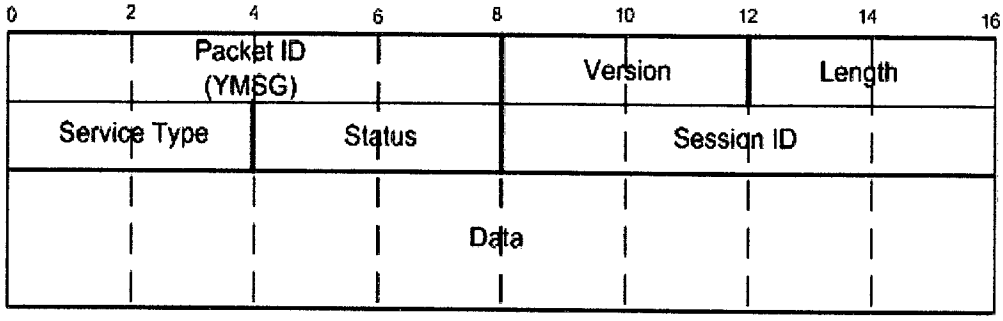

Figure 2.1 : Yahoo Messenger Packet by using TCP/IP protocol

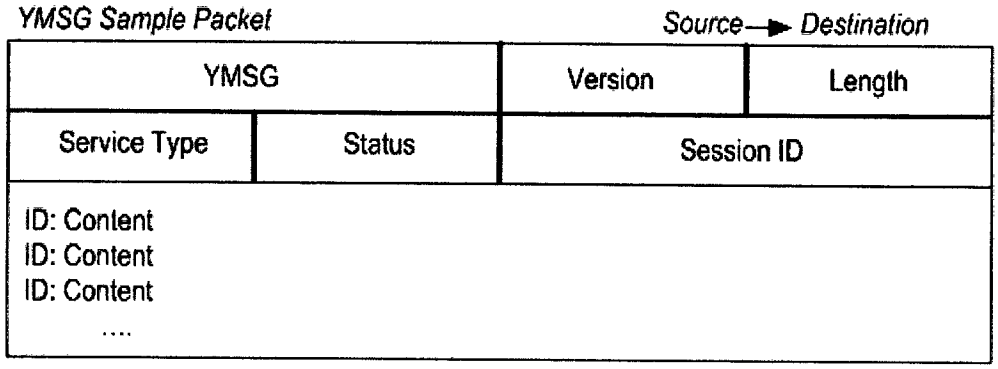

Figure 2.2 : Yahoo Packet Reference

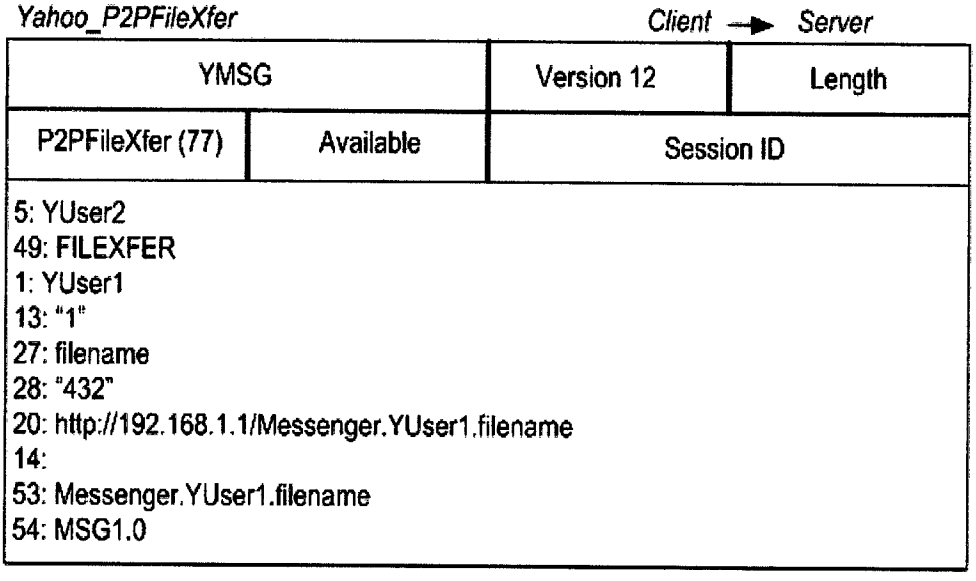

Figure 2.3 : Yahoo P2P File transfer

2.1.2 MSN

MSN formerly known as Windows Live Messenger and then its name was modified as MSN Messenger. Currently it is simply referred as MSN. MSN is an instant messaging client application created by Microsoft Company. It is compatible with much type of operating systems such as Windows XP, Windows Vista, Windows 7, Windows Server 2003, Windows Server 2008 and Windows Mobile. MSN have been part of Microsoft.NET Messenger Service since 2005. It's first version was created on July 22, 1999 and as Windows Live Messenger on 13th December 2005. This service has about 330 million users per month and it is also available in fifty (50) languages. It is referred as instant messaging client type of service. Its license is proprietary and advertising supported software. It can be downloaded freely form the website: http://messenger.live.com.

MSN also provides the basic functions as all other chatting systems does such as folders, PC-to-phone calls, interoperability, offline messaging, games and application, "i'm" initiative, S60 Platform and Xbox integration. MSN Messenger provides IM services including messaging, presence, file transfer and file sharing facilities. It uses Session Initiation Protocol (SIP) for instant messaging and MSN uses FTP for file transfer and sharing. Besides that it also uses peer-to-peer technique to establish connection via internet.

The communication between the MSN client and server are performed over TCP/IP. The TCP protocol is used with the client using various ports to generate the requests and the server using port 80 to receive the request and for the reply notification the server uses ports >1024. Meanwhile for the transport layer protocol, it uses Hypertext Transfer Protocol (HTTP) protocol. HTTP is one of the protocols in the TCP/IP suite which was originally

developed to publish and retrieve HTML pages and is now used for distributed, collaborative information systems. HTTP is used across the World Wide Web for data transfer and is one of the most used application protocols. HTTP specifies a request/response protocol. The HTTP protocol defines the message types of the client uses to request the web page and the server uses to respond.

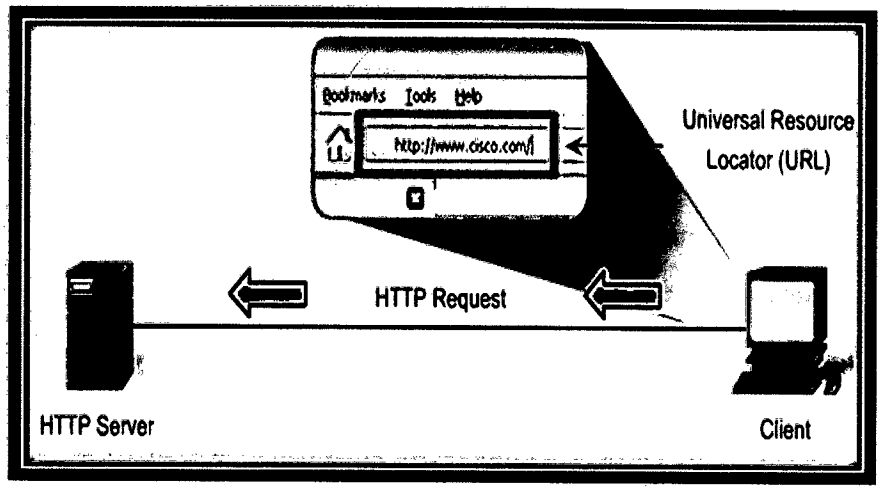

Figure 2.4: HTTP server and the client process.

There were two techniques used to develop this MSN Messenger. First is the MSN API (Application Programming Interface) and the second is Communication based Protocol Technique. The use of this MSN API's technique is for the purpose of providing interfaces for objects and events in the system. These interfaces are used for interaction in the MSN Messenger system by the instant messaging client. Consequently the client also has to intercept appropriate message calls sent to between the Operating System (OS) and MSN Messenger. The figure below does give clear picture of this technique.

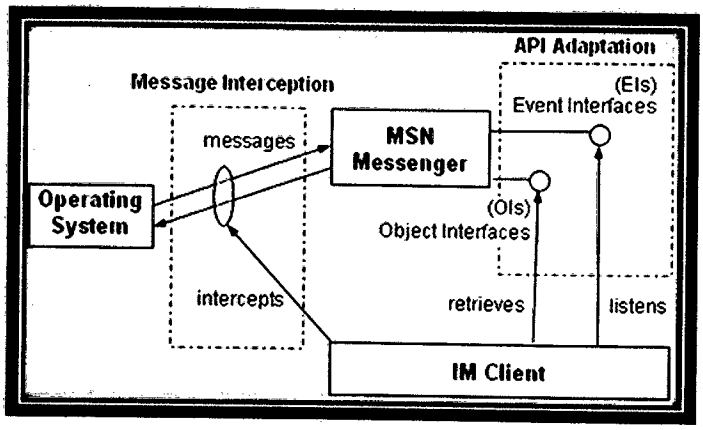

Figure 2.5: Architecture of API based technique

While the second technique is about the communication protocol in the MSN Messenger which enables the clients to communicate each other via the Messenger server and switchboard. There are two major software components of the architecture are Adaptation Library and IM Client for MSN communication protocol. The Adaptation Library component plays a role as a middle software layer, sitting between an IM client and the MSN Messenger communication protocol and responsible for interacting with the MSN Messenger communication protocol, such as establishing connection and authentication with the MSN Messenger server, and sending and receiving messages via the MSN Messenger switchboard. While the IM Client component listens to users' interaction and passes captured events and messages onto Adaptation Library. Moreover, the client is responsible for manipulating the contents of messages sent from other clients for presenting the adaptive threaded interface.

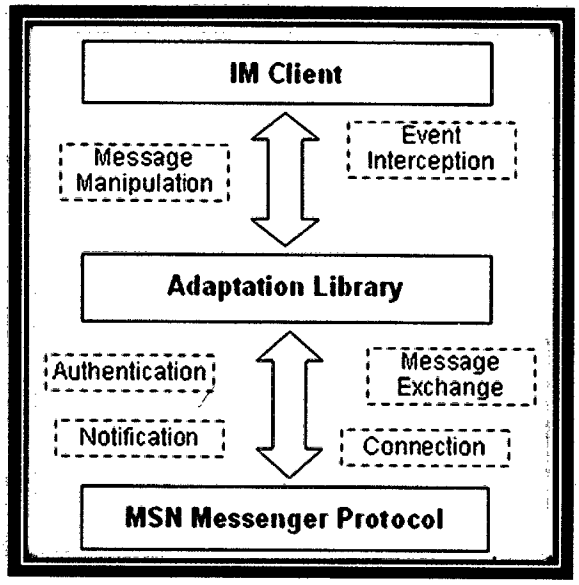

Figure 2.6: Architecture of the Communication Protocol-based Technique.

2.1.3 Skype

Skype is another chatting system application that widely used by chatters around the world for communication via network. Skype is a peer-to-peer (p2p) VoIP client developed by the organization that created Kazaa. Actually Skype is a software application that mainly used for voice calls over the internet. While its additional functions are instant messaging, file transfer and video conferencing. Skype was written by Estonia-based developer Ahti Heinla, Priit Kasesalu and Jaan who was originally developed Kazaa.

Originally the initial name of this application project was "Sky peer-to-peer" which was then reduced to "Skyper". Anyhow since this name was already taken so the "r" was dropped and left with the current name "Skype". Skype was launched first in August 2003. It does supports for Windows operating system and Linux too. Moreover, Skype also available in multi language. Skype is a type of voice over IP, instant messaging and video conferencing application. The license is freeware with some paid features. This software application can be downloaded from the website http://www.skype.com.

Skype uses peer-to-peer application to develop its application. All communication in Skype is secured by encryption. Skype application relies on supernodes which is responsible to facilitate communication among all the users. Skype is an overlay peer-to-peer network. There are two nodes that are ordinary host and supernodes (SN). An ordinary host is a Skype application that is used to place voice calls and send text messages. A super node is an ordinary host's end point on the Skype network. Any node with a public IP address having sufficient CPU, memory, and network bandwidth is a candidate to become a super node. An ordinary host must connect to a supernode and must register itself with the Skype login server for a successful login. Skype login server is an important entity in the Skype network where the user names and passwords are stored at the login server and, user authentication at login also done here. This server also ensures that Skype login names are unique across the Skype name space. Figure 1 illustrates the relationship between ordinary hosts, supernodes and login server. There is no central server in the Skype network which means that the online and offline user information are stored and propagated in a decentralized fashion as well as the user search queries.

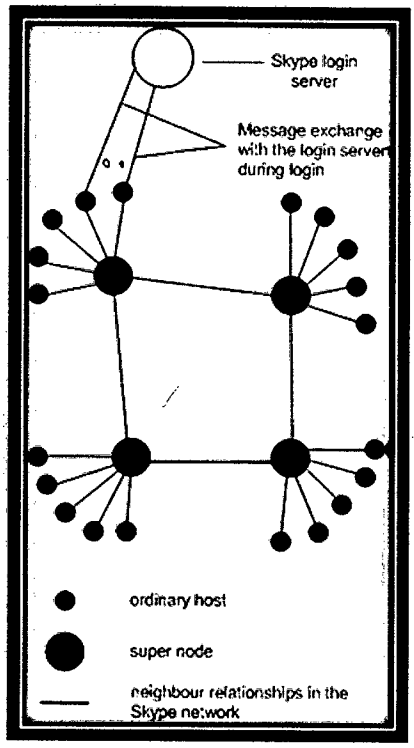

Figure 2.7: The relationship between ordinary hosts, supernodes and login server.

2.2 Chatting System Application

Chatting system is a two-way communication that is used by the users for the purpose of exchanging text messages and files between the system's users. It is more common to say as peer-to-peer system which supports two way communications. The user of the system is defined as client-server. Chatting system is a distributed programming which consist two distributed components, chat server and chat client. Chat client supports for all communication including requesting chat server location

information from a location server and display received chat messages. Chat server will conduct chat session and manage all chat clients. Basically chat client starts the chat session by requesting the communication parameter (server name and port number). There are two type of communication between client-server which were control message (used to join and leave chat session, create chat room and switch to chat room) and chat message (supports only public chat message).

There are two transport layer protocol suite are applied for a chatting system that is Transport Control Protocol (TCP) and User Datagram Protocol (UDP). TCP takes place for control message while UDP takes place for chat messaging.). Both protocols manage the communication of multiple applications. The differences between the two transport layer protocols are the specific functions that each protocol implements. UDP is a simple, connectionless protocol, described in RFC 768. It has the advantage of providing for low overhead data delivery. The pieces of communication in UDP are called datagrams. These datagrams are sent as "best effort" by this Transport layer protocol. Applications that use UDP including Domain Name System (DNS), video streaming and Voice over IP (VoIP). While TCP is a connection-oriented protocol, described in RFC 793. TCP incurs additional overhead to gain functions. Additional functions specified by TCP are the same order delivery, reliable delivery, and flow control. Each TCP segment has 20 bytes of overhead in the header encapsulating the application layer data, whereas each UDP segment only has 8 bytes of overhead. Applications that use TCP are web browsers, e-mail and file transfers systems.

Chatting system provides many type of function for the user such as text messaging, voicemail, file sharing, plug-ins, email service, webcam, URI scheme, offline messaging, interoperability, games application, PC-PC calls, PC-Phone calls and others.

For a real-time chatting system, there will be a limited number of contacts for a user and to be chat with at a same time with their contacts especially for conference chat. It is because in the system the number of contact per user already has been set. Besides that the events of the chatting system are only saved if the user saved before sign out. Otherwise the events such as text messaging, file transferring and others will be deleted automatically after the sign out. But in certain chatting systems do provides the ability to save the events or history by the system itself for a certain period.

Chatting system	**Number of host to chat with contacts**	**Message Archived**	**Own chat room**	**Offline messaging**	**Conference chat**
Yahoo! Messenger	150 host for normal chatting	Enabled till the chatters sign out.	Disabled on June 19,2005	Enabled	Enabled
MSN	7 host	Enabled with manual set up by the user	Enabled	Enabled	Enabled
Skype	9 host for conference	Enables for one month. Can be deleted if the user wishes.	Enabled	Enabled	Enabled

Table 2.1 Chatting system Application comparisons

2.3 Peer-to-Peer

Nowadays, it is more convenient to use peer-to-peer application because it operates directly to the clients. The communication to the servers consumes more time if compared to the peer-to-peer. That is why peer-to-peer is more preferred to be used for chatting systems as main technique and also as an alternative method to use especially for the purpose of file transferring feature.

The term peer-to-peer refers to the concept that the network of equals (peers) using appropriate information and communication systems, two or more individuals are able to spontaneously collaborate without the need of a central coordination (Schoder & Fischbach , 2003). Besides that a P2P network promises improved scalability, lower cost of ownership, self-organized and decentralized coordination of previously underused or limited resources, greater fault tolerance, and better support for building ad hoc networks. It also provides opportunities for new users scenarios that could scarcely be implemented using customary approaches.

The characteristic of peer-to-peer has many advantages to the network field. Firstly, sharing of distributed resources and service, where each node can provide both client and server application, that it can act as both a provider and consumer of services and or resource. Usually these network nodes are referred as servants' which derived from the terms client and server. Secondly, decentralization which is the major concept of peer-to-peer applies to the fact that no node has central control over the other and more clearly can be concluded that the communication between the peers takes place directly. The advantage of decentralization is an increased extensibility, higher system availability and improved resilience. The third characteristic is autonomy. This refers to each node in a peer-to-peer network can autonomously determine when and to what extent it makes its resources available to others.