CERTIFICATION OF APPROVAL

DEVELOPMENT OF VIDEO CONFERENCE USING JMF

by

Siti Safinas Mohd Rashidi (3058)

A project dissertation submitted to the
Information System Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF TECHNOLOGY (Hons)
BUSINESS INFORMATION SYSTEM

Approved by,

(Mr Saipunidzam Mahamad)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
DECEMBER 2005

CERTIFICATION OF ORIGINALITY

This is to certify that I'm responsible for the work submitted in this project, that original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by specified sources or persons.

SITI SAFINAS MOHD RASHIDI

ABSTRACT

Video Conferencing is well-planned to offer high quality of real time video and audio transmission. Video Conferencing has added extra flavors to students and lecturers interaction in UTP by having stable communication channel via real time video. Live feed from the media file and captured video can be broadcasted through the thousand of university's population in network by concentrating in reserving the quality of the video while at the same time reducing the cost of bandwidth. It's always great compromise in maintaining the quality of video with the cost bandwidth. Here it goes the need of good compression technique as compression will cause the data to lose some of the information and degrade the quality. The tolerable degradation is always at the author's spotlight.

The student has undergone 3 significant phases of system development which are Analysis, Design, and Coding. The critical function of Java Video Conferencing has been successfully implemented. Open the media file, capture the real time video, transmit the file, transmit the real time captured video, open the file in another computer, broadcast to the network attached computers and view the real time broadcasted video in the network attached computers. Communicating in text mode is an added feature in the Video Conferencing. This Video Conferencing has a room for improvement in achieving the best interaction mode in Information Communication Award. Video Conferencing is seen to have a bright future in realizing the need of Virtual Learning in UTP.

ACKNOWLEDGEMENTS

First off, I'd like to express my heartiest gratitude to my supervisor, **Mr. Saipunidzam Mahamad,** who has being a good supervisor with a blend of dedication, grace, and patience that made me working with him an unqualified joy. I am especially beholden to him, whose assistance and counsel have been indispensable. As always, I've profited from his advice, visits, opinion, and my completion of my project is demonstrably better as result of his contribution. I'm deeply thankful.

I also extend thanks to the FYP coordinator, **Mr. Nordin Zakaria,** who has provided us with overall direction (planning) that helped us build the credibility in managing all the projects being assigned with the deadlines.

I also extend thanks my Artificial Intelligence Tutor **Mr. Boshara Merghani Sukar**, who has giving me a very 'summary' lessons on networking and compression techniques. I also owe thanks to all **lecturers and staff** at **Information Technology / Information System department** and to all my **colleagues** at UTP, I've enjoyed not just completing the project, but the people are great as well. They are very supporting and good individuals. They are truly a gem.

Finally, I would like to thanks to my family for shouldering the bulk of work and being the techno-wizards morally and financially – my papa, my mother and my brother. Their love and support give special meaning to my life. A final word of thanks goes to all that is not mentioned here and those who have assisted me throughout the completion of FYP project directly and indirectly. Thanks! I owe many thanks for making this learning journey very exciting. May Allah bless all of you!

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES **PAGE NO.**

LIST OF TABLES

TABLES **PAGE NO.**

ABBREVIATIONS AND NOMENCLATURES

LAN	: Local Area Network
JVC	: Java Video Chat
UTP	: Universiti Teknologi Petronas
MF	: Java Media Framework
ISDN	: Integrated Services Digital Network
FYP	: Final Year Project

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND OF STUDY

Video Conference using Java Media Framework is a multi-platform compatible application that is grounded on Java technology. Java is chosen as the programming tool as it promise with fame, fortune, or even a job if this project is successfully implemented. Still, it is likely to make this project better and requires less effort than other languages.

Its utilize the Java Media Framework as the real time video conferencing is the most striking feature to be highlighted in this project. Java Media Framework provides powerful support for multimedia including timing, sound, animation, and video. It can add sound input, text-to-speech, speech recognition, and more.

1.2 PROBLEM STATEMENT

1.2.1 Problem Identification

Java Video Chat supports the multi-communications modes and multi-users which are via the text, audio (Internet Call) and video through one to one user or group based within the Local Area Network. As this messenger is intended for the big university population (covers the student and lecturer population), this messenger promote effective and efficient transmission of video and audio.

Java Video Conference is a platform independent (can run anywhere, Linux, Window, Solaris). Some of the students are using non-different platform like Linux and UNIX. Sometimes it's not possible or practical to have a face-to-face meeting with two or more people. Sometimes a telephone conversation or conference call is inadequate. Other times, an email exchange is adequate. Video chat adds another possible alternative. Video Chat is used when a live conversation is needed as physically presence to the same location is unachievable and visual information is an important component of the conversation. The well-distributed of students villages in the campus promotes the need to save the expense, time and safety of travel. It's major concern when they are urgently in need to have a live conversation in the mid-night and both parties are located at different villages. Java Video Chat in campus promotes security to the students.

1.2.2 Significant of the Project

Real time is at everyone's spotlight when it comes to communication technology. Paying attention in real time transmission is not enough as the people tends to compare the other important elements of communication like the quality of the video, network congestion due to this transmission any many more.

Networks built to transfer data are being strained by new applications sending video across the network. This strain will increase as these applications increasingly satisfy our needs nowadays. Serious use of video conferencing requires monitoring of performance to rapidly identify the onset of problems. Once this problem has been identified, it is necessary to analyze and fix it. This brief summarizes the author's approach to develop this Video Conferencing by reserving the video quality and promote the low-cost of bandwidth product.

Video streams can consume a considerable amount of bandwidth. It's even big in storage. Thus, compression is important to make it transferable over the network channel and distinguishable at the receiver's end. Many products use variable compression to achieve a compromise between quality and bandwidth consumption. The author will discuss in details about this in the Chapter 2 and Chapter 4.

1. 3 OBJECTIVES

The objectives of this project are as follows:

1. To develop real time video conferencing using JMF
2. To reserve the quality of the video
3. To reduce the cost of bandwidth

1.4 SCOPE OF STUDY

The scope of this project is to focus on developing a messenger that facilitates the communication in local area network. It supports the audio (like an internet call), text, and video. This project is intended to produce a fully functional prototype of very unique video conferencing like. It support the group based chatting, video conference (real time transmission) etc. It offers high speed and high quality video and audio transmission.

1.4.1 The Relevancy of the Project

Video Chat application will offers valuable experiences and very relevant to the people around the campus. They should be given an access to experience low cost solution for continuous video and audio feed in order to make best use of resources and all the learning assistance. With the video included, the learning process makes the intranet chat more spicy and taste as the students can see, hear and type at the same time. They can communicate in real time and through face to face and hence save the precious students time.

Video Chat application can be used by the lecturers, administrators, Rector, management, librarian and all the population in the campus. Guest lecturer brought into a class from different faculty, live Rector speech to the students' population within the Local Area Network, researcher collaborates with colleague at the other departments on a regular basis without lose of time due to travel. The administrators are on tight schedules collaborate on a budget preparation from different parts of campus can use this Video Chat without lose of time due to meeting arrangement. The faculty committee auditions a scholarship candidate via Video Chat. Librarian briefs the new freshman students about library doesn't have to be physically in the orientation hall. The librarian can brief the freshman with the real background of the library.

1.4.2 Feasibility of the Project within the Scope and Time Frame.

The time given to complete this project is 14 weeks. Within the time given, it's compulsory for this project to be successfully done. The scope of the project is clearly defined as to achieve the stated objectives. In order to make use of the effective time management, the Gantt chart was developed. It's clearly listed all the main phases and activities, significant milestones, targeted duration etc.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

Video Chat or Video Conferencing is a powerful marketing solution that enables to conduct full-featured, real-time events for up to thousands of participants despite of the geographical barriers. The geographical barriers are bound within the Local Area Network or Wide Area Network. The technology has been available for years but the acceptance it was quite recent. Complete with interactive audio and visual capabilities, the users can collaborate anytime, anywhere with just a PC and an Internet connection [JD-VS03]. Video chat application uses online or network technology to allow people to send audio and video recording that are delivered in real time. Video conferencing is adapted to the giant companies and education line. Big and giant companies are well-distributed in locations. The need to cut the cost of traveling budget for meetings is at their utmost concern [Mic-LM02]. Video conferencing is seen to have a bright future is realizing and promoting the virtual learning in the education [DBS-VC02].

2.2 VIDEO STREAMING

Capturing and streaming live video has become increasingly popular on the Internet and local intranets. There are three methods for sending video over a network from a media server to a viewing client [Mic02-2]. In the first method, the complete file is downloaded from the server to the client. The client can then play the locally cached video file. This method is only acceptable when the video file is relatively small and the client does not need to see real-time video. With the second method, the video file is downloaded to the client, but the client is able to start viewing the video once enough of it has been cached; the video plays while it is downloading. This allows the client to start viewing the video sooner, especially when viewing large video files. This method is still not acceptable for real-time video since the reception of the video always lags behind the transmission. The third method sends video from the server to the client in real-time, and the client may or may not cache the video. This method requires significantly higher bandwidth for faster transmission than the previous two methods and is the only method acceptable for viewing live video. If the video is not cached, the server must re-transmit video for clients wanting a second viewing. This of course causes an increase in network traffic and increases the demands of the network. However, for viewing video in a pay-per-view scenario, this disadvantage becomes an advantage since free replay of cached content is not available. Because real-time video transmission requires immediate delivery, intermittent video frames must sometimes be dropped to cope with limited throughput. Although it is usually acceptable to lose frames periodically, a large delay in transmission of those frames is not acceptable. This method requires balancing the competing needs for a high quality picture and a high frame rate. Streaming live video from a web cam requires the use of quick transport protocols like RTP [SCF96].

RTP is a higher layer transport protocol developed by the Internet Engineering Task Force (IETF) for providing end-to-end delivery of time-based media. It is typically used with the User Datagram Protocol (UDP) for its lower transport layer instead of the Transmission Control Protocol (TCP) [Mic02-2]. Connection-oriented transmission protocols like HTTP that are based on TCP are generally not acceptable for media transmission because of the additional overhead required for reliable connections such as retransmission of lost packets. IP delays in transmission due to packet retransmission can degrade live video. UDP does not guarantee that packets make it to their destination or arrive in the correct order; this is acceptable for video which can tolerate occasionally dropped packets [Wil02]. UDP is also connectionless which allows stream multicasting to multiple listeners. JMF provides hooks for using other transport protocols besides UDP. On a high-speed intranet where bandwidth is more plentiful, TCP or other native high-speed ATM protocols may be useful for their speed and Quality-of-Service (QoS) features. Because RTP does not provide any mechanisms for ensuring transmission rate or quality, a control protocol (RTCP) is used with RTP to monitor the quality of data distribution and to monitor and identify RTP transmissions [Sun99]. Figure 2 shows the relationship of RTP to underlying transport protocols and media frameworks that use RTP.

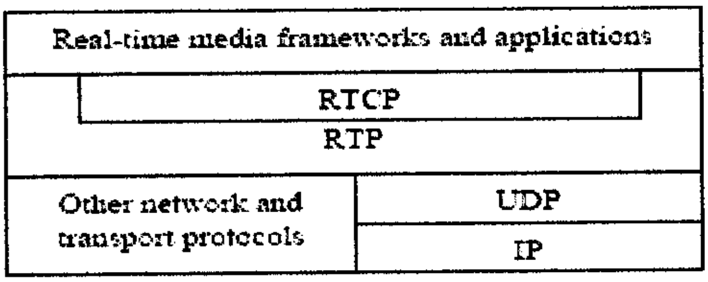


Figure 2.1: RTP Architecture from [JMF99]

In recent years, Java Media Framework for the Video Chat application has gained more and more popularity. Java Media Framework is a powerful method for working with time-based data in Java as it handles real-time transmission of video and audio. Java Media Framework requires the video and audio supported hardware. It promotes the real-time and progressive streaming. The receiver doesn't have to wait for the whole media to be downloaded before watching it. To enable real-time streaming, dedicated streaming media servers and streaming protocols, which are in this case utilizing the Real Time Protocol (RTP), are required. RTP is an Internet standard for transporting real-time data. It uses the unreliable UDP protocol to transmit packets.

The Java Media Framework (JMF) is a recent API for Java dealing with real-time multimedia presentation and effects processing. JMF handles time-based media, media which changes with respect to time. Examples of this are video from a television source, audio from a raw-audio format file and animations.

2.3 VIDEO COMPRESSION TECHNOLOGY

At its most basic level, compression is performed when an input video stream is analyzed and information that is indiscernible to the viewer is discarded. Each event is then assigned a code - commonly occurring events are assigned few bits and rare events will have codes more bits. These steps are commonly called signal analysis, quantization and variable length encoding respectively. There are four methods for compression; discrete cosine transforms (DCT), vector quantization (VQ), fractal compression, and discrete wavelet transform (DWT).

Discrete cosine transform is a lossy compression algorithm that samples an image at regular intervals, analyzes the frequency components present in the sample, and discards those frequencies which do not affect the image as the human eye perceives it. DCT is the basis of standards such as JPEG, MPEG, H.261, and H.263.

Vector quantization is a lossy compression that looks at an array of data, instead of individual values. It can then generalize what it sees, compressing redundant data, while at the same time retaining the desired object or data stream's original intent.

Fractal compression is a form of VQ and is also a lossy compression. Compression is performed by locating self-similar sections of an image, then using a fractal algorithm to generate the sections.

Like DCT, discrete wavelet transform mathematically transforms an image into frequency components. The process is performed on the entire image, which differs from the other methods (DCT) that work on smaller pieces of the desired data. The result is a hierarchical representation of an image, where each layer represents a frequency band.

MPEG stands for the Moving Picture Experts Group. MPEG is an ISO/IEC working group, established in 1988 to develop standards for digital audio and video formats. There are five MPEG standards being used or in development. Each compression standard was designed with a specific application and bit rate in mind, although MPEG compression scales well with increased bit rates.

JPEG stands for Joint Photographic Experts Group. It is also an ISO/IEC working group, but works to build standards for continuous tone image coding. JPEG is a lossy compression technique used for full-color or gray-scale images, by exploiting the fact that the human eye will not notice small color changes.

H.263 is based on H.261 with enhancements that improve video quality over modems.

DivX Compression uses the MPEG-4 standard to compress digital video, so it can be downloaded over a DSL/cable modem connection in a relatively short time with no reduced visual quality. DivX works on Windows 98, ME, 2000, CE, Mac and Linux.

Lossy compression - reduces a file by permanently eliminating certain redundant information, so that even when the file is uncompressed, only a part of the original information is still there. [VCompTut-1]

2.4 REAL-TIME MEDIA HANDLING

Time-based or real-time media are termed any data that change meaningfully with respect to time. Audio clips, MIDI sequences, movie clips, and animations are common forms of time-based media. A key characteristic of time-based media is that it requires timely delivery and processing. Once the flow of media data begins, there are strict timing deadlines that must be met, both in terms of receiving and presenting the data. For this reason, time based media is often referred to as *streaming media*: it is delivered in a steady stream that must be received and processed within a particular timeframe to produce acceptable results. [RTSP-01]

The format in which the media data is stored is referred to as its *content type:*

QuickTime, MPEG, and WAV are all examples of content types. A *media stream* is the media data obtained from a local file, acquired over the network, or captured from a camera or microphone. Media streams often contain multiple channels of data called *tracks*.

Common operations on streaming media are:

1. Capturing
2. Processing (ex. applying effects, transcoding, de/multiplexing)
3. Rendering (often also called playback)
4. Streaming

Capturing – Processing - Presenting

In this section we will see how to take advantage of **the Java Media Framework version 2.1.1** [JMF02-03] functionality to do capturing and processing. In the following section we will look at how JMF enables streaming media through the use the RTP protocol.

The JMF design goals are:

1. be easy to program
2. Support capturing media data
3. Enable development of media streaming and conferencing applications in Java.
4. Provide access to raw media data
   Enable the development of custom, downloadable demultiplexers, codecs, effects processors, multiplexers, and renderers (JMF *plug-ins*)
5. Maintain compatibility with JMF 1.0 Capturing is accomplished easily within the JMF. [Sun99]

The developer just queries the available capture devices (microphone for audio, camera for video) using the CaptureDeviceManager Object and then creates a DataSource object. A DataSource in JMF encapsulates both the location of media and the protocol and software used to deliver the media. Then a processor object to encode the captured audio and video signals - has to be constructed and initialized with the created DataSource object as its input source. A Processor Object in JMF is a specialized media player: it takes a DataSource as input, performs some user-defined processing on the media data and then outputs the processed media data either to a presentation device (headphones, display) or to a DataSource object (to allow streaming or further processing by another processor). Our delivery platform will use two custom developed Processors to implement video and audio signal compression.

1. **Encoding Processors** –one for audio and one for video - will be implemented to encode the data from the capturing DataSource - using the WaveVideo encoder and an audio encoding standard respectively- and to store the compressed data in another DataSource to allow subsequently RTP packetization and streaming. This Processor will be activated, whenever the participant becomes the actual sender in the session. [JMF02-04]
2. **Decoding Processors** –one for audio and one for video - will be implemented to decode real-time media delivered from the network (after RTP depacketization) - using the WaveVideo decoder and an audio decoding standard respectively- and send the uncompressed data to a presentation device. These Processors will be activated, whenever the participant is acting as a receiver. Audio-video stream synchronization is performed easily by using JMF's methods [JMF02-05].

2.5. REVIEW OF SIMILAR EXISTING APPROACHES AND SYSTEMS

2.5.1 Synchronous Teleteaching systems

2.5.1.1 IRI-h

IRI-h (Interactive Remote Instruction-heterogeneous, [SCC-01], [SCC-02]) is distance education system, which is currently being designed and implemented in an R&D project at the Old Dominion University in Norfolk, Virginia, USA. The system will work on a number of heterogeneous platforms and within heterogeneous network environments. Cross-platform nature is achieved through the use of the Java language. The implementation is successfully tested on multiple platforms including PCs running the Microsoft Windows operating system (NT, 98, 2000) and Unix machines running Solaris. The existing prototype is by now running over the university's LAN and its extension to cater heterogeneous network environments (ex. To support home users) is considered in future steps.

The main functional features of the current prototype are:

1. **Audio/video communication between participants.** It allows a maximum of three video windows to be present in the shared view at one instant of time.

The respective number for audio is ten.

2. **Application sharing:** every participant is capable of sharing every application running in his workstation with the rest of the session participants.

3. **Annotation tool shared among participants.**

They are both striving for cross-platform nature and network heterogeneity.

4. The Java Media Framework is used to capture, render and transmit time-based media (audio/video). At least we will start with JMF as an implementation framework, as long as its performance proves satisfactory.

However, discrete differences exist also between the two projects:

5. ET&L II will support the creation of sub-session within a main session. This will allow private communication between participants and thus provide tutoring and working in groups capabilities.

6. From the beginning of ET&L II and at the end of every implementation stage the test bed will consist of the targeted access technologies (LAN, cable networks and

ADSL as described later). We won't just work on a LAN until project completion and then implement intermediary components to achieve network heterogeneity.

In the past, we have established a good co-operation relationship with the developers of IRI-H. One of our collaborators visited the IRI-H team for an extended visit, and we plan to share knowledge and technology between the two projects.

2.5.2 Video conferencing systems

2.5.2.1 Microsoft Netmeeting

Microsoft Netmeeting ([39]) is a video conferencing tool, offering point-to-point communication with video/audio. Apart from this, it offers:

1. Whiteboard tool
2. Application Sharing
3. Remote Desktop Control
4. File transfer
5. Text-based chat

The main drawback of Microsoft Netmeeting is that it doesn't support multipoint audio/video communication. Its multipoint capabilities are constrained to the whiteboard tool and to application sharing. Therefore it can't be used as distance learning delivery platform. Moreover, because of its origin (Microsoft) it is not cross platform, but limited to the Windows operating system.

2.5.2.2 Click to Meet

Click to Meet ([40]) is a complete end-to-end solution for rich media communications. *Click to Meet* provides a framework for group communications using live, interactive voice, video and data collaboration as well as streaming technologies. *Click to Meet* supports T.120 standards for data sharing, so users can collaborate online using popular tools to chat, send files, draw on a whiteboard, and share applications. It is completely automated. Behind the scenes, the system securely connects users and seamlessly assures that all network resources will be available for the duration of each call. If a meeting is scheduled with a roam-based conferencing system, *Click to Meet* will automatically connect that endpoint at the appropriate time, allowing the users in that room to attend their meeting without ever touching a dial-pad or keyboard. With *Click to Meet*, you can support:

1. **The widest range of endpoints**, at varying bandwidth rates – whether you're using clients from First Virtual Communications or Microsoft, or endpoints from PictureTel, Polycom, as well as other H.323 vendors.
2. **Multiple server environments**, including Windows NT, Windows 2000, Sun Solaris, and Linux operating systems.
3. **Diverse applications**, from high-end corporate conferencing and application sharing to video instant messaging, both one-to-one and group conferencing.
4. **Flexible customization** that will fit your network topology.
5. **Easy scalability** as conferencing usage grows – by linking multiple conference servers together.
6. **User-friendly interface** through web-based graphic interface that provides easy-to use system administration and makes multipoint conferencing easier than ever.
7. **The latest advanced features**, including Microsoft Exchange compatibility for easy scheduling and meeting management, continuous presence for viewing multiple windows at the same time, and streaming media integration to extend a conference live to a larger, view-only audience; or to record a conference for playback later.

Drawbacks

1. Designed for and tested over the US backbone, mainly taking advantage of the existing ATM infrastructure. In the showcase of e-learning systems based on Click to Meet there are not any already installed systems mentioned, that use a cable or an ADSL network as their underlying network infrastructure.
2. Not explicitly aimed at education. It is purely a conferencing system, that doesn't provide the mechanisms to "simulate" the real classroom.
3. H.323 ([20]) is used by the application. This is a conferencing standard that covers most of the demands of a today's conferencing system. This also means that the system is not that modular and there are restrictions in the selection of codecs and session control protocol.

2.6 BENEFITS OF VIDEO CONFERENCING IN EDUCATION

Students meet with tutors for enrichment, remediation, or a helpful bit of personal attention. This is great way for business to support schools. A librarian offers an introduction to library services and library tour for local schools before they come to the library. Videoconferencing facilitates distributed cooperative learning, where groups at distant sites take on a learning task and teach remote peers. Distributed projects make use of videoconferencing technology for collaboration and communication. Students take classes not offered at their school, such as advanced honors, foreign language, or music courses. Schools and community colleges offer classes during off-hours and to students who cannot attend traditional classes. Community colleges team up with businesses to offer employee training or certification [GB-00].

Educators and librarians from around the country report that videoconferencing technology impacts student learning in the following ways:

1. Heightens Motivation

Video conferencing in the education environment has getting positive feedback from the some of educationist practitioners. Quoted as saying below some of their feedback on the implementation of Video Chat in the education industry:

According to Kayla Dove, Liberty Science Center, New Jersey, "The excitement of using new technology and interacting with other students or adults increases motivation." We have had students give up recess to do our programs and ask 'When are we doing that fun thing again?' "

Quoted as saying by James K. Tice, Supt. in Strafford, MO "Several of our students gave up lunch hour to continue the dialogue and viewing!!!"

2. Improves communication, presentation, and "SCANS" skills

Students perceive video chat as important and are more conscious of their appearance and oral communication. When the students plan and implement the videoconference, they learn important communication and management skills.

According to Paul Massmann, Concordia University Irvin "Students see themselves on screen and realize that is how others see them. Over the course of the semester I have seen dress change, posture change, and poise change, all for the positive."

3. **Increases Connection with the Outside World**

When a live class or meeting is not possible, videoconferencing makes a face-to-face visit is possible and an ongoing relationship can take place. Video chat is usually easier than setting up the live meeting so communication can be more frequent, saving time and resources. Students can have a greater opportunity to form meaningful relationships with their peers who may be different location from them. The richness of the communication supports the formation of relationships between learners and mentors/role models.

According to Bruce Betts from San Juan Capistrano Research Institute, "By removing the need for either the content provider or the students to travel, yet still providing a two-way audio AND video link, you're providing educational opportunities for interactions that would not otherwise exist." "In terms of cost effectiveness, a number of video conferencing systems could be placed into schools and other centers for the same cost of bringing all Highland Council Secondary Guidance Teachers to Inverness for an in-service day. " newspaper excerpt contributed by John Bruce, Highland Council, UK.

4. Increases Depth of Learning

Students learn to ask better questions. Video chat support learning from the primary source rather than from a textbook. Students show more depth understanding from this mode of learning as necessary planning contributes to a better learning experiences.

According to Beth Bustamante, SBC Education Advocate, "Videoconferencing lends itself to viewing multiple perspectives on an issue and it better addresses the needs of visual learners. Also, collaborative learning is practically automatic with videoconferencing. Videoconferencing helps set up authentic learning situations--students are working on a real-world problem or project and they are communicating with real people involved in the problem or project. This also supports the idea of authentic assessment--you must have your information pretty accurate before you connect with an "expert" and ask meaningful questions."

According to Kayla Dove, Liberty Science Center, New Jersey, "Because of the multiple camera set up we have here, there are times that off-site guests see things better then if they were here."

CHAPTER 3

METHODOLOGY

3.1 LIFE CYCYLE MODEL

For the project to be completed successfully, the method used is very important and draws many advantages. There are list of lifecycle models can be adopted, the student has chosen the basic Waterfall model as this model is very systematic and support the system development life cycle. There are 3 significant phases throughout the system development which are analysis phase, design phase and implementation phase.

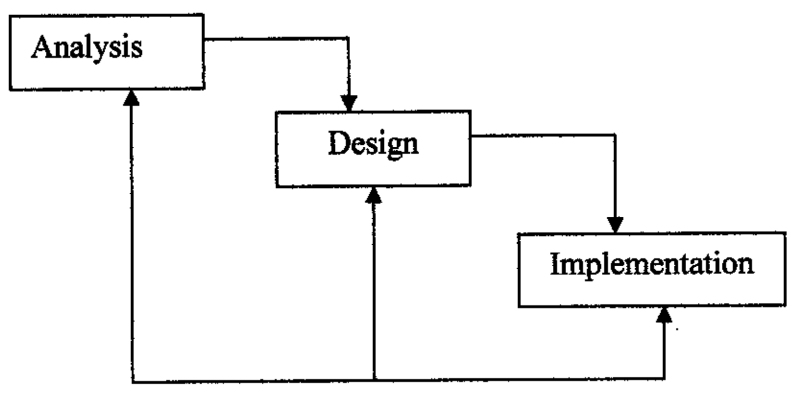


Figure 3.1 Waterfall Model

3.1.1 Analysis

Observation is the first stage, in which the student had try to do research and review of the similar existing approaches and systems of Video Conferencing as presenting in details in the Literature Review. The observation was useful in identifying problems and opportunities with respect to this messenger. Through the observation, the student gains some understanding on the video chat application in which author was able to have general ideas on the problems observed and the practical solutions that might be appropriate for it.

The student tries to seek any information in depth, of what is observed during the preliminary investigation, scope of study and tools to be used. Some informal discussions were held several times among peers and the author's supervisor and by reading all the publications that involves in the research work related to this application. All the available resources are utilized to get the essence of each and every elements of Video Conferencing application. The findings were very useful for the student to further refine the research to come out with the choices of solution that can be applied regards to this application.

3.1.2 Design

The third phase is one of the most important stages because it's the stage where all the data gathered is put into design. It's involves designing the navigational structure, distributed of the system and graphical user interface. The author has designed the application interface design. In order to complete the system functionality, it's important for the author to integrate the interface and the function button with the coding. As a result, the system can running well along with the function that the system is provided.

3.1.3 Implementation

The implementation stage will show all the activities such as the system development for the whole part including the interface and programming part. The testing will also will be done during the stage to ensure all the system components functioning well. There are unit testing, integration and validation.

3.2 Tools Required

3.2.1 Hardware and Software work together to make it function.

Table 3.1 Software used for Development Tools

Purpose	Development Tools	Description
Managerial	- Microsoft Words - Microsoft Project	- This Software is useful in doing the user manual and project report. - This software is used for scheduling and project activities.
System Development	- J2SDK1.4.2 (Java)	- This software allows the user to run the java program that saved and edited in the notepad. The user has to specify this folder directory inside the classpath and path in the environment variable.
	- JMF 2.1.1	- The Java Media Framework (JMF) is a recent API for Java dealing with real-time multimedia presentation and effects processing. JMF handles time-based media, media which changes with respect to time.
	- Adobe Photoshop 7.0	- This software is usable for image editing and graphic design.

Table 3.2 Special Hardware for Development Tools

Purpose	Hardware
Capture the Video	USB Web Camera
Capture the Audio	Microphone
Listen to the Audio	Speaker

3.2.2 Minimum Requirements for the System

- Windows 98/ME/2000/XP
- OS9.2.2 or higher / OS X or higher
- P200 MHZ (or equivalent) processor or higher
- 32MB of RAM (64MB recommended)
- Video card with 4MB of RAM for minimum 24-bit (High Color) display
- CD-ROM drive
- USB port

CHAPTER 4

RESULT AND DISCUSSION

4.1 INTRODUCTION

It's done! The author has successfully realizing the significant functions in her Video Conferencing application, thus answering the targeted objectives. It'll present and discuss the full specification of the system and its environment to name it a few features, connection, compression, bandwidth, few snap shot of the final product, and much more to offer in this section. Let's begin to understand the author's Video Conferencing!

4.2 VIDEO CONFERENCING DESIGN

4.2.1 Distributed Approach Design

For this Video Conferencing, the author has opted for the distributed approach. This solution is directed towards stable users and allows mobility for further expansion and promotes simplicity. There is no concrete server, as the first computer will act as a server (Peer-to-Peer).

It's an improvement over the centralized approach due to geographical barriers scattered all around the place. This server becomes the user's local server. Once the transfer is completed the profile on the local server is cleared. When other servers need an access to the user profile, they contact the user's local server if the user is online or the user's home server if the user is offline. The address of the user's home server is a part of his user handle. The distributed approach is illustrated in Figure 4.1 (Please refer to the next page).

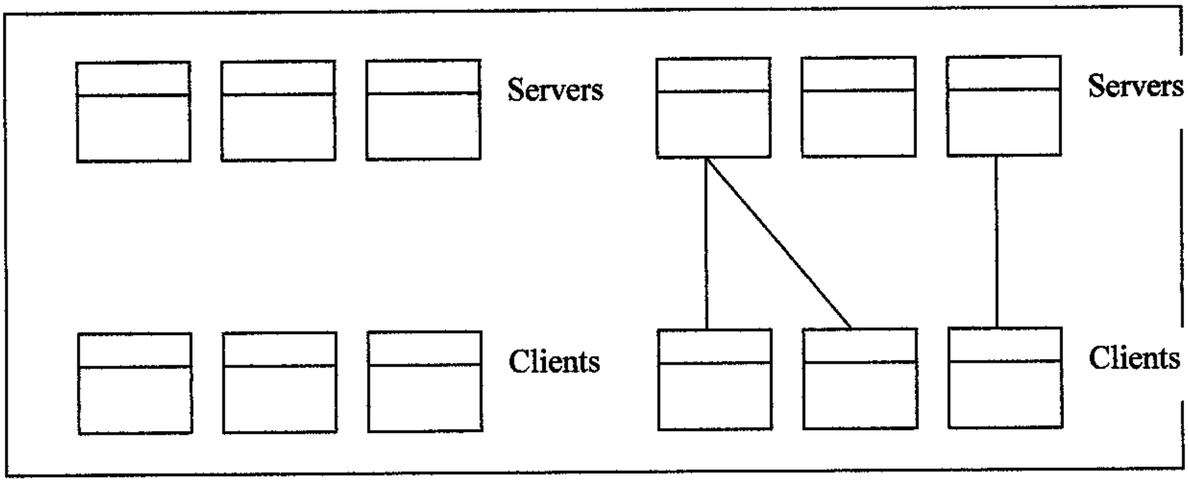


Figure 4.1: Servers and clients connection in a distributed approach

Why the author chose this design? Let's have a look at the advantages of this solution has to offer.

Pro:

i) Its scalability with respect to the number of users. The capacity of one server is limited but we can always add new servers when the existing ones become overloaded.

ii) No unnecessary redundancy of users. The users are only to be found at the user's client, the user's home server at his local server in the case of two different servers.

The pro of this design comes with its cons to be considered. Let see it together as listed 2 below.

Cons:

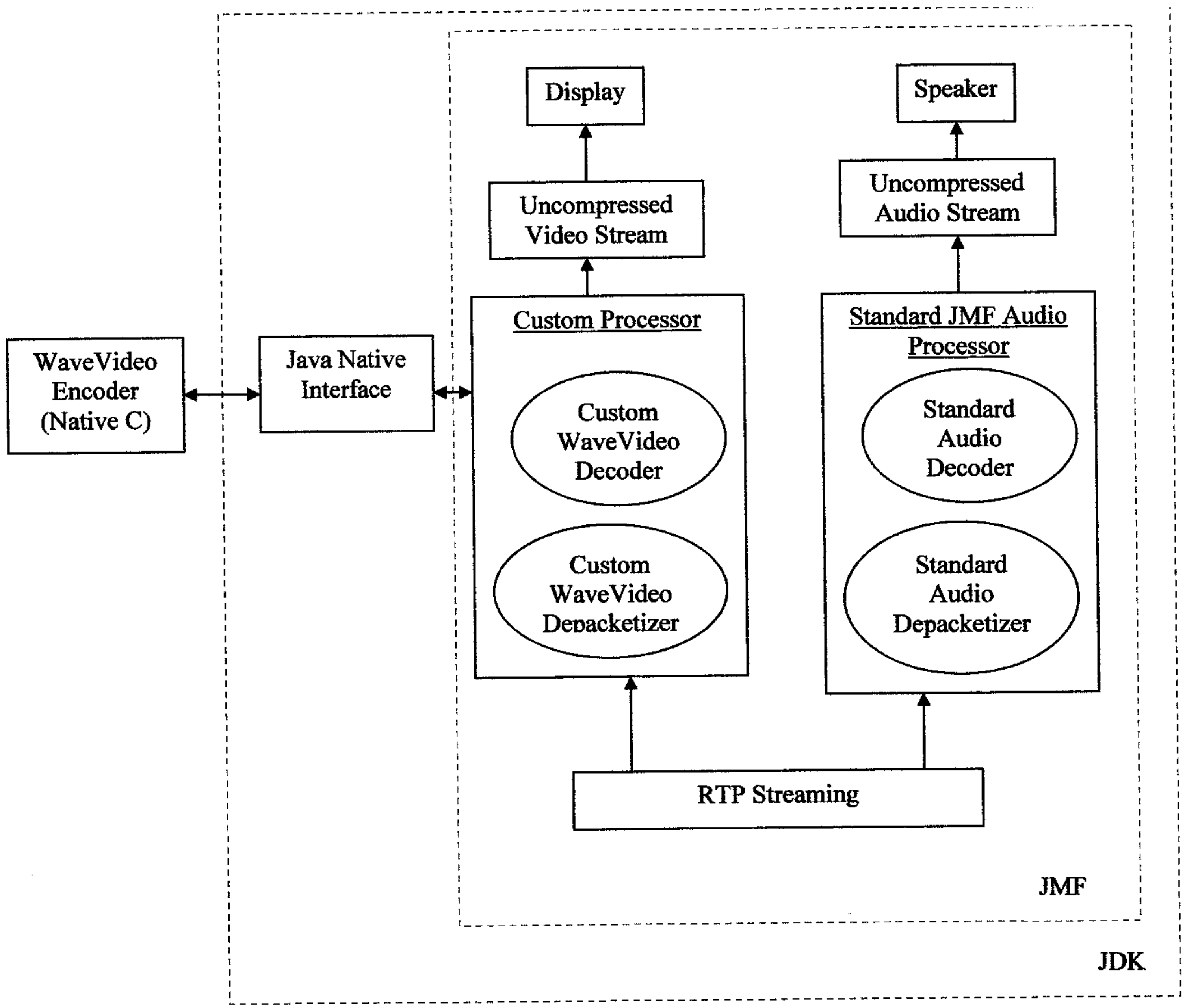
i) The distributed database works best when most of the users generally log into the same server. Highly mobile users constitute a problem since their user must often be sent to new servers and may need to travel through long communication paths.

ii) Another disadvantage is that, if one of the servers is down, the users for whom it is the home server cannot use the instant messaging system, since their user cannot be accessed.

4.2.2 Video Conferencing Architecture

Presenting to the reader two sketches of diagrams showing the complete handling process of real-time media, one is showing the handling on the sender's side and handling on the other end (receiver's side). There is some confusion on the term session, which is sometimes used for a conference and sometimes for a single media stream transported by RTP. In this section, the author use the term with both meanings, but in the RTP section (in the Literature Review) the author will use it only with its second meaning to resolve ambiguity. Please refer to the next two pages to see the real time media handling on the both sender and receiver side.

Please Refer to Figure 4.2 – Real – Time media handling on the sender's side
Please Refer to Figure 4.3 – Real – Time media handling on the receiver's side


WaveVideo Encoder (Native C)
Java Native Interface
Display
Uncompressed Video Stream
Custom Processor
Custom WaveVideo Decoder
Custom WaveVideo Depacketizer
Speaker
Uncompressed Audio Stream
Standard JMF Audio Processor
Standard Audio Decoder
Standard Audio Depacketizer
RTP Streaming
JMF
JDK

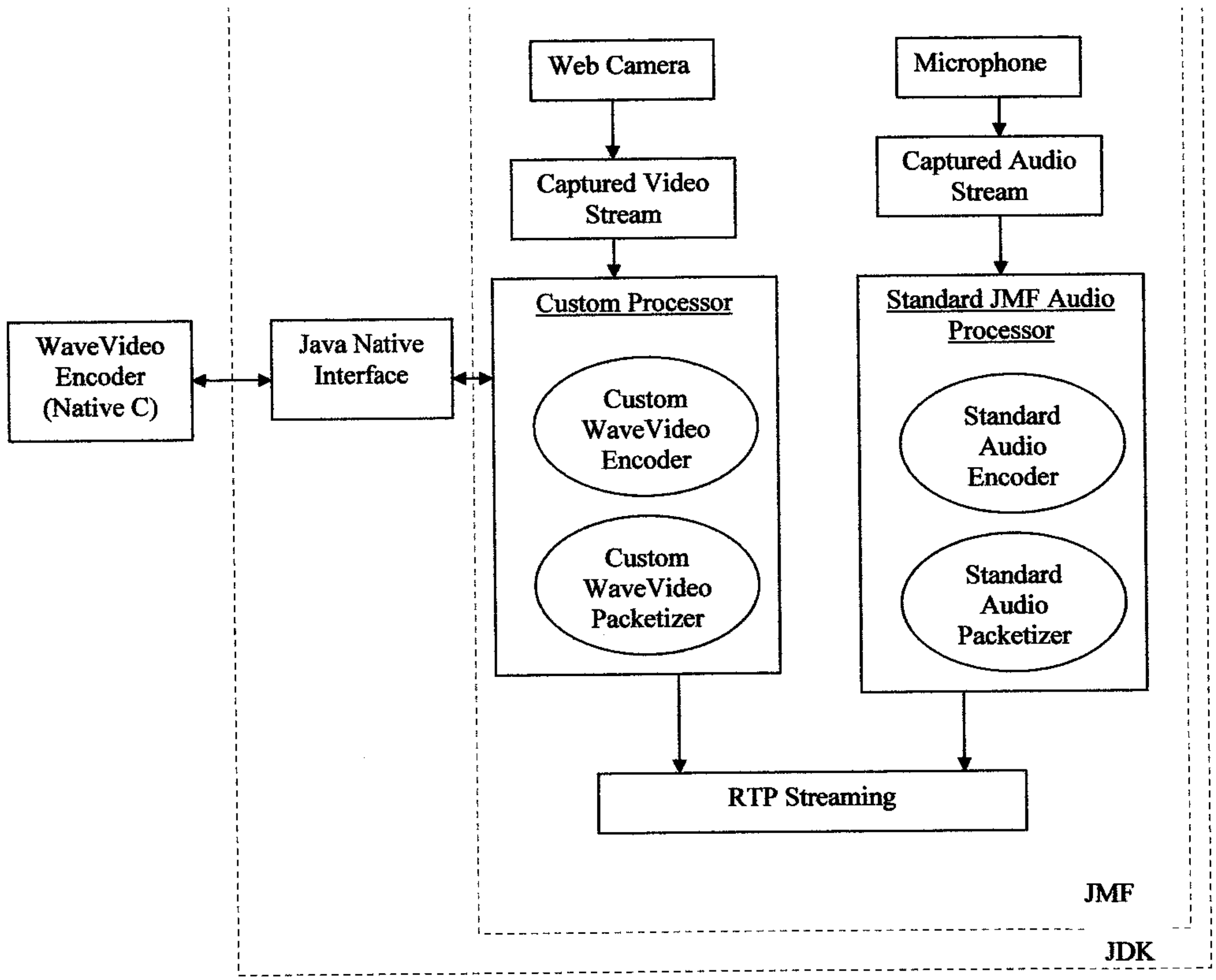


Figure 4.2 – Real – Time media handling on the sender's side

4.3 APPLICATION REQUIREMENTS

In implementing a video chat application that deals with the multimedia elements using JMF embedded in the RTP network transport layer, certain critical constraints of real-time traffic need to be considered:

4.3.1 Compression

Video streams can consume a considerable amount of bandwidth. Thus, compression is important to make it widely practical. Codecs based on G.723 for audio and H.263 for image compression is often used, particularly for low-end products. The standard Video Conferencing products use 384 and 768 kb/s to produce attractive quality. Compression technique is used to achieve a compromise between the quality and bandwidth consumption. Increased compression can result in more streams going through a fixed capacity; however, more "headroom" is needed to minimize packet loss and jitter that will occur when there are bursts in competing network traffic.

4.3.2 Audio and Video delays

Audio delays can occur because it takes a little longer for information to compress, travel, and decompress. Videoconferencing novices usually experience a few awkward conversations due to this time delay. Since there's no way to prevent the delay, learn to finish thoughts in a single statement with an obvious conclusion. Listeners should avoid interrupting and use visual cues (like nodding) instead of verbal affirmations (like "uh huh"). And it takes a little while for the video call to "hang-up." It's not because one of the chatter pressed the button to end the call doesn't mean that the other end has instantly disconnected. Be careful what to say until the call has completely disconnected.

For audio, inter-packet delays in the order of 200 ms are the maximum tolerable for maintaining a conversation, whereas video has even more complex requirements.

Both of video and audio requires compression and decompression technique to improve the effectiveness and efficiency in transmitting over the network. Few options for audio compression techniques are DVI/RTP, G723/RTP, GSM/RTP, ULAW/RTP, MPEGAUDIO / RTP. All are embedded with the RTP extension. The options for how the video looks like are RGB and YUW.

4.3.3 Audio "clipping" or echo

Audio "clipping" or echo might take place when audio system isn't properly configured. If the chatter experiences audio problems, reset the echo canceller and reduce background noise. The chatter checks your equipment documentation for volume and microphone placement guidelines. Depending on equipment, use of headsets and external speakers can also improve audio quality.

4.3.4 Video Chat versus 3G Technology

As the emergence of Video Call known as 3G, people tends to compare Video Conferencing with 3G technology. Undoubtedly, both deal with video and audio transmission, but one thing that makes Video Chat significant in students' population is when come to budget and pocket money. Video Conferencing is saving. They don't have to pay for the very expensive hand phone and paying a very high monthly service.

4.3.5 Video "ghosting" or "pixilation"

Video "ghosting" or "pixilation" is the codec's way of compensating for rapid information flow. One way the codec compacts information is by reducing frame rate (number of video images per second), which can make rapid motions appear jerky. The codec also drops resolution to compress information, as in the photo shown, which can make an image fuzzy or chunky. To reduce these effects, reduce the amount of change. Avoid rapid motion, wear plain clothing, and hang a pastel curtain behind participants to reduce extraneous visual information.

4.3.6 Bandwidth

Generally bandwidth is the amount of information that can be transmitted over an information channel. High bandwidth Internet access means those audio, video, and graphics load quickly. High bandwidth videoconferencing means picture and sound will be clear. Compression techniques will reduce the media size up to 70% and hence the media will go faster across the network channel. Thus low cost of bandwidth is achieved and is utilized.

In order to see how this application going to affect the network performance, few computers in the network are taken as the sample. Bandwidth will be calculated to see how all the computers performance will be affected when they are many flows of sending and receiving activities (Objective Evaluation). Please Refer to the **4.5.34Sample of Objective Evaluation in Measuring the Image Quality**. The performance of the sending and receiving the media elements can be analyzed critically by observing the quality of video and audio being transmitted (Subjective Evaluation). Please Refer to the **4.5.3 Sample of Subjective Evaluation in Measuring the Video Conferencing.**

4.3.6 Delays

To comply with the timing needs described above, end-to-end network delays (including processing time, transmission latencies and queuing) and delay variation (jitter) need to be deterministic or, at least, accurately modeled. Traffic not conforming to this rule is equivalent to lose and, as such, will be discarded by the application. In this case, by using JMF-RTP, the jitter has successfully reduced both the audio and video.

4.3.7 Distribution

As this java video chat caters in the distributed environment around the campus and since variations in delivery path can have significant impact on the delays and the loss characteristics imposed by the network. It's because of the resource management, bandwidth restrictions, scalability considerations, and quality of service requirement, efficient delivery mechanisms need to be employed in order to make sure the media elements at their utmost interest.

4.3.8 Working with Technological Constraints

Transmission of compressed video is done via a smaller "pipe" than a televised broadcast. The camera and microphone take in more information than the "pipe" can handle. Outgoing video and audio information must be processed by a piece of equipment called the codec (**co**der-**dec**oder) before it can be transmitted. Incoming signals are then decoded before they are sent to the monitor and speakers. All this processing takes its toll on the resulting picture and sound.

4.3.9 Videoconferencing etiquette must be established immediately by the users.

Most people have not experienced videoconferencing and do not communicate as they would in a face-to-face situation. For example, in the middle of a carefully prepared demonstration, a remote viewer breaks into conversation, interrupting and ignoring the designer. One of the best examples of "video rudeness" we've ever heard of involved a meeting in which one of the remote participants did not want to attend. They focused the camera on the floor and left the room without saying anything.

Shuffling papers or dangly bracelets in constant motion near the microphone, entering or leaving during a meeting, even serving food and drinks can be extremely disruptive. By muting and switching camera positions, the chatters mask these activities and eliminate interruptions for the remote sites.

4.3.11 Two-way videoconferencing is unlike one-way television.

Many people have a difficult time changing habits and preconceptions produced by years of experience with television. Not only do people tend to "tune out" what's on a television screen; we also expect to be entertained by it. People expect broadcast quality video, slick graphics, and a quick pace to keep us engaged. And if people are not fascinated, the people quickly change channels to something more interesting. Teachers who use two-way video must challenge basic learner preconceptions and set new expectations to maximize learning. Fortunately, good classroom instructional strategies are also good two-way video instructional strategies.

4.4 Video Conferencing JMF Architecture

This section provides a summary of how JMF is used by the author's product of Video Conferencing to capture video from a web cam, display live video to a user, transmit live video across a network, and store live video to disk.

Please See Figure 4.4 – Video Conferencing Processor: Video from a web cam is captured and cloned for viewing, streaming and storing to file.

Figure 4.4 shows how the Video Conferencing Processor application clones the DataSource that is associated with the web cam. A Player is used to view the live feed from the web cam, and separate Processors are used to transmit the video across the network and record to file. The QuickTime format is used for storing video to file. Figure 4.4 shows how a Video Conferencing Controller application displays streaming video from a Cam Processor or Media Server to the user. The streaming video is accessed using an RTP SessionManager. When a video stream becomes available, the SessionManager creates an RTP-encoded DataSource that is fed into a Player. The Player allows the Cam Controller user to see the live video stream. If the streaming video is halted because the Cam Processor is brought down or the video file has finished being streamed from the Media Server, the SessionManager is alerted and stops the Player. A SessionManager would not be necessary if the Media Server notified the Video Conferencing Controller via callbacks when the video streaming had finished. In the case of the Video Conferencing Processor, the SessionManager is not necessary at all since the Media Server already notifies all Cam Controllers when a Video Conferencing Processor is no longer available. Figure 4.5 shows this process in detail.

Please See Figure 4.5 – Video Conferencing Controller: Streaming RTP video is sent to a Player for viewing.

Figure 4.6 shows how the Media Server application transmits archived video over the network to Video Conferencing Controllers. QuickTime video files are first converted to MPEG/RTP encoding. Then it is transmitted over the network using a DataSink. Once the video has finished streaming, the DataSink and Processor are closed and removed. Closing of the RTP channel signals the receiving Video Conferencing Controller that transmission of the archived media is complete.

Please Refer to Figure 4.6 – Media Server: A QuickTime file is streamed with RTP over the network.

JMF uses a system of MediaLocators, SessionManagers, Players, Processors, DataSources, and DataSinks for coordinating the capturing, transmission, and reception of time-based media as presenting in the Table 4.1 below. These objects hide the underlying complexity associated with stream-based applications.

Table 4.1 Objects inside JMF with its Functions

Objects Inside JMF	Funtions
MediaLocator	Identifier for a media DataSource, similar to a URL
SessionManager	Coordinates RTP sessions by keeping track of the session participants and the streams that are being transmitted within the session
Player	Processes an input stream of media data from a DataSource and renders it at a precise time. It provides standard user controls like play, pause, etc.
Processor	Specialized type of Player that provides greater processing control on an input media stream. It can output media to a media presentation device or to a DataSource.
DataSource	Delivers the input media-stream to a Player or Processor.
DataSink	Reads media data from a DataSource and renders the media to some destination (file, network, RTP broadcaster, etc.).

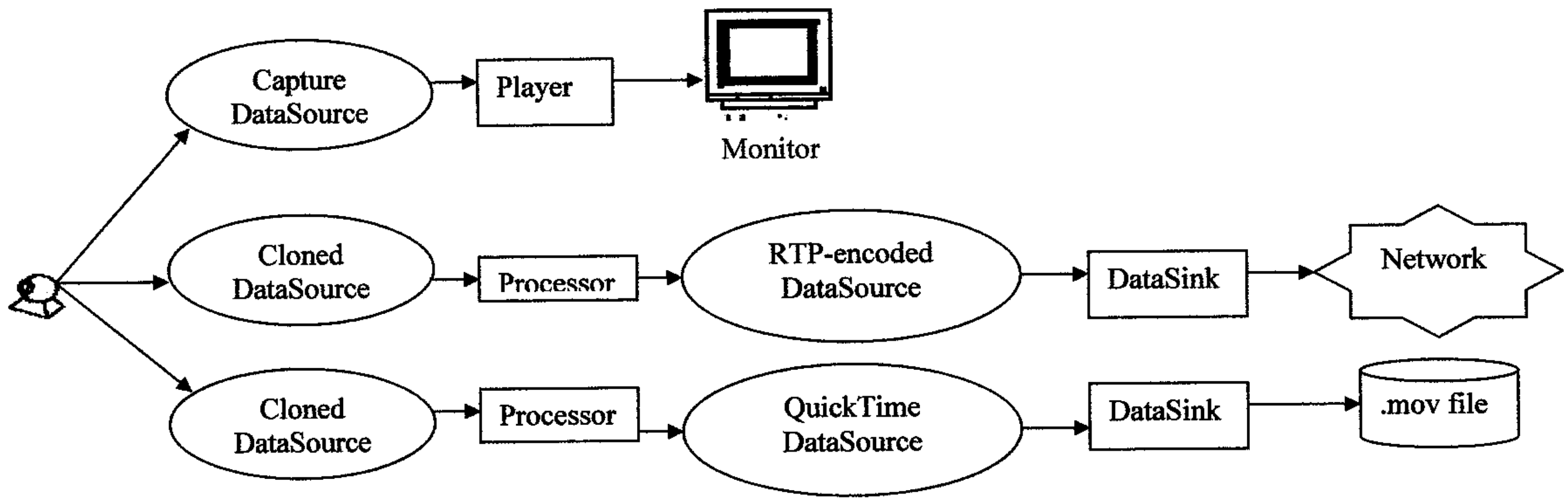


Figure 4.4 – Video Conferencing Processor: Video from a web cam is captured and cloned for viewing, streaming and storing to file.

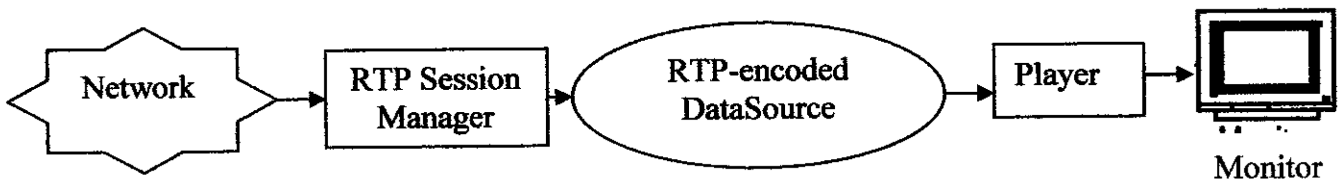


Figure 4.5 – Video Conferencing Controller: Streaming RTP video is sent to a Player for viewing.

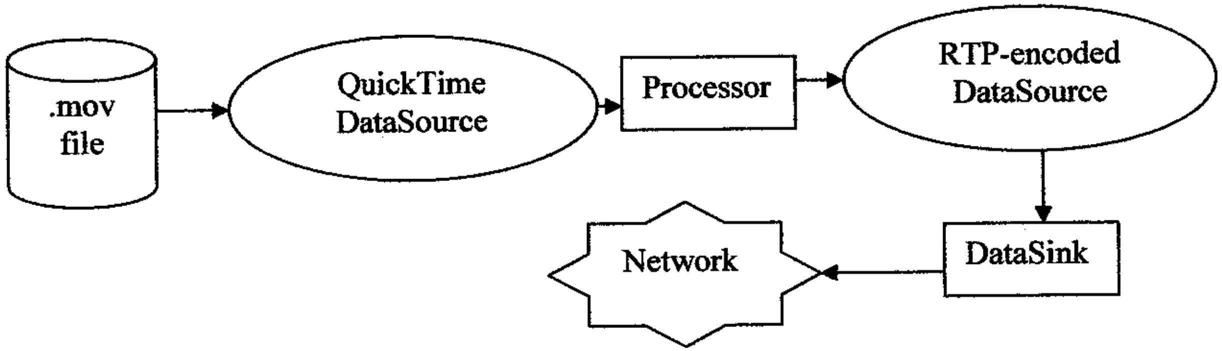


Figure 4.6 – Media Server: A QuickTime file is streamed with RTP over the network.

A MediaLocator object is used to access video for a web cam on a Windows computer like so:

MediaLocator mediaLocator = new MediaLocator("vfw://0");

Once the video feed is obtained, a DataSource is created to store the live video feed.

DataSource dataSource = Manager.createDataSource(mediaLocator);

This DataSource can be fed to a Player for the user to view the live video, or it can be sent to a Processor that transforms the DataSource into a different type of DataSource. The code segment below shows the web cam video feed being converted into an RTP-encoded DataSource for later transmission over the network.

Processor processor = Manager.createProcessor(dataSource);

// Transform into RTP DataSource (code is omitted for brevity)

DataSource rtpDataSource= processor.getDataOutput();

A converted DataSource can be stored to disk or transmitted across the network with the use of a DataSink.

The following example shows how a DataSink would be used to transmit an RTP-encoded DataSource to the foobar computer on port 5050.

MediaLocator rtpLocator = new

MediaLocator("rtp://foobar:5050/video");

DataSink rtpTransmitter = Manager.createDataSink(rtpDataSource,

```
rtpLocator );
rtpTransmitter.open();
rtpTransmitter.start();
```

If the web cam's DataSource is to be transmitted over a network and recorded locally to file at the same time, it must be created as a cloneable DataSource. A cloneable DataSource can be cloned any number of times for different uses. The following shows how a cloneable DataSource is created from the web cams DataSource:

```
DataSource cloneableDs =
Manager.createCloneableDataSource(dataSource);
```

The JMF objects discussed in this section allow a Java application easy access to time-based media.

4.5 DISCUSSION

4.5.1 Video Conferencing Features

1. Multi-platform application that is grounded on Java technology.
2. Easily find the connection to the main server due to connection of socket between two computers.
3. Able to transmit the real time and good quality video and audio to the remote pc / over the network using JMF and JMF-RTP.

4. Able to broadcast a live feed to your company's intranet. The source of the media can be a file, live media using a capture device or any other source supported by JMF.
5. Transmit the real time and good quality video and audio to the remote computers or over the network by specifying the subnet.
6. Broadcast a live feed to the network computers. The source of the media can be a file, live media using a capture device. By specifying the subnet to Subnet Mask: 255.255.255.0
7. Multiple users, each capable of broadcasting to and receiving feeds from many users
8. Low cost solution for continuous video/audio feed

4.5.2 Video Conferencing System

Video Conferencing gives the users special connection with typed words and at the same time enables them to lively see and hear the person they are talking to. Video Conferencing connects people using words and then establish live talk.

To establish the connection between people, drag the cursor and click **Connect**. It'll prompt with an input box asking for the next user IP address they wish to connect to as shown in Figure 4.4. As the IP address is entered, and click OK and thus the connection is established and "Welcome" decked the top line of the Text Area. The people can start to chat between each other.

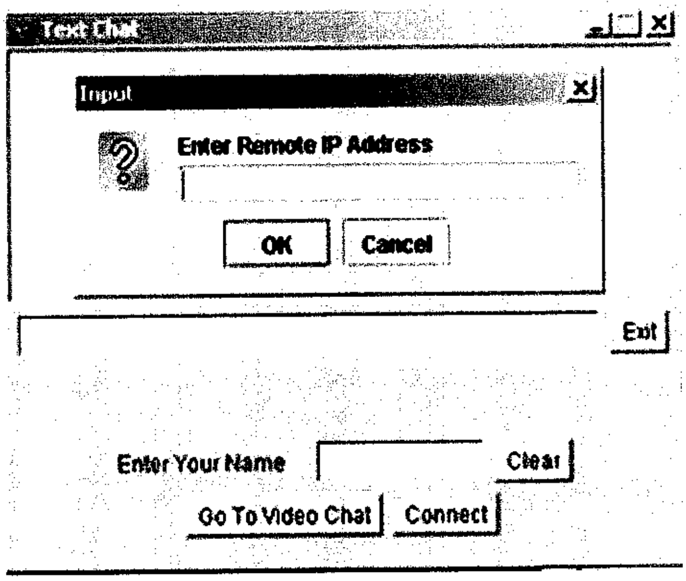


Figure 4.7 Text Chat

The people will come to the world of live chat by give a shot at the **Go To Video Chat** button. The world of video conferencing starts. Not only live chat as shown in **Figure 4.8** Drop down Menu, they would be able to send any supported of wide arrays of media types, including

- **protocols:** FILE, HTTP, FTP, RTP
- **audio:** AIFF, AU, AVI, GSM, MIDI, MP2, MP3*, QT, RMF, WAV
- **video:** AVI, MPEG-1, QT, H.261, H.263
- **other:** HotMedia

*MP3 is supported only on the Windows platform.

They would be able to set the player to be Auto Play, Auto Loop and Snap and Save the Shot file as shown in **Figure 4.9**.

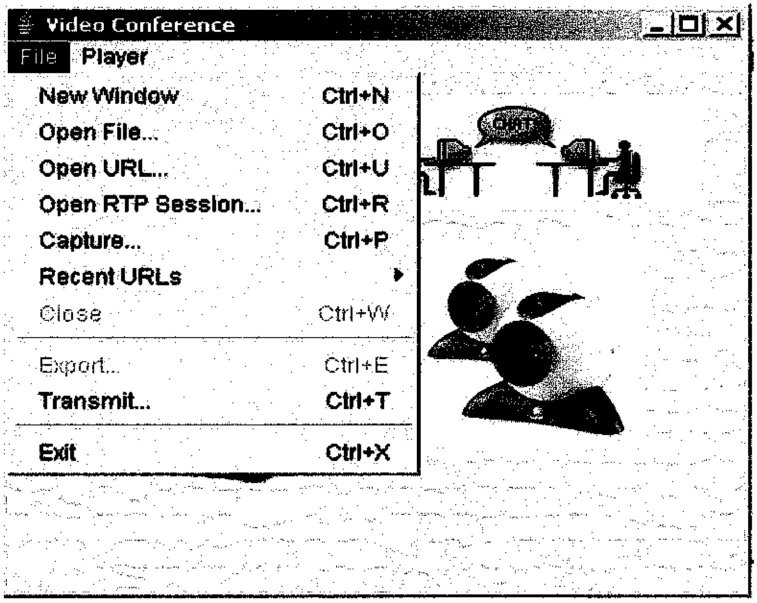


Figure 4.8 Video Conference Drop down Menus

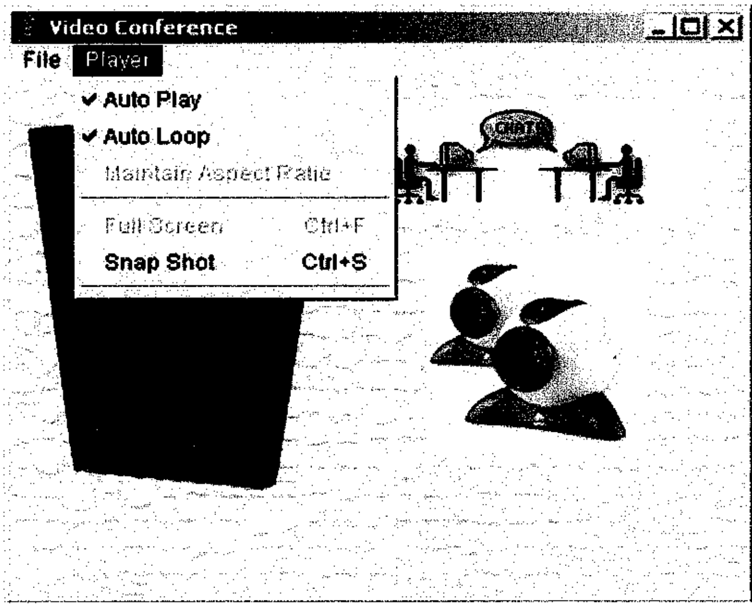


Figure 4.9 Video Conferencing File Player

Video Conferencing uses RTP to transmit the file and captured video across the network. As being discussed previously about RTP and video and audio compression techniques, the users can choose either to send it both or having options to send it either one of them. The user would be able to specify the Video Size, Frame Rate for video and Sample Rate, bits per sample and channel for audio as shown in Figure 4.8. To enable real-time streaming, dedicated streaming media servers and streaming protocols, which are in this case utilizing the Real Time Protocol (RTP), are required. RTP is an internet standard protocol for transporting real-time multimedia data. It's uses unreliable UDP protocol to transmit the packets.

JMF does provide the basic building blocks for a conferencing application to transmit and receive media over RTP. It is possible to build fully- featured standard-based conferencing solutions on JMF if the author has combined it with conferencing protocols like H.263. As for low-bandwidth video, the author has used H.263. For audio, GSM, G.723, DVI is good choices. Refer to Figure 4.10.

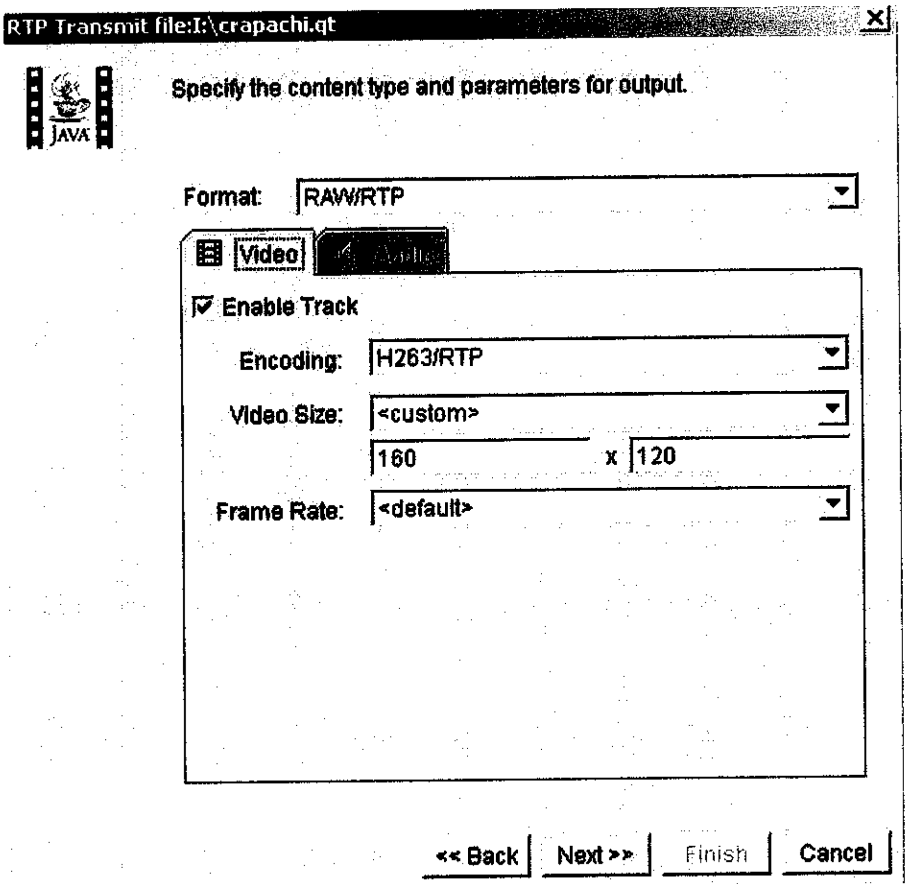


Figure 4.10 Transmit File and Media

The user would be able to select the capture devices which are both from the video and audio device. The user needs to specify all the attributes, which are consists of the encoding techniques for both of audio and video, Video Size, Frame Rate, Sample Rate, Bits per Sample and Channels as shown in Figure 4.11. While transmitting, the user would be able to monitor the performance of the video and audio by check to monitor the audio and video box. The user at the sender site can see the captured video and recorded audio. The user at the sender site can see the captured video and recorded audio. Standard web camera is used for the video device, whereas the speaker is user for audio device.

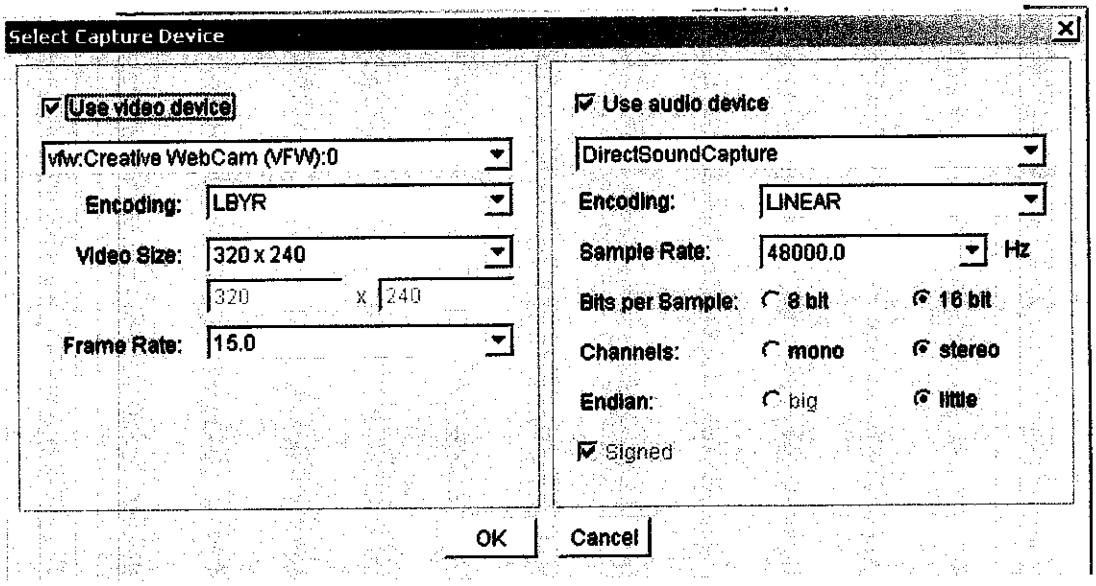


Figure 4.11 Select Capture Device both Video and Audio Devices

4.5.3 Sample of Subjective Evaluation in Measuring the Video Conferencing

While transmitting the captured video and audio to the remote computer, the sender would be able to monitor the video and audio that being transmitted as shown in Figure 4.12 Video Monitor and Figure 4.13 Audio Monitor. The Subjective Evaluation can be done here. In order to see how this application going to affect the network performance, few computers in the network are taken as the sample. Based on the author observation on the quality of the video and amount of tolerable delays and jitters is determined. The author has opted for the very good compression technique that reduces up to 70% of the video size. Video size is even big in the storage. Again compression will cause of losing some of the information and degradation of the quality. But bear in mind, as long as the degradation is tolerable and the image can be distinguished after being sending to the remote computer, the objective is achieved.

For audio, inter-packet delays in the order of 200 ms are the maximum tolerable for maintaining a conversation. Video has even more complex requirements. Based on the observation no significant delays is observed because there is no such very real time application that has ever built in this world. Undeniable there is slightly delays of 0.1 sec. As the receiver would be able to maintain an ongoing conversation thus the objective is achieved.

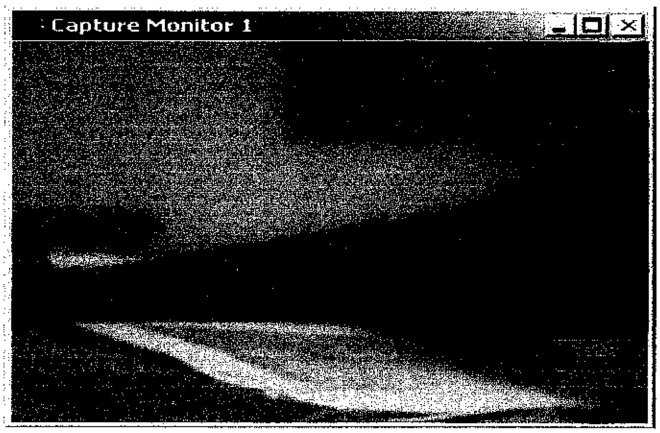


Figure 4.12 Capture Monitor

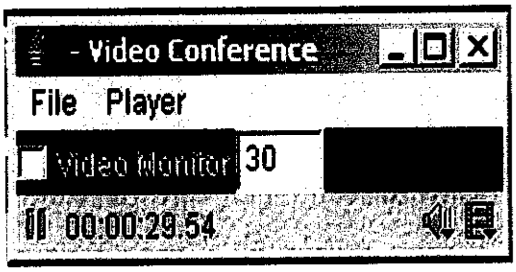


Figure 4.13 Audio Monitor

4.5.4 Sample of Objective Evaluation in Measuring the Image Quality

Comparing restoration results requires a measure of image quality. Two commonly used measures are *Mean-Squared Error* and *Peak Signal-to-Noise Ratio.*

The mean-squared error (MSE) between two images $g(x,y)$ and $\hat{g}(x, y)$ is:

$$e_{MSE} = \frac{1}{MN} \sum_{n=1}^{M} \sum_{m=1}^{N} [\hat{g}(n, m) - g(n, m)]^2 \qquad (1.25)$$

One problem with mean-squared error is that it depends strongly on the image intensity scaling. A mean-squared error of 100.0 for an 8-bit image (with pixel values in the range 0-255) looks dreadful; but a MSE of 100.0 for a 10-bit image (pixel values in [0,1023]) is barely noticeable.

Peak Signal-to-Noise Ratio (PSNR) avoids this problem by scaling the MSE according to the image range:

$$\textbf{PSNR} = -10 log_{10} \frac{e_{MSE}}{S^2} \qquad (1.26)$$

where S is the maximum pixel value. PSNR is measured in decibels (dB). The PSNR measure is also not ideal, but is in common use. Its main failing is that the signal strength is estimated as S^2, rather than the actual signal strength for the image. PSNR is a good measure for comparing restoration results for the same image, but between-image comparisons of PSNR are meaningless. One image with 20 dB PSNR may look much better than another image with 30 dB PSNR.

MSE and PSNR figures provided in this thesis were calculated after quantization (i.e. after converting floating-point pixel values to integer), but before clipping of the intensity range.

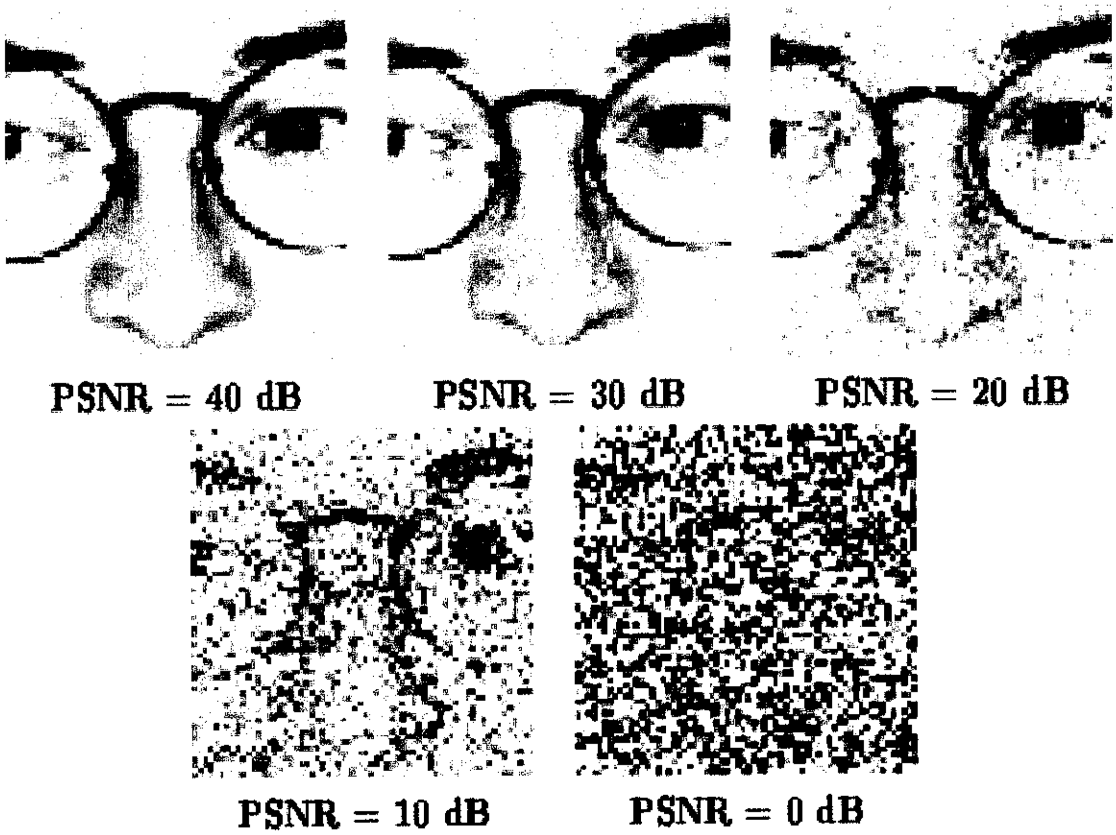


Figure 4.14: Illustration of the PSNR measure

Jitter Calculation

The Jitter J is computed based on the formula – $J = J + (|D(i\text{-}1,\ i)| - J)/16$, where $D(I$ $1,\ i)$ corresponds to the difference between the delay for ith RTP packet and the delay for the (i-1)th RTP packet. Based on one of the researcher calculation on it, undeniable JMF-RTP is the most efficient in reducing the jitters and delays up to 0.8 milliseconds. Very small! It'll look like real video and very smooth. Significant achievement of JMF-RTP compared to the others protocol. Please refer to the table below.

	Mean Delay (milliseconds)	Standard Deviation (milliseconds)
JMF-RTP	0.898	0.494
JMS-RTP	3.282	0.877

Table 4.2 Comparing delays between JMF-RTP and JMS-RTP

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

The Video Conferencing system was built using Java and JMF to transmit the real time video and audio between computers attached to the network with JMF-enabled web cam. Other developers have experienced similar problems with JMF [JMF02-1]. Sun is currently in the process of refining and extending JMF, and future versions are likely to be more stable [JMF02-2]. Sun also provides the complete JMF source code for developers who need to fix bugs, extend functionality, or remove bottlenecks for use in their application.

The quality of videoconferencing is getting better and better all the time. Video compression algorithms rely on the assumption that there are usually only small changes from one "frame" to the next. Video Conferencing is bandwidth hungry especially if there are multiple video streams being sent on a network. The project is intended in a way to keep the bandwidth used to a minimum to achieve its target to serve enough people and with its sufficient interconnectedness.

5.2 RECOMMENDATION

Just like the advent of Internet 10 years ago, we expect the use of video chat to expand and to proliferate relentlessly in the next decade. This will be encouraged through the growth of widespread access to the imminent arrival of Internet -III, with ten time's greater bandwidth and cheap off-the-shelf audio/video devices. As Technology student, I should prepare myself for taking these momentous steps, especially in view of the growth of borderless world. As a ground floor technology, video chat will undoubtedly assume greater importance in the future. I can see a growing trend that the web-cam will migrate from the PC, as PDA's are becoming capable clients for enabling a web-cam. Overseas in countries like Japan and Finland, mobile phone services already exist that allow using cameras for video streaming. 3G wireless technologies for cell phones will employ data transfer rates of over 300kbps, while transferring audio and video from a small wireless device. This is the direction that the web-cam trend will take in the future, as wireless protocols become fast enough to support the demands of streaming quality video, which is vital in the Earth sciences. Such a video-streaming capability will be useful in broadcasting simulations. This technology assists distant learning and education, where an instructor can navigate through the jungle of complex data for explaining key concepts to the students.

5.3 FUTURE ENHANCEMENT

5.3.1 UTP TV during the weekend ("Movie in the Room")

Java Video chat can transmit the real time and good quality video and audio to the remote computers or over the network by specifying to the Subnet Mask 255.255.255.0. Every computer that attach to the network can view the broadcasted video or in this case they can view live movie broadcasted from the server computer. The source can be varies. This Java Video Chat promotes low cost solution for continuous video and audio feed that's very convenient for the students' population.
This UTP TV can encourage the students to be in the hostel during the weekend as they don't have to go to the cinema to view the movie. This project definitely will be getting positive response from the parents and UTP.

5.3.2 Video Chat – Phone call

For the time being, Video Chat can only be connected between two computers. It would be very good if the user can contact another user's hand phone or fixed line phone from their computer. In United States, they already were having this system but only for the audio. With this new enhancement, both of video and audio can be transmitted.

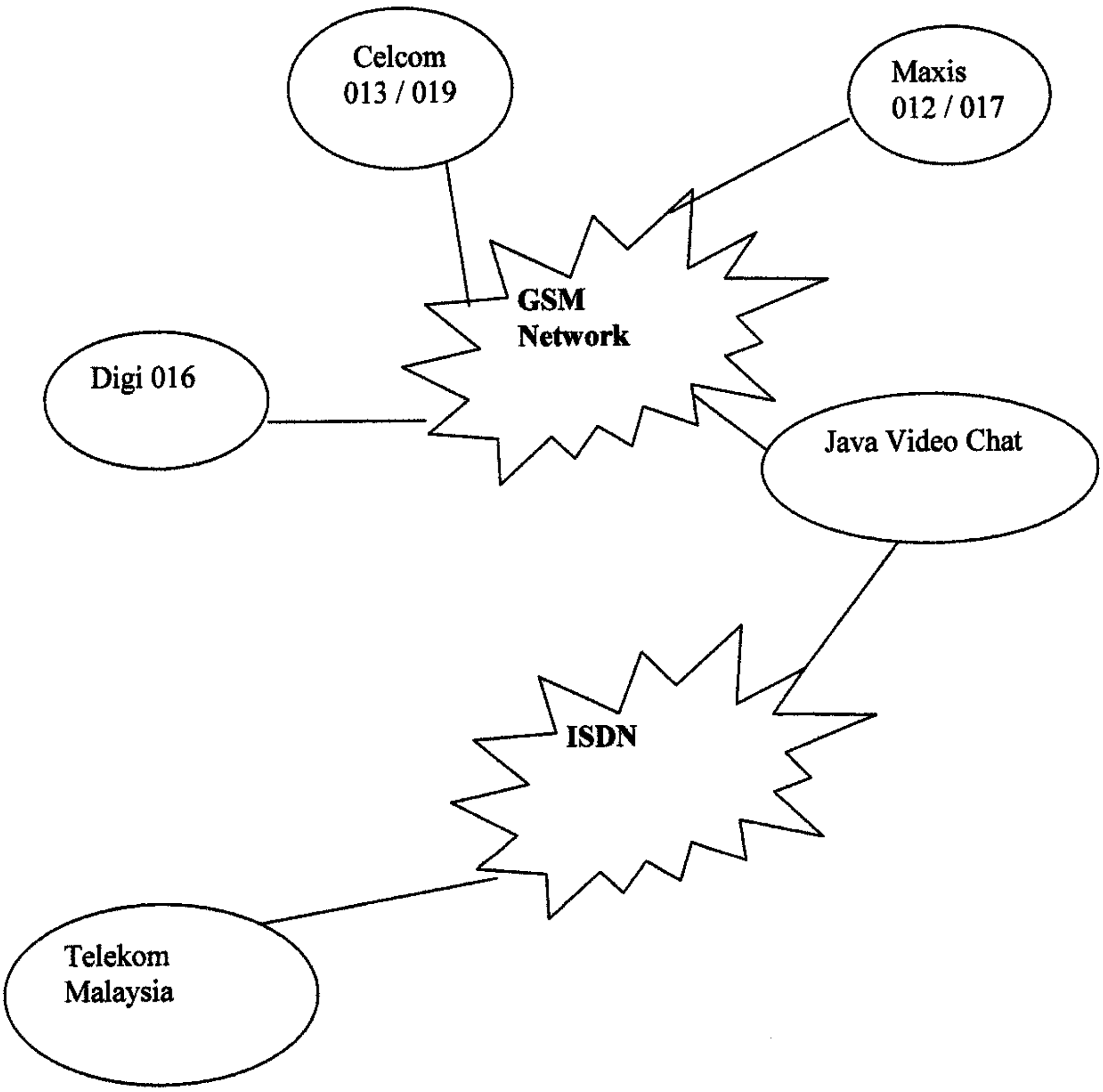


Figure 5.1 Computers to Phone Architecture

REFERENCES

1. [DBS-VC02] Dr Ben Shneiderman, 2002, "Video Conferencing", United Kingdom
<http://www.staffs.ac.uk/COSE>

2. [GB-00] Greg Buczeck, 2000, "Benefits of Video Conferencing in Education" journal, United States <http://www.kn.pacbell.com/wired/vidconf/intro.html>

3. [JD-VS03] John Deeney, 2003, "Video Services", Irwin McGraw-Hill <http://www.its.uiowa.edu/tns/videoservices/confwhat.htm/>

4. [JMF02-1] JMF interest mailing list, June 2002,
<http://swjscmail1.java.sun.com/cgibin/wa?A2=ind0206&L=j mfinterest&D=0&P=2733>.

5. [JMF02-2] JMF interest mailing list, June 2002,
<http://swjscmail1.java.sun.com/cgibin/wa?A2=ind0107&Lmfinterest&P=R34476&>

6. [JMF02-03] Java Media Framework 2.0 Programmer's Guide, November 19, 1999, Sun Microsystems
(http://java.sun.com/products/javamedia/jmf/2.1.1/guide/JMFTOC.html)

7. [JMF02-04] Java Media Framework 2.0 API Specification, March 19, 2001, Sun Microsystems
(http://java.sun.com/products/java-media/jmf/2.1.1/apidocs/)

8. [JMF02-05] Linden De Garmo: "Core Java Media Framework", Prentice Hall, 1999

9. [Mic-LM02] Microsoft Co., Copyright © 2004, "Live Meeting", United States
<http://www.microsoft.com/office/livemeeting/prodinfo/scenarios/marketing.mspx>

10. [Mic02-2] Microsoft, Streaming Methods: Web Server vs. Streaming Media Server, June 2002, <http://www.microsoft.com/Windows/windowsmedia/compare/ webservvstreamserv.asp>.

11. [RTSP-01] H. Schulzrinne, A. Rao, R. Lanphier: "Real-Time Streaming Protocol (RTSP)", IETF Network Working Group, Request For Comments 2326, April 1998
[RTP-02] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson: "RTP: A Transport Protocol for Real-Time Applications", IETF Audio-Video Transport Working Group, Request for Comments 1889, January 1996

12. [SCF96] Schulzrinne, H., S. Casner, R. Frederick, and V. Jacobson, "RFC 1889: RTP: A Transport Protocol for Real-Time Applications," January 1996, June 2002, <http://www.ietf.org/rfc/rfc1889.txt?number=1889>.

13. [Sun99] Sun Microsystems, JMF 2.0 API Guide, "Working with Real-Time Media Streams,"November 1999, June 2002, <http://java.sun.com/products/java-media/jmf/2.1.1/guide/RTPRealTime.html>.

14. [VCompTut-1] http://www.wave-report.com/tutorials/VC.htm, "Video Compression Tutorial"

15. [Wil02] J. R. Wilcox, Videoconferencing and Interactive Media: the Whole Picture, Telecom Books, 2000, p. 169.

16. [SCC-01] A. Abdel-Hamid, S. Ghanem, K. Maly and H. Abdel-Wahab: "The Software Architecutre of An Interactive Remote Instruction System For Heterogeneous Netwrok Envirnonments",6th IEEE Symposium on Computers and Communications, Tunisia, 2001

17. [SCC-02] K. Maly, C. Wild, C.M. Overstreet, H. Abdel-Wahab, A. Gupta, A. Youssef, E. Stoica, R. Talla and A. Prabhu: "Interactive Remote Instruction: Lessons Learned", Computer Science Department, Old Dominion University, Norfolk Virginia, presented in WebNet96

18. [SCC-03] Netmeeting Website:"http://www.microsoft.com/windows/netmeeting/"

19. [SCC-04] Click 2 Meet Website: "http://www.fvc.com/products/clicktomeet.htm"

APPENDICES

APPENDIX A:

VIDEO CONFERENCE SYSTEM GUIDE

Video Conferencing System Guide

This guide describes how to use Video Conference. It's an application that uses the JMF 2.0 API to play, capture and write media data. Video Conference uses JMF RTP APIs to receive and transmit media streams across the network.

Video Conference will capture, send the video and audio stream from a USB web camera input. The user can later use the Microsoft windows media player to display this file. It's 100% written in Java, using the media framework library (JMF) from Sun. It's tested on Windows 2000.

Steps in preparing for the Video Conferencing environment

1. Install J2SDK1.4

Install the Java 2 System Development Kit version 1 and higher. Don't forget to specify the environment variables in the Class and ClassPath in order to make the .java program is run able.

2. Install Java Media Framework 2.0 (Only this latest version support Video Conferencing)

3. Install Web Cam Drivers

Contents

2.5 Closing the Current Player

3. Saving a Media Stream to a File

4. Transmitting Media Streams via RTP

5. Capturing Media with Java Video Chat

5.1 Previewing Captured Media Data

1. Getting Started

Before the user can run the Java Video Chat, the user must have J2SDK1.4 kit and JMF 2.1.1 installed. The user will just need to follow the very user friendly wizard while installing both of these kits. Both the JMF 2.1.1 and J2SDK classes must be specified in the class path.

1.1 Running Text Chat
To launch Text Chat, the user need to run or invoke: java Client12
When click connect, there will be input message box asking for the IP address of the next user. When entered, and click OK, the connection is established and there will be "Welcome" in the Text Area. Both of the users can start to chat with each other. When click "go to video chat", there will be "Video Conference" pop up.

When Text Chat is launched, the menu is visible:

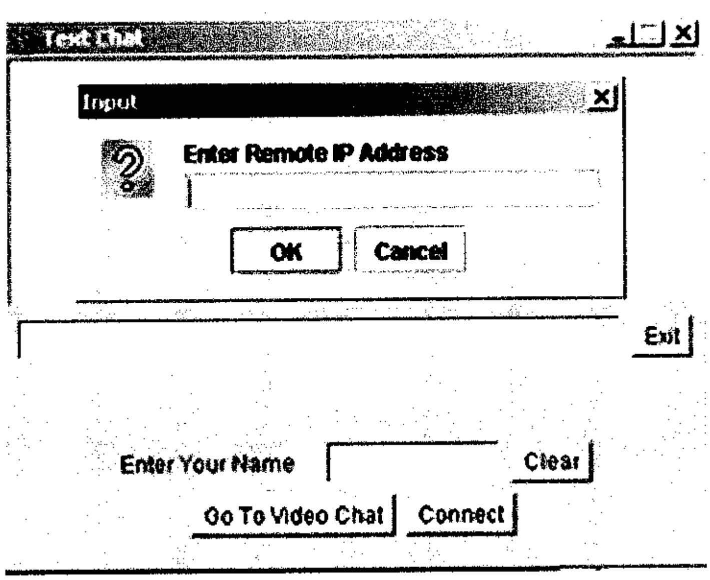


Figure AppendixA.1 Text Chat

1.2 Running Java Video Chat

Video Conference started.

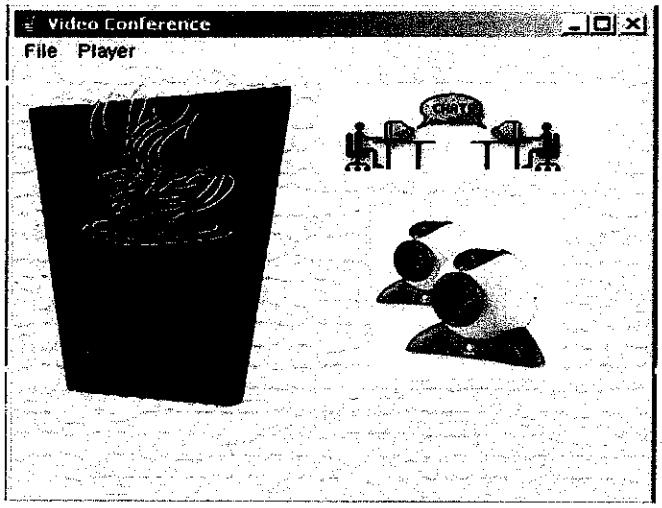


Figure AppendixA.2 Video Chat

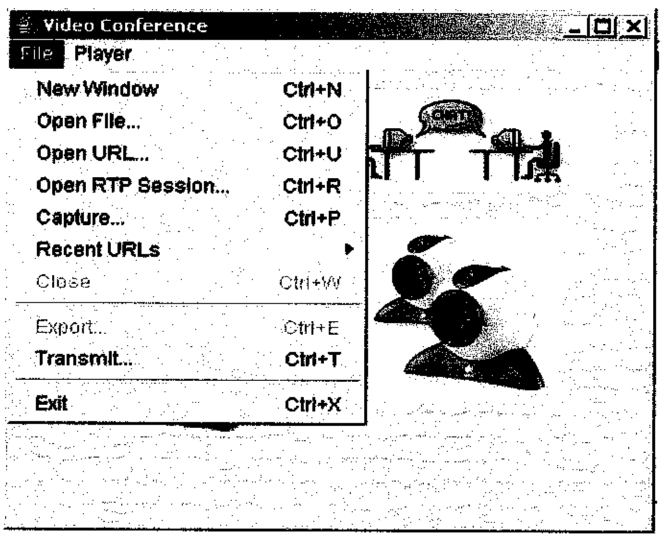


Figure AppendixA.3 Video Conference File Menus

Figure AppendixA.4 Video Conference Player Menu

1.3 Opening a New Java Video Chat Window

To create a new instance of the Java Video Chat application, select **New Window** from the File menu. A new instance of Java Video Chat without affecting the current one. Both applications share the same Java Virtual Machine.

1.4 Exiting Java Video Chat

To exit Java Video Chat, select Exit from the File menu.

1.5 Menu Summary

Table AppendixA.1 Video Conference's Navigational Structure

Navigation	Shortcut	Descriptions
File | New Window	Ctrl+N	Launches a new instance of the JVC applicatior
File | Open File	Ctrl+O	Opens a file for playback.
File | Open URL	Ctrl+U	Opens a URL for playback.
File | Open RTP Session	Ctrl+R	Opens an RTP receive / playback session.
File | Capture	Ctrl+P	Opens a dialog from which the user can select a capture device and begin capturing data.
File | Recent URLs		Opens a menu from which the user can select a URL from a list of URLs recently accessed by Java Video Chat.
File | Close	Ctrl+W	Closes the current Player.
File | Transmit	Ctrl+T	Opens a dialog from where the user can transmi media data over the network.
File | Exit	Ctrl+O	Terminates the application.
Player | Auto Play		Toggles the auto play state of the current Playe When auto play is enabled, the player begir presenting the media stream automatically.
Player | Auto Loop		Toggles the auto loop state of current Playe When auto loop is enabled, the playe automatically replays the media stream when th end of the media is reached.

Java Video Chat lets the user play media streams from a variety of sources. There are files, URLs, or RTP transmissions.

2. Playing Media Data with Java Video Chat

2.1 Playing a File

To play a file:

1. Select Open File from the File menu. The Open File dialog is displayed:

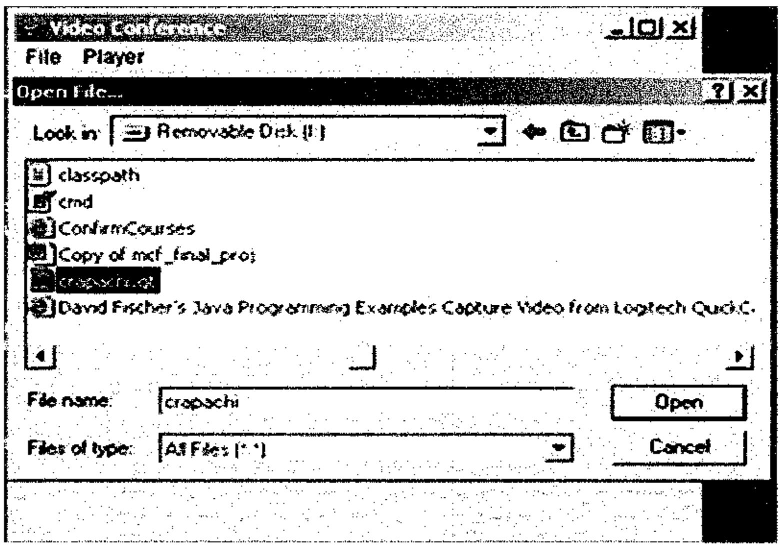


Figure AppendixA.5 Open File Menu

2. In the Open File dialog, select the file that the user wants to play and click OK. A JMF Player for the selected file is created. If Auto Play is enabled, the player is started and presentation of the selected begins automatically.

Figure AppendixA.6 Play File

1. If the format of the file is not supported by JMF, an error message is displayed.

Playing a URL
To play a URL:

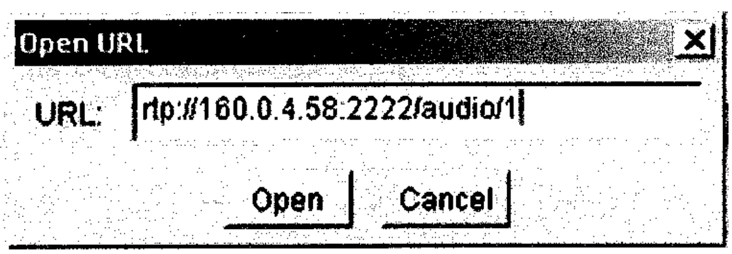


Figure AppendixA.6 Open URL

1. Select Open URL from the File Menu. The Open URL dialog is displayed:
2. In the Open URL dialog, enter the URL that the user want to play, for example http://164.0.4.58/Clip1.mov
3. Click **Open.**

If the URL exists, and its format is supported by JMF, a JMF Player for this URL is created and started. If the URL cannot be found or the format is not supported by JMF, an error message is displayed. Supported protocols for the JMF 2.1.1 implementation include HTTP, FTP, and FILE.

The user uses the Open URL command to play any URL, including RTP players. When user use Open URL to play RTP stream, the first stream of the RTP session is played. To use Open URL to play an RTP Stream:

1. Select **Open URL** from the File Menu
2. In the Open URL, enter the media locator that identifies the RTP source which the user wants to receive the RTP stream for example, rtp://164.0.4.58:49150/Clip1.mov
3. Click **Open.**

The **Open URL** command is also used to capture media data.

2.2 **Receiving and Playing RTP Media Streams**

To receive a media stream using the RTP protocol:

1. Select Open RTP Session from the File menu. The Open RTP Session dialog is displayed:

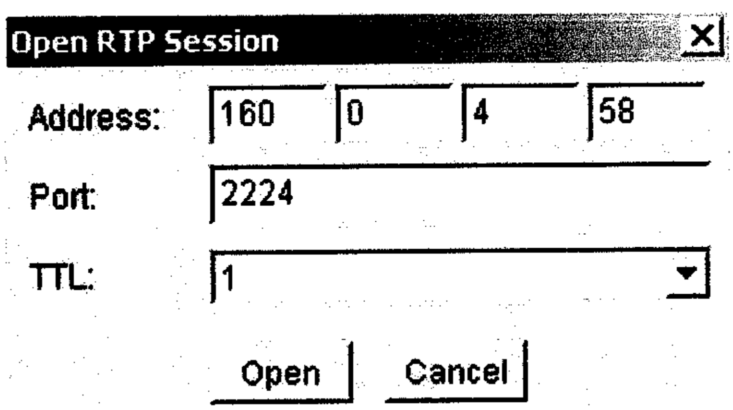


Figure AppendixA.8 Open RTP Session

2. In the Open RTP Session dialog, enter the session Address field (eg. 164.0.4.58), enter the port no. in the Port field (eg. 2222) and select a TTL (Time To Live). This constitutes the address of the transmitting source (or multicast address if it's a multicast transmission).

The format of an RTP Media Locator is: rtp://address: port[:ssrc]/content-type/[ttl] where:

Table AppendixA.2 Format of RTP Media Locator

Format	Definition
Address	The IP address of the RTP session
Port	The port of the RTP session
SSRC	SSRC Identifier of the source from which data is to be received. If SSRC is not specified, the first stream detected by the RTP Session Manager will be selected as a stream for the Data Source.
TTL	Time to Live of the RTP session
Content-type	A string defining the data content type. E.g. video, audio, motion, text, etc. The RTP media handler (Player) will be created for this specific type.

3. After the user enter the RTP session address and port, click the Open button. Java Video Chat creates an RTP Session manager for RTP media location.

Java Video Chat displays two buttons during an RTP session: **Statistics**, and **Close RTP Session**:

- Clicking **Close RTP Session** closes the RTP Session Manager.
- Clicking the **Statistics** button opens a Participant List window that displays information about the RTP session participants. This window contains another **Statistics** button that displays a window that displays the RTP session statistics.

When RTP packets actually begin to arrive, a JMF Player is opened to present the RTP media stream. This player has all of the standard JMF presentation controls except for the Seek bar, which is not relevant for RTP streams.

2.4 Controlling the Presentation

While media data is being played, the following controls are available:

Table AppendixA.3 Controls Actions with Description.

Controls	Description
Play	Starts presentation of the media data. If the presentation was paused, playback resumes from the point at which it was paused.
Pause	Pauses presentation of the media data. The Play button will resume playback from the point at which it was paused.
Step Back / Forward	Steps back or forward by one frame. If either button is kept pressed, the action automatically repeats at the frame per second.
Seek	Sets the media to a new playback position. You seek by dragging the Seek slider or clicking within the Seek bar. When the media is playing, the seek does not take effect until the mouse button is released. When the media is paused, dragging the Seek slider will cause the video to be continually updated.
Mute	Mute the audio. The presentation (including video, if applicable) continues while the audio is muted. Clicking the Mute button mutes the audio, clicking it again resumes normal audio playback.
Volume	Right-clicking on the Mute button pops up a volume slider. This sets the volume level. The user can drag the volume slider to increase or decrease the volume. Dragging to the right increases the volume. In some situations, the volume slider in unavailable.
Information	Displays the media properties, which contains information about the media being played, including the JMF version, URL name, duration, bit rate, video encoding, window size, frame rate, audio encoding, and audio quality.
Zoom	Right-clicking on the video window displays a menu from which you can set the image scale. Available options are 1:2 (half size), 1:1 (normal size), 2:1 (double size) and 4:1 (quadruple size).
Rate	Right-clicking on the Information button displays a menu from which you can set the media playback rate. The rate can be set to

	any value from 1/4 of the normal speed up to 8 times the normal speed.
Auto Loop	Enabling **Auto Loop** in the Player menu causes the media to play continuously, restarting each time it reaches the end of the media. Selecting the auto replay option repeatedly toggles the replay setting on and off.

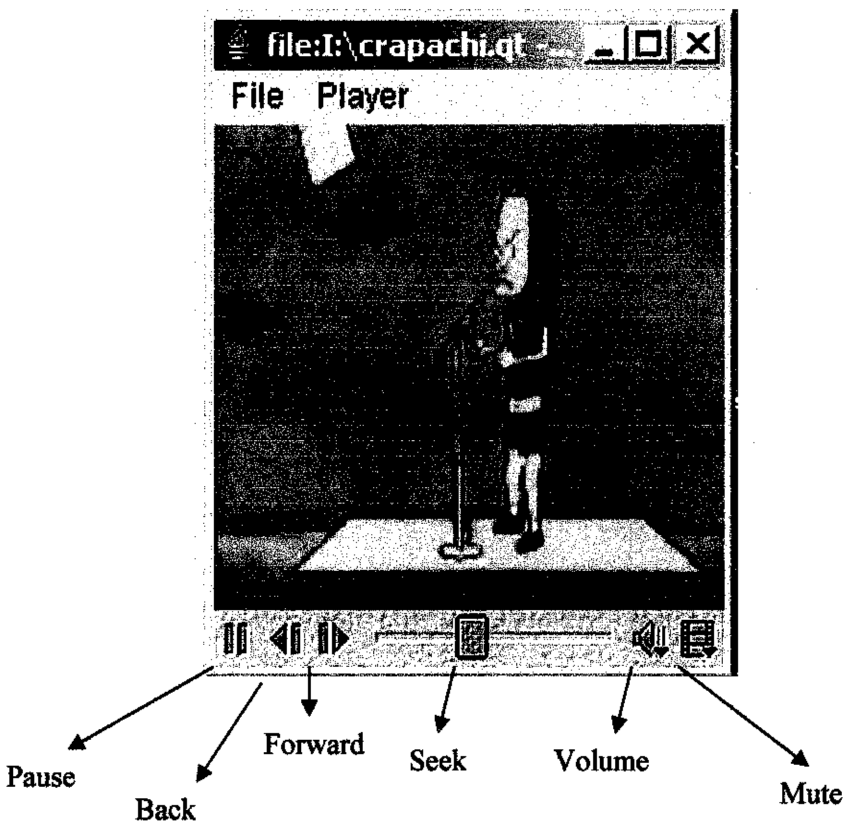


Figure AppendixA.9 Control Actions

2.5 Closing the Current Player

To close the current Player, select Close from the File Menu. This removes the presentation controls and closes the current player.

4. Transmitting Media Streams via RTP
The user can use the Java Video Chat Transmit command to transcode a media stream and transmit it across the network using RTP.

1. Select Transmit from the File Menu. The Transmit wizard is displayed:

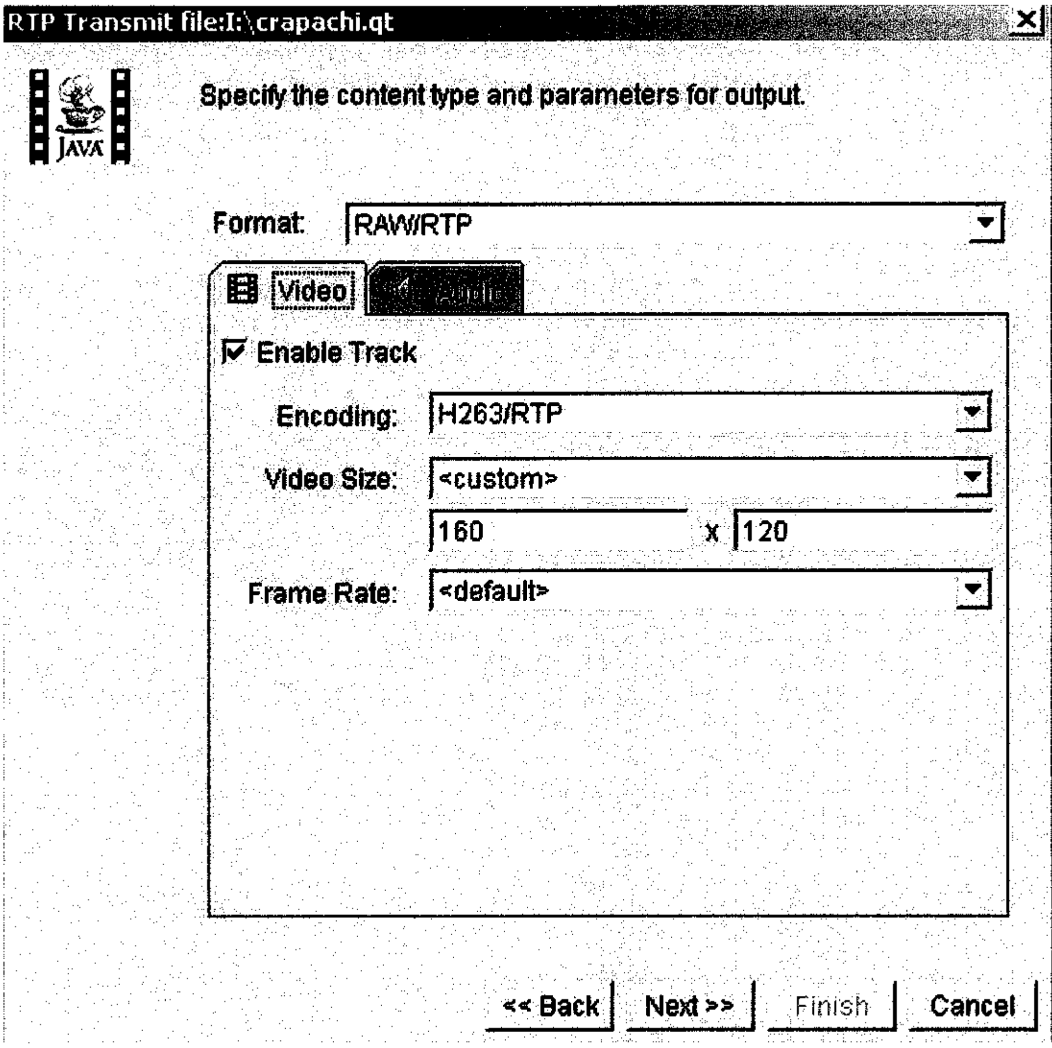


Figure 4.17 Transmit File and Media

2. Select the source location of the media. The user can select a File or a Capture source.
3. Click on Next to go to the RTP format page and set the formats of the Video/Audio tracks to be transmitted. The user can disable any tracks that she/he doesn't want to transmit.

4. Click on Next to go to the RTP parameters page. Enter the session address, port, and TTL for each track that the user wants to transmit.

5. Click on Finish to start the transmission. Click on Back to change the settings on the previous
pages.

5. Capturing Media with Java Video Chat

Java Video Chat supports media capture through special DataSources called Capture DataSources.

5.1 Previewing Captured Media Data

Java Video Chat enables the user to monitor the media data as it is captured. To preview captured media data:

1. Select Capture from the File menu. The Select Capture Device dialog is displayed:

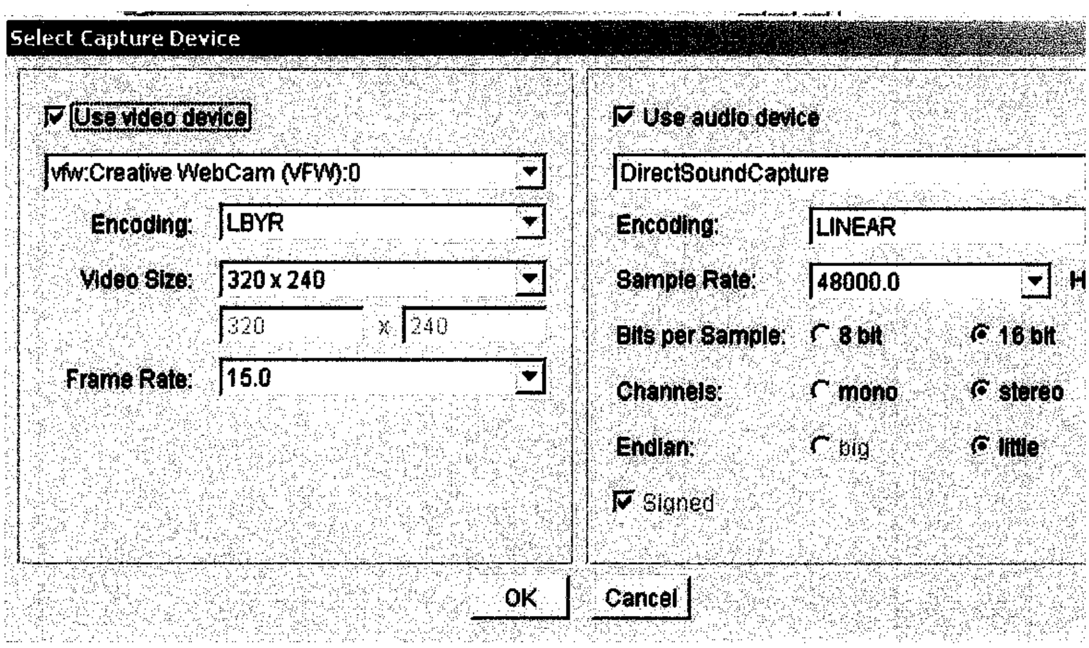


Figure AppendixA.10 Select Capture Devices

2. Select the capture device from which the user wants to preview incoming media data
and the forma in which the user wants to capture.
3. Click OK to begin capturing the media.

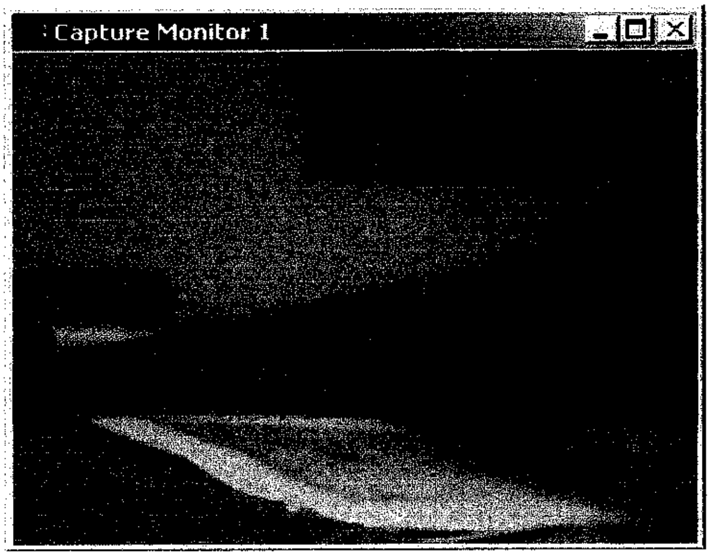


Figure AppendixA.12 Capture Monitor

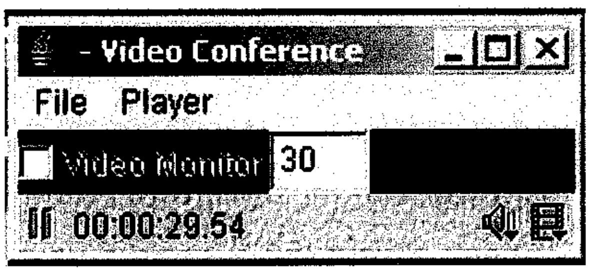
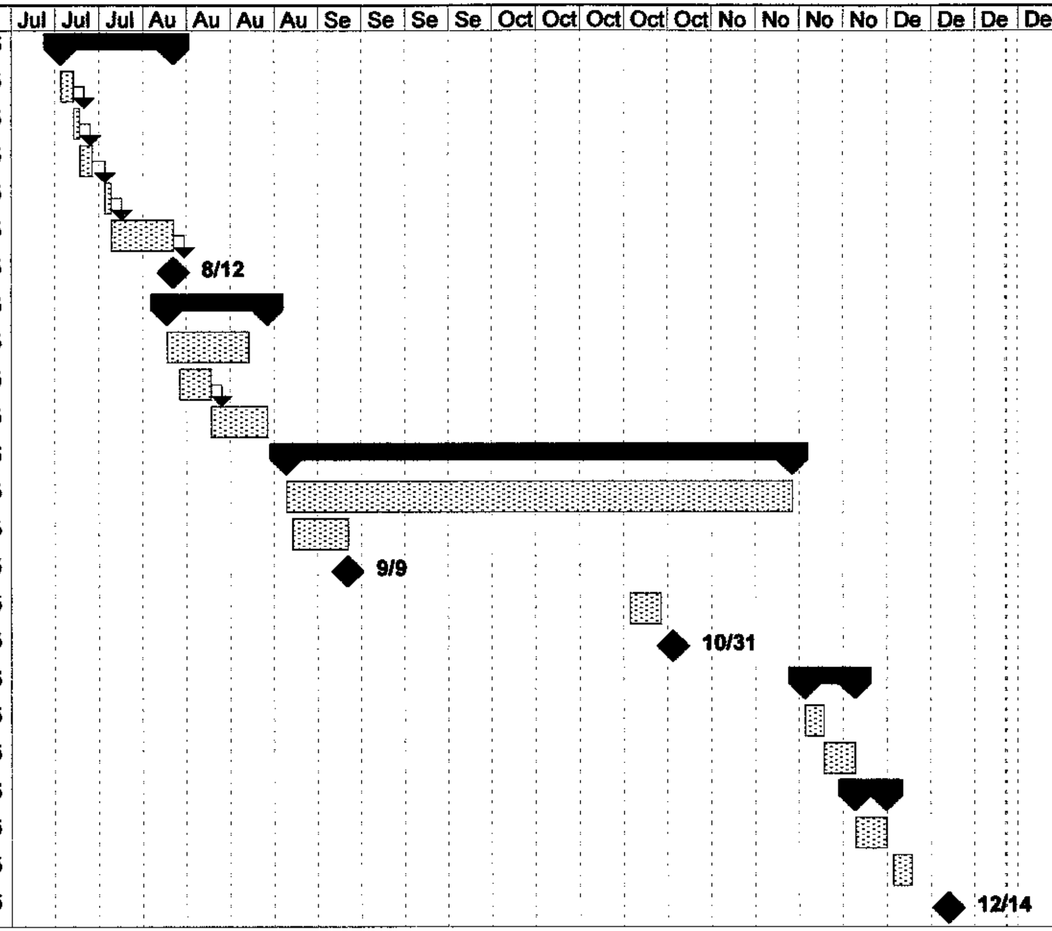


Figure AppendixA.13 Audio Monitor

APPENDIX B:

PROJECT GANTT CHART

ID	Task Name	Duration	Start	Finish
1	**Requirement Analysis**	**14 days**	**Mon 7/25/05**	**Fri 8/12/05**
2	Selection of project topics	2 days	Mon 7/25/05	Tue 7/26/05
3	Determine project scope	1 day	Wed 7/27/05	Wed 7/27/05
4	Determine project Schedule	2 days	Thu 7/28/05	Fri 7/29/05
5	Specify objectives	1 day	Mon 8/1/05	Mon 8/1/05
6	Preliminary Research Work	8 days	Tue 8/2/05	Thu 8/11/05
7	Submission of preliminary	0 days	Fri 8/12/05	Fri 8/12/05
8	**Design and Specification**	**13 days**	**Thu 8/11/05**	**Fri 8/26/05**
9	Research	10 days	Thu 8/11/05	Tue 8/23/05
10	Designing of flowchart	4 days	Sat 8/13/05	Wed 8/17/05
11	Designing Interface	7 days	Thu 8/18/05	Fri 8/26/05
12	**Development**	**59 days**	**Tue 8/30/05**	**Fri 11/18/05**
13	Program the project	59 days	Tue 8/30/05	Fri 11/18/05
14	Prepare progress report	7 days	Wed 8/31/05	Thu 9/8/05
15	Submission of progress report	0 days	Fri 9/9/05	Fri 9/9/05
16	Prepare final draft	5 days	Mon 10/24/05	Fri 10/28/05
17	Submission of final draft	0 days	Mon 10/31/05	Mon 10/31/05
18	**Testing**	**6 days**	**Mon 11/21/05**	**Mon 11/28/05**
19	Questionnaires	3 days	Mon 11/21/05	Wed 11/23/05
20	Full testing	3 days	Thu 11/24/05	Mon 11/28/05
21	**Operation and maintenance**	**4 days**	**Tue 11/29/05**	**Sat 12/3/05**
22	Product Touch up	4 days	Tue 11/29/05	Sat 12/3/05
23	**Oral Presentation**	3 days	Mon 12/5/05	Wed 12/7/05
24	**Submission of project dissertatio**	0 days	Wed 12/14/05	Wed 12/14/05



Project: FYP Timeline
Date: Fri 12/23/05

Task	Milestone	External Tasks
Split	Summary	External Milestone
Progress	Project Summary	Deadline

APPENDIX C:

EDX-16 CERTIFICATE OF PARTICIPATION

ENGINEERING DESIGN EXHIBITION 16

CERTIFICATE OF PARTICIPATION

This is to certify that

SITI SAFINAS MOHD RASHID
830808-10-5490

is awarded this certificate for his/her participation in the category of

Information Communication Technology &
Business Information System
Final Year Project

in the sixteenth edition of Universiti Teknologi

PETRONAS' Engineering Design Exhibition held

on 19th October 2005.

Rahmat I.K. Shazi
Advisor, EDX 16

Ibrahim Kamaruddin

AP Ir Dr Ibrahim bin Kamaruddin
Director, Undergraduate Studies
Universiti Teknologi PETRONAS