REQUIREMENTS ANALYSIS OF A MULTIMEDIA PATIENT INFORMATION SYSTEM IN TELEMEDICINE APPLICATIONS

GERARD MICHAEL DUNPHY

REQUIREMENTS ANALYSIS OF A MULTIMEDIA PATIENT INFORMATION SYSTEM IN TELEMEDICINE APPLICATIONS

BY

©GERARD MICHAEL DUNPHY, B. ENG., P. ENG.

A THESIS SUBMITTED TO THE SCHOOL OF GRADUATE STUDIES IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ENGINEERING

FACULTY OF ENGINEERING AND APPLIED SCIENCE

MEMORIAL UNIVERSITY OF NEWFOUNDLAND

DATE: SEPTEMBER 30, 1999

ST. JOHN'S NEWFOUNDLAND CANADA

ABSTRACT

The medical patient record still consists for the most part of paper and film based records. Diagnostic information obtained from a variety of imaging modalities, text records, laboratory results, and transcripts of audio dictations make up a typical patient record. These records are then collected and stored in a central record database, filed by the patient's name or medical identification number.

This method of access leaves much to be desired. The record has a considerable chance of being lost or mislaid as it is handled by medical personnel. Requests for the record from a practitioner with an interest in the patient can mean that the record is inaccessible for a period of days. Considerable time and expense is expended in an effort to store and organize patient records, particularly at a large institution such as a hospital. In the meantime, records generated at a different institution may not be accessible at all. This is particularly significant when dealing with applications in telemedicine, when the practitioner with a requirement to access the patient record may be geographically remote from the institution in which the record is stored.

Several computerized systems have been developed to store some or all of this record in a digital format. In particular, much emphasis has been placed on the storage of text information and on the storage of images. In most cases, these databases exist alone and without the capability of communicating with each other.

The ubiquitous presence of the World Wide Web as a medium for the dissemination of information is a recent occurrence, but it is also a growing phenomenon.

Web access is commonplace, especially in a professional setting.

Given these factors, it was decided to investigate the feasibility of addressing the shortcomings of traditional medical record storage systems through the use of a Web based multimedia patient information system in a client-server network topology. A prototype system was developed and used to perform testing. User testing with medical personnel was conducted to assist in establishing the parameters of the design of a system of this nature. The system was also used to measure delay in downloading information from a central database.

It was found that a high degree of acceptance exists for a system of this nature. In order for a solution of this type to be practical, low download time, multimedia file access (audio, image, and video as well as text), and conferencing would be requirements. Access using a high speed access technology such as ADSL or possibly ISDN would be required in order to provide reasonable access time to the central database.

ACKNOWLEDGMENTS

As I have come to realize, the writing of a thesis is a uniquely solitary endeavour. However, while this document bears my name alone, it could not have been completed without the support and encouragement of many others, all of whom I owe a deep debt of gratitude.

To my wife Edie, I do not have the words to express my gratitude for your patience, encouragement, and support. You have provided tremendous support and I literally could not have done it without your assistance and advice. I hope I can show the same degree of support in my turn.

Dr. Cecilia Moloney's patience and understanding of the constraints placed on this endeavour by full-time employment are greatly appreciated. She is both an excellent educator and a true leader.

Thanks to all those at Newfoundland & Labrador Hydro, NewTel Communications, and the Health Care Corporation of St. John's, for their invaluable assistance and input: Madan Rana, Eric Downton, and Don Richards, for their support and understanding, particularly during my leaves of absence; Doreen Whelan and Peter Green at NewTel, for unhesitatingly providing access to an ADSL circuit; Dr. Peter Hollett, Dr. Khalid Aziz, and Marian Crowley, for providing honest, positive input; and a host of others. I had no idea when I began how much assistance would be required, nor how willing so many others would be to assist. I thank you all.

Many others have been friends and mentors to me, and in no small way this thesis would not have been possible without the examples they have shown: my mother, Margaret Dunphy; my brother Joseph; Leo Ryan, and Fred Martin. I am privileged to have been guided by the principles that you demonstrate in your daily lives, and I strive to emulate the example set by you all.

And last but not least, to Sunny and Sam, my constant companions.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

ACR	American College of Radiologists
ADSL	Asymmetrical Digital Subscriber Line
API	Application Programming Interface
ASTM	American Society for Testing and Materials
ATM	Asynchronous Transfer Mode
BISDN	Broadband Integrated Services Digital Network
BRI	Basic Rate Interface
CCD	Charge Coupled Device
CORBA	Common Object Request Broker Architecture
CORBAmed	Healthcare DTF of CORBA
CT	Computerized Tomography
DBMS	Database Management System
DICOM	Digital Imaging and Communications in Medicine
DMS	Document Management System
dpi	dots per inch
DTF	Domain Task Force
EC	European Community
GUI	Graphical User Interface
HIS	Hospital Information System
HL7	Health Level Seven standard

HTML	HyperText Markup Language
HyTime	Hypermedia/Time-based Document Structuring Language
IEEE	Institute of Electrical and Electronics Engineers, Inc.
IMSIG	Image Management Special Interest Group
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ISP	Internet Service Provider
JPEG	Joint Photographic Experts Group
LAN	Local Area Network
LCR	Least Cost Routing
MTU	Maximum Transmission Unit
MHEG	Multimedia/Hypermedia Experts Group
MIS	Medical Information System
MRI	Magnetic Resonance Imaging
NEMA	National Electrical Manufacturers Association
ODA	Open Document Architecture
ODBMS	Object Database Management System
OO	Object-Oriented
OOD	Object-Oriented Design
OMG	Object Management Group
OSI	Open Systems Interconnection
PACS	Picture Archiving and Communications System

PCM	Pulse Code Modulation
pdf	probability distribution function
PIS	Patient Information System
QoS	Quality of Service
SGML	Standard Generalized Markup Language
SLIP	Serial Line Internet Protocol
SNR	Signal to Noise Ratio
TCP/IP	Transmission Control Protocol/Internet Protocol
TDM	Time Division Multiplexing
UDP	User Datagram Protocol
WAN	Wide Area Network
WHO	World Health Organization
WIMP	Windows, Icon, Menu, Pointer interface
WWW	World Wide Web

Chapter 1

Introduction

1.1 General

In a typical medical setting such as a hospital or clinic, large quantities of patient-related information must be processed, stored, and retrieved. Text records store the patient's vital statistics, such as name, date of birth, and historical information regarding past treatments. In addition, there may be several different types of device which perform some sort of diagnostic or monitoring function. These devices usually have a common set of features: they monitor or record some aspect of the patient's health; they provide an output, (often visual, but in any case some format that may be considered to be a subset of the class of information which is generally termed "multimedia") for the practitioner's assessment; the information provided is often stored in some form or another for future reference; and the information is usually not stored in a format which makes it easily and quickly transportable between different systems or geographically dispersed locations.

Traditionally, this information has been stored and processed in a "patient-centric" manner; that is, the information at a particular location has been filed in physical documents, and in some manner such that the only consistent method of indexing this information is the name or identification number of the patient. This method has many limitations, including the following: the inability to easily search databases for epidemiological information; the possibility of loss of part or all of the record; long time delays in the storage and retrieval operations; and the inability to easily manipulate the component of the information which is increasing most rapidly in volume, i.e. computer-generated information from various diagnostic equipment.

Many systems have been proposed and implemented which attempt to resolve some or all of the problems associated with patient record storage and retrieval. These range from systems which are concerned mostly with image management (picture archiving and retrieval systems, or PACS) to systems which attempt to mimic the entire patient record. Additional functions such as group/collaborative work, search capabilities, and distributed architecture may or may not be components of such a system.

Additionally, as public funds for health care become less flexible, more emphasis is being placed on centralized access to medical products and services as an area of potential cost savings. The concept of *telemedicine* has recently arisen in an effort to realize potential savings, whereby remote medical practitioners may avail of the services of central specialists and equipment which would otherwise be inaccessible. Telemedicine uses communications infrastructure in assisting the medical practitioners to exchange information between remote and central locations.

1.2 Problem Statement

While several computerized systems have been implemented which attempt to address some of these issues, little or no work has been performed to quantify the requirements of such a system in a telemedicine setting. By quantifying the requirements of a document management system operating in a telemedicine environment, we are better able to understand how such a system can be implemented in an operational milieu. This quantification would permit optimization of system configurations based on user requirements.

With this background, then, the problem may be stated in the following form:

> *It is proposed to investigate the requirements of an integrated medical document management system (DMS), particularly as they relate to telemedicine. The requirements of such a system pertaining to an operational setting will be examined. It is anticipated that the result will be an understanding of the parameters of a DMS as it pertains to telemedicine applications.*

Such a system would result in fast and accurate file storage and retrieval, permit information gathering in a more sophisticated manner than is permissible with a traditional system, permit consolidation of an individual's records that may currently exist at more than one institution (and therefore be relatively difficult to share between locations), and result in faster service delivery as the result of shorter waiting periods. It is anticipated that the system will also take advantage of the multimedia applications capabilities of commonly available computer platforms to provide services which exceed the requirements of document management, e.g. collaborative work and search/retrieval capability.

As described herein, a prototype document management system utilizing standard file formats, encoding methods, and communication protocols, and running on a commonly available and inexpensive computer platform, has been developed as a solution to the limitations of the record keeping methods outlined above. Utilizing the ubiquitous HTML format and the functionality of the Java programming language, the system is distributed, easily accessible, and capable of functioning across a wide variety of operating systems, hardware platforms, and communications infrastructures. This system could be capable of configuring itself to optimize its performance based on selected criteria, such as user requirements or communications, hardware, and software constraints.

The system has been used to evaluate the needs of and constraints placed on this type of application, particularly as they pertain to the field of telemedicine. By analyzing the user needs, information requirements, and bandwidth requirements of these applications, a clearer understanding of the optimal design of a telemedicine DMS may be determined. By measuring user response to the system, optimal configurations may be obtained which maximize the utility of such a system for an individual user.

This thesis begins with an introduction which will establish the motivation behind, and background for, this investigation. Chapter 2 will discuss literature relevant to the thesis, including aspects of the design of a system of this nature - software design, multimedia databases in general, relevant standards, and a description of similar implementations. Chapter 3 contains the specific details of the design, implementation, and testing of this system. The test results are discussed and interpreted in detail in Chapter 4, and finally Chapter 5 presents conclusions, recommendations, and future areas of investigation.

Chapter 2

Literature Review

2.1 Introduction and Definitions

The recent rapid deployment of computer systems in the field of health care has permitted the development of document management systems whose purpose is to increase the accuracy and speed of storage and retrieval of patient information; such systems are generally referred to as Hospital Information Systems (HIS), Medical Information Systems (MIS), or Patient Information Systems (PIS); in this document the acronym MIS will be used exclusively to avoid confusion. A subset of this type of system concerns itself solely with visual information and is known as the Picture Archival and Retrieval System (PACS).

In this chapter the literature regarding the topic of information systems pertaining specifically to the MIS will be reviewed. The deficiencies of current practices regarding the storage and transmittal of traditional paper-based medical patient records will be examined. The constraints on a computer-based patient record will be discussed, with particular emphasis on aspects of the design of such a system, and a review of some significant implementations described in the relevant literature will be reviewed.

The development of MIS has been driven by the recognized limitations of current techniques for the storage and retrieval of patient information [Kar90]. Typically, in traditional information storage systems, the patient record is a series of paper or film documents stored in an envelope or file folder in the central record database of a health care institution. This method has several disadvantages:

- Significant time is spent storing, indexing, and retrieving the patient file by records administrators and health professionals;
- There is considerable risk of the patient record being mislaid, damaged, or otherwise mishandled during the transmittal process, resulting in increased treatment costs as repeated examinations may be made necessary;
- Organization of records by a single indexing variable - the patient name or identification number - precludes the use of the collected information for other purposes, such as epidemiological studies and medical research;
- The health care practitioner may not have access to patient information located at institutions other than the one at which he/she is located. These records may provide significant information to the practitioner regarding the individual under their care.

The issues surrounding the management of patient information are significant. The delay resulting from the handling of the records of medical patients, especially those of hospital patients in need of immediate treatment, has been identified as one of the most significant problems facing the medical profession today [Hic90]. Estimates of the amount of information generated in a hospital vary; however, one institution alone has been estimated to produce 15 Gbyte of image information **per day** [Abo95]. With the proliferation of computerized information storage, manipulation and dissemination systems, development is shifting from the generation and acquisition of medical information to its management and processing [Won97].

The **integrated patient record** consists of a variety of information collected in multiple formats, often over an extended period of time, as shown in Table 1. Additional information may include a consulting health professional's thoughts and opinions, a record of activity, and plans for future treatment [Rig95]. The information contained within the patient record may be grouped; for instance, multiple records of several different types may constitute the record of an examination or treatment session, or one diagnostic modality may produce more than one record per session [Buc94].

Information source	Information type	Common Format
Magnetic Resonance Imaging (MRI)	Image	Computer Graphic
Computerized Tomography (CT)	Image	Computer Graphic
X-Ray	Image	Film
Ultrasound	Image	Computer Graphic/Video
Voice Narration	Audio	Magnetic tape/digital storage device
Written word	Text	Paper record/Computer database

Table 1. Common Patient Information Types

The term **multimedia** has been defined as

> *A computer based user interface technology that uses multiple input and output mediums, including audio, speech recognition, animation, video (disc, tape, etc.), bitmapped images, generated graphics, tactile controls, inferred and derived controls, hypertext and other multisensory rich data types in end user applications* [Sha95].

It is obvious from the above definition that the patient information types shown in Table 1 can be considered multimedia information.

The characteristics of a system which manipulates multimedia information of this nature may be more rigidly defined. A *multimedia information system* permits the segmentation, navigation, and storage of multimedia information [Swa93]. The *information type* may be classified as presentation, controlled by the sender, and non-presentation, controlled by the receiver [Yam93]. The information will be temporally *static* or *dynamic* in nature [Mal97]. Multimedia *traffic time constraints* will be either asynchronous, with unrestricted transmission delay; synchronous, with bounded transmission delay for each message, or isochronous, with constant transmission delay for each message [Stu95]. Multimedia *information interchange* will consist of message content, structure, and communication information [Col93].

The advantages of a computerized medical information system which allows the medical practitioner access to the information contained in the integrated patient record defined above have long been recognized. The World Health Organization (WHO) has stated that information technology solutions in health care can help to achieve the following goals [Who88]:

- increased efficiency of operation;
- improved recording and communication of information among health care professionals;
- improved accessibility of patient information;
- increased quality of health care;
- improved quality assurance;
- improved cost-effectiveness;
- improved epidemiological performance.

The most significant advantages of an integrated, computerized patient record include ensuring a coherent structure for data, thereby facilitating data storage and communications; providing data consistency and security, along with fast and selective access to information; and suitability for manipulation of large quantities of data [Kar90]. Banerjee has also pointed out that such a system provides low delay in search and storage, the ability to allow simultaneous access to information by more than one user, image manipulation, and almost instant information distribution [Ban94].

2.2 MIS Design Characteristics

Careful consideration of the design of a MIS will result in an implementation which is efficient, effective, and easily maintained and upgraded. The goals for such a system can be considered as follows [Lit94]:

- It maintains a complete, accurate repository of all data relevant to the health of an individual;
- It provides a readily available, useful and relevant representation of its controls to users at all times;
- It contributes to a collaborative workspace for health care professionals;
- It complements and communicates effectively with other tools and information systems as required.

Some of the more important aspects of the system design are considered herein. The relevant characteristics for such a system are illustrated in Figure 1; each characteristic will be discussed individually below.

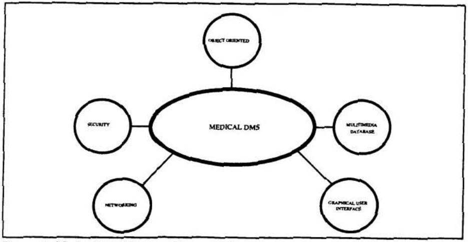


Figure 1. Medical DMS Design Characteristics.

2.2.1 Networking

It is almost a foregone conclusion that the implementation of a multimedia medical information system will be accessible through some sort of computer network. Virtually all computers in use today will have access to a network, whether through a direct, permanent connection to an Ethernet or Token Ring Local Area Network (LAN), or through access to the Internet or an Intranet via a Mosaic-style browser such as Netscape™ or Internet Explorer™ using an asynchronous TCP/IP protocol such as Serial Line IP (SLIP). Network access allows use of the system by users both in the local environment and remotely, and may permit access to distributed databases or heterogeneous systems that would otherwise remain inaccessible.

One advantage of a networked MIS is the ability of the information to be distributed between servers and even locations. In the course of an individual's lifetime, most people will have treatment and services performed at multiple medical institutions. A distributed medical information system would permit the access and retrieval of information from a variety of locations [Adj97]; as an example, an MIS might make use of a resource agent to retrieve historical patient records from distributed databases to a central query manager [Huh98].

Transport of large volumes of multimedia data over a network has inherent problems which may restrict the types of data being used. Connectionless protocols such as TCP/IP can severely restrict the accessibility of information which may have a temporal content, such as audio and animated video [Bra97]. This effect will be more pronounced in some areas than in others; for example, delay which causes jerky motion in video may be unacceptable for entertainment transmissions, while some delay may be acceptable in a low quality videoconference session.

In considering the applicability of a multimedia MIS in a networking environment, performance issues arise which must be contended with. The requirements of a multimedia MIS operating over a network include high available bandwidth, low latency, low jitter, and multipoint communication ability [Fuh94]. Issues of throughput, Quality of Service (QoS), multithreading, and synchronization must be considered [Paz97] [Bla94].

Current networking technologies commonly used include 10 Mbit/s and 100 Mbit/s Ethernet and 4, 16, and 32 Mbit/s Token Ring LANs; as well as Wide Area Network (WAN) technologies which can range from low speed (down to 56 kbit/s or below) serial communications to high speed Asynchronous Transfer Mode (ATM) or Synchronous Optical Network (SONET) communications operating in the tens to hundreds of Mbit/s range. Routers are commonly used to interconnect LANs, as they provide packet filtering and Least Cost Routing (LCR) capabilities in addition to routing. In most cases, TCP/IP is the transmission protocol which operates across these WANs. The term TCP/IP actually refers to a suite of protocols: the Internet Protocol (IP) is a Layer 3 (Network) protocol, while the Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and others correspond to Layer 4 (Transport) of the Open Systems Interconnection (OSI) model [Fre95].

Throughput of TCP/IP is limited by the inherent properties of the protocol itself. Because TCP/IP is a handshaking protocol which communicates bidirectionally and requires a positive acknowledgment of every packet sent, the throughput is limited by the maximum packet size and the delay, and is to a degree independent of the transmission speed of the physical connection [Tho97]. The maximum throughput in bytes is given by

$$Throughput \leq \frac{2^{16}-1}{Delay}$$

If the maximum delay acceptable in a multimedia MIS is assumed to be 150 ms [Stu95], this translates into a maximum theoretical throughput of 436,900 byte/s.

Currently, the IP protocol in use (IPv4) does not permit the specification of QoS parameters. Under the QoS prioritization scheme proposed for the next version of IP (IPv6), the IP header will contain a flag to indicate the relative priority of the packet [Bra97]. Applications which require low jitter or synchronization will then be better able to compensate for delay variations.

Perhaps the most commonly used architectural paradigm used in a networked environment today is the client-server architecture. Data and programs reside on one or more servers, to which multiple clients connect when they wish to execute an application. Utilizing client-server architecture allows resources to be concentrated in a relatively small number of centrally located servers, while clients often need not be as fast or as powerful as if they were required to run and store the complete application and its associated data.

2.2.2 Object-Oriented Design

Most software being developed today for distributed environments is written in a language which supports to a greater or lesser degree the concepts of Object-Oriented Design (OOD). Object-oriented programming is defined as

> *a method of implementation in which programs are organized as cooperative collections of objects, each of which represents an instance of some class, and whose classes are all members of a hierarchy of classes united via inheritance relationships* [Boo90].

An *object* is then defined as

> *[an entity] that combines the properties of procedures and data since [it performs] computations and [saves] local state. An object can only change state, behave, be manipulated, or stand in relation to other objects in ways appropriate to that object* [Boo90].

Object-oriented languages have several advantages over traditional, structured programming. The distinguishing characteristics of object-oriented languages include the concepts of abstraction, encapsulation, modularity, hierarchy, typing, concurrency, and persistence. Of these characteristics, those of abstraction, encapsulation, modularity, and hierarchy are considered the major elements. These principles permit the development of compact, easily modified programs whose design may be constructed so that they mimic the behavior of the real world. Code reuse through the judicious definition of objects allows for rapid program development, while information hiding (encapsulation) simplifies programming by multiple developers by removing the details of object implementation.

2.2.3 Graphical User Interface

Any system of this nature should provide a natural, intuitive user interface which minimizes the learning time for a user while providing the required information in the desired degree of detail. A user interface is not merely a display; it is the medium by which the application user communicates with the application. The user interface must balance the need for providing complete information with the requirement for ease of comprehension. Most software being implemented today utilizes a Graphical User Interface (GUI) for ease of use and flexibility. The Windows, Icon, Menu, Pointer (WIMP) concepts have become familiar to most users of computers, through the ubiquity of GUI-based operating systems such as Microsoft's "Windows" family , and the UNIX-based XWindows.

The design of an effective user interface for any system of a relatively high degree of complexity is an iterative process. User input is crucial to the design process; it allows feedback regarding the relative usefulness of the interface and allows the developer to improve the interface while still in development. The user interface design process consists of five steps [Hic90]:

1. Goal setting - e.g. easy to learn, satisfactory to users
2. Identification of design issues
3. Interface prototyping
4. Usability testing
5. Evaluation

Many professional programming environments now have the capability to generate user interface components quickly and easily, thereby greatly reducing the time required to develop and modify the user interface components.

2.2.4 Security

In order for an MIS to be considered useful, the user must have a high degree of confidence that the information presented is valid. This addresses the concept of trust. By saying that a source is trusted, two ideas are implied: that the information is authentic, i.e. the information has not been modified by an unauthorized source; and that confidentiality has been maintained, i.e. that the information has not been disclosed to an unauthorized source [Won96]. The security requirements for a networked MIS application include enabling the implementation of strong security measures, and enabling the sharing of data across the network [Abr98]. Strategies for security include limiting access, effective password administration, filtering through the use of firewalls, data administration, and the use of cryptography [Won96].

2.2.5 Object-Oriented Programming Environment

In conjunction with HTML, use of the Java programming language permits the use of such familiar information manipulation objects as toolbars, radio buttons, pick lists, and edit fields. Java is fast becoming the language of choice for object-oriented development in an "Internet/Intranet" environment. An OO programming language, Java is designed to facilitate fast software development through the innovative use of such capabilities as the garbage collection function, which automatically releases memory that is no longer needed [Nie96] [Hof97].

The most unique and useful capability of Java is its platform independence. Java source code is compiled into platform independent *byte code*, which is then executed on a platform specific Java virtual machine. In this manner, code which is written following the standards of Sun's 100% pure Java specification will be guaranteed to run on any platform for which a virtual machine exists. This feature ensures the maximum level of portability for programs written using Java, and it allows the developer to concentrate solely on the specifics of the implementation without having to be concerned with the details of the hardware and operating system on which the software will run [Fla97].

2.3 Multimedia Database Systems

The Database Management System (DBMS) used to manipulate multimedia information has evolved in parallel with the demands of applications and users. Early implementations of multimedia DBMSs usually consisted of a series of multimedia files stored in concert with a traditional relational database, wherein textual information associated with the multimedia database was stored. Manipulation of the multimedia files was performed by applications which existed externally to, and separate from, the DBMS itself. This type of DBMS has largely been supplanted by the Object DBMS (ODBMS) paradigm. In this instance, the DBMS has been designed solely for the purposes of storage and manipulation of information whose requirements surpass those of traditional text databases. The storage, retrieval, search and display requirements of multimedia information are intrinsic properties of a ODBMS.

Figure 2 shows an example of the architecture of a multimedia DBMS. The database can be considered three separate logical, although not necessarily physical, entities: an alphanumeric database of textual information, a multimedia database of multimedia objects, and a feature database which is used for querying and content-based retrieval. The user interface consists of a composition module, allowing the development of composite multimedia objects from the individual objects contained within the database; an insertion module, to allow the creation of new objects; a feature processing module, to extract the pertinent features to allow search and retrieval, and an interactive query module, permitting the user to search and display information contained within the database [Gro94].

Multimedia DBMSs have some unique requirements which separate them from traditional text-oriented DBMSs. A multimedia DBMS must provide for the acquisition and display of a variety of media, most of which require relatively specialized equipment

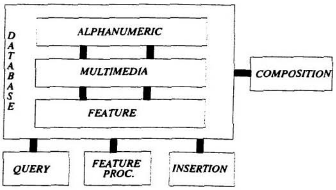


Figure 2. Multimedia database architecture (from [Gro94]).

- cameras and displays for image and video, microphones and speakers for audio, and so on [Gol94]. It must be capable of storing large amounts of data; both offline and online storage devices are therefore required. It must also be capable of post-processing the multimedia data, performing such operations as compression/decompression and classification. In order to maximize utility, information search and retrieval must be content-based. Unlike the exact match search used in traditional relational databases, multimedia search capability must by necessity take into consideration a vague, "near-match" type of argument [Yan95]; a requirement, for example, which would be necessary in performing epidemiological studies. The multimedia DBMS must also take into consideration the spatial and temporal constraints imposed on media types such as video and audio, where sequence and spatial arrangement are properties which may be inherent to the understanding of the information presented [Adj97].

Owing largely to the unique properties of the data stored in a multimedia DBMS, a satisfactory method for search and retrieval of information remains unresolved. Information retrieval in a multimedia context is not as straightforward as when dealing with a simple alphanumeric query; instead, it may be considered as a "best match" process, whereby documents are retrieved based on their similarity to a keyword search criterion [Sud93]. In this instance, there may be no clear "equality" to a search term. Multimedia information is rich in content; there may be many ways to categorize the information contained therein. One may specify the information using key tags or identifiers, conditional statements, and by similarity descriptions - "like" and "as" statements [Gol94]. On the other hand, a more sophisticated specification method would be to allow the user to search using a segment of the information sought [Yos94], and use a proximity function to determine the best fit of the retrieved records.

Object DBMSs are uniquely suited to the manipulation of multimedia information, and are becoming the medium of choice for the development of multimedia DBMSs [Adj97]. Using an object-oriented approach, interfaces to manipulate the characteristics of each individual data type may be incorporated into the design of the database itself [Paz97], rather than remaining as separate entities external to the database in the traditional relational database approach [Buc94]. In this way, the details of the implementation of the multimedia data are hidden to the database implementer. Support for new data types and new functionalities may be incorporated without disturbing the utility of existing databases.

2.4 Standards

The design and implementation of any complex system should utilize standardization to the greatest degree possible. The use of appropriate standards ensures interoperability, maximizes reuse, and allows the development of a system which maximizes functionality while minimizing development time.

2.4.1 Medical Informatics Standards

Standards for the definition and management of computer based information are by and large relatively new. Early medical information system implementations often utilized proprietary operating systems, file formats, and communications protocols. The use of standards is highly desirable, as it permits vendor independence and allows systems to be developed independent of specific hardware or software [Liu94] [Buc94]. Inconsistencies between heterogeneous systems lead to conflicts in naming, data structure, value, and accuracy, and issues in translation of data in a meaningful manner [Wil94]. Progress is being made by various standards bodies in the medical information management arena through the development, adaptation, and application of standards which are being targeted specifically to a medical environment. This section will focus on some of the more widely used standards currently being implemented in MIS applications.

The Health Level Seven (HL7) standard applies to textual medical information. Its purpose is to create an interface for data exchange among heterogeneous computer applications that reduces or eliminates custom programming and maintenance [Als94]. The standard was created, and is maintained, by the Health Level 7 Working Group, which is comprised of interested medical and industry participants. It has established liaison with several other standards bodies, including the American Society for Testing and Materials (ASTM) Healthcare Information Standards Board (HISB) and the American College of Radiology/National Electrical Manufacturers Association (ACR/NEMA) Digital Imaging and Communications in Medicine (DICOM) group. The HL7 standard is open and readily available, and is therefore becoming widely accepted as the standard to which textual information system implementations are being designed.

The HL7 standard covers all aspects of textual patient information, including medical records, admission and registration, referral, clinical observations, and patient care. On the other hand, the Digital Imaging Communications in Medicine (DICOM) standard is being developed to address the issues surrounding diagnostic imaging. Developed by a joint ACR/NEMA committee, the DICOM standard is rapidly being adopted as the *de facto* standard for medical image display [Abr98]. It specifies protocols for the interchange of imaging information, commands and semantics for systems that utilize these protocols, file formats, conformance, and networking operation [Nem98].

DICOM is an object-oriented standard. It defines classes which are analogous to real-world entities. In addition to its own image storage format, it also provides conversion facilities between other commonly used image storage and compression formats, thereby increasing its usefulness by permitting information interchange more readily.

The HL7/NEMA Image Management Special Interest Group (IMSIG) has been formed to facilitate information interchange between HL7 and DICOM compliant systems [Ims99]. This type of initiative assists in the standardization process by ensuring that the highest degree of interoperability possible is maintained. The increasing proportion of networking implementations means that interoperability requirements will continue to increase as networking capabilities become inherent to these systems; as an example, networking was not addressed until Version 3.0 of the DICOM standard [Nem98], but now that it has been addressed, maximizing interoperability has become a more pressing issue.

Interoperability of networked object-oriented information systems, particularly those operating under TCP/IP, is also being addressed by the Common Object Request Broker Architecture (CORBA). The CORBA standard is being developed by the Object Management Group (OMG), a consortium of vendors and software developers whose purpose is to assist in the standardization of object-oriented technology [Omg99]. Within this group, the Healthcare Domain Task Force (DTF), also known as CORBAmed, has been formed to provide a public domain standard for healthcare object communications.

2.4.2 Document Presentation Standards

Multimedia document presentation standards include Open Document Architecture (ODA), Standard Generalized Markup Language (SGML), the Multimedia/Hypermedia Experts Group (MHEG) Composite Objects structure, and Hypermedia/Time-based Document Structuring Language (HyTime) [Col93]. However, the use of HyperText Markup Language (HTML) as a basis for multimedia presentation has become a *de facto* standard through the ubiquity of World Wide Web (WWW) browser software, such as Microsoft Internet Explorer and Netscape Navigator. Use of HTML almost guarantees that users will have access to the information through a familiar interface.

2.4.3 Image Storage Format Standards

Opinion is divided among health care professionals on the subject of image storage; more specifically, on whether lossy compression of medical images is acceptable. On one hand, compression of images results in faster transmission times and reduced storage requirements; on the other, lossy compression, while achieving higher compression ratios than lossless, may introduce artifacts which impair the ability of the viewer to utilize the image in diagnosis.

Three measurement components are used to evaluate the degree of distortion experienced in a medical image which has been exposed to a compression/decompression operation: the Signal to Noise Ratio (SNR), which, although objective, is insufficient to completely express the degree of distortion; the subjective quality of the compressed image, expressed on an ordinal scale; and the diagnostic accuracy - the degree to which the image is perceived to be degraded in terms of its usefulness in clinical decision-making [For97]. Lossless compression, by definition, does not normally result in any image quality degradation (except that due to quantization noise), but lower compression ratios are achieved compared to lossy coding. Examples of lossless image coding techniques are Huffman and runlength coding [Gol90]. Lossy coding, such as the Joint Photographic Experts Group (JPEG) standard commonly used, can result in image degradation at high compression ratios.

The JPEG standard has been investigated extensively for medical image applications. Originally intended for full colour still frame applications, and resulting in an average compression ratio of 15:1 [Fuh95], JPEG operates by normalizing the pixel coefficients, subdividing the image into 8x8 or 16x16 pixel blocks, performing a Discrete Cosine Transform (DCT) operation on each block, and Huffman coding the results in a predetermined pattern. While a lossless transform can be performed under JPEG coding, the lossy coding technique is more commonly used; the compression ratio achieved is a function of the storage requirements and degree of acceptable image degradation.

In one clinical trial, JPEG coding was used to compress pathological images with average compression ratios ranging from 20 to 35 without compromising their clinical usefulness; the "just noticeable difference" compression ratio varied from 13.9 to 22.9 [For97]. Other trials reported similar results, indicating a compression ratio of approximately 10 without noticeable image degradation [Ahn94] [Man94].

Detractors maintain that block based coding techniques such as JPEG result in unacceptable blocking artifacts being introduced into the image at high compression ratios [Bre94]. Other compression techniques which do not produce these artifacts have been proposed, including the Discrete Wavelet Transform [Tho97] [Man94], overlapped transform coding [Bre94], and pyramid coding [Gol90].

Audio coding standards are in a state of flux. Digital storage and transmission of audio has traditionally been performed in telephone networks using Pulse Code Modulation (PCM) with 8 bit quantization at a rate of 8000 samples/s, for an overall transmission rate of 64 kbit/s. The need for reduced bit rates in transmission systems has spurred development of other, less bandwidth intensive coding schemes, such as Code Excited Linear Predictive (CELP) coding, which can transmit voice using transmission rates as low as 10 kbit/s. On the other hand, digital audio files stored on computers are frequently stored using proprietary formats such as WAV and Sun's AU format; however, user acceptance of the MP3 standard for digital audio storage is growing.

2.5 MIS Implementations

2.5.1 General

There are several characteristics by which one can classify current DMS implementations for a medical setting. In general, the trend is towards object-oriented, networked systems. At the same time, development seems to be in two parallel streams: integrated multimedia systems, combining text-based patient information systems such as HL7-based systems, and PACS; and image management PACS which deal solely with image information. This section will describe a sample of these implementations to illustrate the range of designs which have been used to achieve the goals of MIS.

2.5.2 Integrated Systems

Early implementations of multimedia MIS typically consisted of a relational database and associated information, with pointers in the database to the multimedia component. An example of this type of system is Iris, which featured an Oracle relational database. Implemented as a client-server architecture operating over an Ethernet network, it used the ODA standard and featured voice and image data storage [Kar90a] [Sud93].

The European Community (EC) has funded several projects in the area of telecommunications. One of these, the BERKOM project, had as one of its goals the development of new services brought about by Broadband ISDN (BISDN) [Fan93]. The Bermed project was developed as one of these new services. Bermed featured an integrated, wide area implementation, designed to improve communications and collaborative work among health care professionals. It implemented two interconnected databases, one textual and the other image, to form a multimedia medical record. It required high bandwidth, however, and was typically implemented across Asynchronous Transfer Mode (ATM) or BISDN communications networks [Kle94]. A third integrated medical record prototype, also of European origin, was the NUCLEUS project. This was an object-oriented system designed to incorporate all patient information into a single multimedia patient record [Kil95].

2.5.3 PACS

A variety of PACS have been implemented in different configurations. One PACS trial in Japan compared the efficacy of two systems: in one, images were directly input from CT and MR imaging equipment to a workstation and from there, transmitted using ISDN to satellite stations; in the second, images were input using a Charge Coupled Device (CCD) camera and again transmitted over an ISDN connection [Ino94]. The two systems represented a tradeoff between quality and cost; in general, it was felt that the lower cost solution was acceptable in many instances. In France, an early PACS implementation featured a relational database of images and an expert system for the interpretation and manipulation of the images. It featured object oriented software design and extensive image manipulation components [Biz93]. A second system implemented in France also featured object oriented design and a relational image database [Abo95].

The Telemed project also originated in Europe. In this instance, a relational database of text information was supplemented by bulk storage of images. Telemed was designed using the client-server model and the Structured Query Language (SQL) for database access [Buc94].

2.5.4 Web Applications

In Europe, one recent system has attempted to integrate Web access and Java applets to create a system that relies less on proprietary hardware, software, and access terminals. Medinet is a pilot teleradiology system that operates in a distributed environment by using the Web. It provides access to medical diagnostic images, a telediagnostic service, and collaborative diagnosis [Abr98]. It has been tested using a variety of network topologies, including Basic Rate Interface (BRI) ISDN, and Ethernet LAN.

Medinet was created "to test the feasibility of a Java/Web-based approach to teleimaging" [Abr98]. While recognizing the necessity of security and privacy, the designers did not implement these aspects of the design at this point, preferring instead to concentrate on an evaluation of Java and the Web based concept. The functionality of the system included medical image access and display, and remote telediagnosis.

In order to test the image download portion of the service, the time to download images using a variety of LAN and MAN configurations were measured. Measurements of download time and the effects of multiple user load were made; it was found that an increase in user load affected the server significantly. In particular, using BRI ISDN, image download time varied from 20 seconds for a 1 Mbyte file to 7 minutes for a 30 Mbyte file when one user accessed the database at a time. Increasing the load to 3 users increased the download time to 10 minutes for a 30 Mbyte file. This type of real-world testing methodology will be used in this thesis to evaluate the network performance of the MIS prototype.

2.6 Summary

We have shown that a need for an integrated, computerized patient record exists in the medical field. It has been shown that the traditional patient record is unacceptable for a variety of reasons. While many computerized systems have been described in the literature, none has been shown to be a comprehensive solution, particularly as it pertains to telemedicine applications.

A telemedicine application requires a system that can be accessed from various locations, particularly remote locations. It should be easy to use, as the user may be geographically remote from a source of technical assistance or guidance; it should not require expensive or difficult to install custom software or hardware, as the user will frequently be a medical practitioner operating either alone or in a small clinic, and financial resources may not exist to support the purchase and maintenance of a capital intensive system; and it should be integrated, ensuring the user is capable of retrieving all necessary information from a single source. In contrast, almost all the solutions described above either address a subset of patient information, or require custom or expensive hardware and/or software, or are limited to a single building or location.

In this thesis a Web based integrated multimedia MIS is proposed as a solution to the problems and shortcomings of the various systems described. The problem lends itself well to a Web based browser solution as a way to overcome the limitations of the solutions described. The intent of this application is to provide a single user interface for all patient information. The next chapter will describe the design and implementation of a prototype of such a system.

Chapter 3

Methodology

3.1 Introduction

3.1.1 General

As described in the previous chapter, most of the development in the field of MIS has concentrated on systems which reside on specific hardware/software platforms and are not generally usable outside the immediate neighbourhood of the installation without a significant investment in equipment, software, and learning. In this chapter, the concept of a Web-based telemedicine system will be further explored. In order to achieve the objective stated in Chapter 1, a "proof of concept" software design of a Web-based multimedia MIS was implemented and tested. The design was implemented in Java using the Visual J++ programming environment. A Web-based trial was tested from two points of view: user testing and performance.

The Web is an ideal environment for the implementation of an information system such as is proposed, provided the constraints of the medium are taken into consideration. Security, available bandwidth, throughput, and QoS are all limiting factors to the usefulness of such an application. On the other hand, if the constraints are properly considered, the benefits include a relatively inexpensive implementation, through the use of commonly available software and hardware; code re-use, through the use of object-oriented design; and a fast user learning curve, through the incorporation of commonly used components and interfaces.

In a telemedicine milieu, the practitioner who requires the information provided by a multimedia MIS will have several requirements which determine to a large degree the form which such a system will take: the user will usually not have time for extensive training; they will be looking for medical information to assist in diagnosis and treatment, rather than epidemiology; they may be performing other tasks at the same time; and they may have a low speed Internet connection over which the data is being transferred. By bearing in mind these factors when the system design is being performed, the resulting software can be optimized to ensure maximum utility for this type of user.

The applicability of this system extends beyond telemedicine applications, however. The use of this system as a hospital information system is equally viable. With the implementation of sophisticated search capabilities, epidemiological studies could be conducted by medical researchers.

3.1.2 Scope

This chapter describes the design, implementation, and testing of QB[1], a prototype multimedia MIS. The software was written and tested to achieve the goals described in the next section.

The software as implemented represents a subset of the final functionality of a working system. Sufficient functionality was implemented to allow users to obtain an understanding of the "look and feel" of the application, and to perform a limited amount of testing, as described later. In this manner, software development time was minimized while allowing a functional prototype to be placed in operation which served the purposes of this thesis.

---

1 In Arthurian legend, the pursuit of the Questing Beast was the lifelong task of King Pellinore, one of the Knights of the Round Table.

3.2 Goals

The objective of this research was to design, implement, and test a rudimentary Web based multimedia medical information system. The system would feature limited search and editing capabilities, and be capable of displaying multimedia information: audio, video, image, and text. The system would then be used as a test environment for the evaluation of the concept and a means of obtaining feedback from medical professionals on the idea. At the conclusion of the research, it was felt that an appreciation of the constraints of such a system being developed in a Web environment would be capable of being determined.

The specific goals of the research were enumerated as follows:

1. To implement a "proof of concept" Web based multimedia MIS.

2. To examine the viability of using the Web for a system of this nature

3. To develop an understanding of the issues surrounding the design of a system of this type.

4. To create a set of performance objectives for a system of this type.

5. To evaluate the potential performance of the system over commonly available access media.

6. To evaluate the suitability of using the Java programming language for the development of a system of this nature.

The "proof of concept" design paradigm was adopted because of the necessity of obtaining input from medical professionals regarding the specific details of the implementation. By developing a rapid prototype which can be evaluated by potential users, design elements which may prove detrimental to the functionality of the final configuration can be avoided. In this way, design time and expense is minimized as the project is able to focus early in the development on its final configuration.

Most Web content is still informational - advertising, promotion, and access to information makes up the content of most publicly accessible Web pages. While corporate "Intranets" have been developed which user browser interfaces for access to corporate computer systems, the concept of using the Web for a professional support system such as this is still relatively uncommon. Issues of security, accuracy, and validity of information must be resolved to the satisfaction of a relatively conservative user group before such a system would gain wide acceptance. The target market is not the public at large, although there is no reason why an implementation of this sort could not be configured to allow a person access to their own medical records, but the intended users would still be widely distributed and not necessarily able to access an Intranet. This system is therefore fairly unique, as it combines the concept of public access with the types of issues currently addressed by limiting physical access to the data.

As this is a unique system, issues are bound to develop during the course of implementation and test which were not considered in the initial design of the system. The proof of concept prototype allows these issues to be addressed early in the design process, saving development time and, in a production environment, development costs. Limitations of the access technology, operating environment, and programming environment may not be apparent to the designers and implementers of the software, but they will often be readily apparent to potential users. This reinforces the concept of early user testing to provide feedback that can be used to modify early design models.

In order to objectively evaluate the performance of a system, it is necessary to state the criteria by which the system will be judged. Ideally, the performance criteria will provide a baseline against which the functionality of the system can be judged, and will allow successive implementations to be compared to establish incremental performance improvements. Server performance, access time, intuitive user interface design, and ease of installation and maintenance are all issues which affect the acceptance of the software in its potential market, and all must be addressed in order to ensure the system possesses maximum utility.

A working system of this type would, as previously discussed, almost certainly be implemented in a WAN topology. Current low cost Internet access methods used by individuals and small businesses to connect a single computer or small network to an Internet Service Provider (ISP) include dial up service (14.4 kbit/s to 56 Kbit/s), cable modem (4 Mbit/s shared Ethernet), and Asymmetrical Digital Subscriber Line (512 Kbit/s upstream, and 128 Kbit/s downstream). Of the three access methods, users in more rural areas are most likely to have access to dialup service. Cable modem and ADSL technology are new services, and are being installed in more densely populated areas first. As they gain widespread acceptance, they will move to less densely populated areas; however, penetration of these technologies will not take place in remote areas for quite some time, and in many instances probably not at all. Given the limits on access technology, it is imperative that a reasonable set of performance expectations be established to assist in delineating the geographical boundaries of the anticipated service area for this system.

Finally, the usefulness of Java as a programming environment in this instance has not been verified. The unique properties and benefits of Java in general have already been discussed, but extensive use of Java to provide access to a large, dynamic database in an Internet medical application has not been demonstrated in the literature. By implementing and testing a limited application in Java, a better appreciation of its strengths and limitations will be obtained. The implementation of the prototype in Java will enable the future design of the software to be refined to capitalize on the strengths and minimize the effect of the weaknesses of the programming language.

3.3 Software Design

3.3.1 Objectives

In order to achieve the goals described above, it was felt that it would be necessary to develop a limited software application to use as a test environment. The software would implement the "look and feel" of a MIS, using a limited amount of strictly fictional user data. Conceptually, the software design would reflect the objectives stated as follows:

1. To design a "proof of concept" Web based multimedia MIS.

2. To design a test environment for user testing and performance testing.

3.3.2 Design Paradigm

Upon beginning the design of the software, the software design paradigm was the first consideration. In this sense, the paradigm refers to the method by which the design will proceed. Two design paradigms were considered - the traditional "waterfall" or linear approach, and the recursive "spiral" approach.

Under the waterfall paradigm, software design is considered a linear process, as shown in Figure 3. Upon establishment of the system requirements and software requirements, analysis and design of the software proceed in a linear fashion, followed by coding, testing, and operation [Sod96]. This paradigm does not conform well to object-oriented design, but rather it favours traditional structured programming. The spiral model provides for a great degree of feedback during the design process. Initial design models result in successively more refined design prototypes as the constraints and

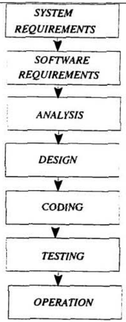


Figure 3. Simplified Waterfall Design Paradigm (from [Sod96]).

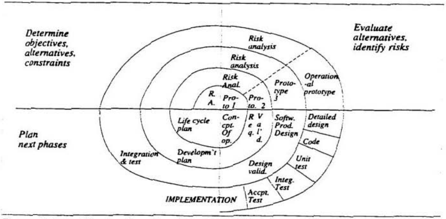


Figure 4. Spiral Design Paradigm (from [Sod96]).

requirements of the system become evident. Only when significant effort has been put into the design and testing of the software does anything resembling a working prototype evolve. The spiral paradigm is a recursive process; it permits later versions of the software to incorporate design elements based on the feedback from previous versions [Sod96]. Figure 4 illustrates the spiral model.

The spiral model is well suited for object oriented design. The implementation of successive prototypes is made more practical because of the object-oriented design philosophies of modularity, information hiding and code re-use. These concepts reduce the amount of code that must be redeveloped in successive prototypes.

A modified spiral model has been used in this instance to design the software. Initial broad design specifications were written to permit rapid prototyping to take place. In this way, software was implemented quickly which could be used to obtain feedback from a subset of the user community. In this instance, it was felt that this method would allow greater understanding of the design constraints while the code base was still relatively small, resulting in a more efficient design process.

3.3.3 Design Principles

In designing the system, several overriding design principles were established at the start of the design process. It was felt that the inclusion of these principles was relatively obvious or at least intuitive.

The first principle considered was object-oriented design. It was felt that the choice of object-oriented design as the programming paradigm was the logical choice given the status of programming languages currently used in application development. Traditional structured programming languages such as Basic, Pascal and C have been supplanted by their object-oriented successors; in the cases of languages mentioned, Visual Basic, Delphi or Object Pascal, and C++ respectively.

Given the choice of an object-oriented implementation, it was decided that the application would be written as a Java applet. The great advantage of Java as a programming language in this instance is its "write once, use anywhere" philosophy. Use of Java meant that the applet would run under a relatively new version of almost any Web browser without the need for specific hardware or software on the part of the user.

In concert with the decision to develop the application using Java was the decision to implement the application using the Web as the access technology. Using the Web for an application of this nature is a relatively recent development, but it was felt that the benefits outweighed the disadvantages. It quickly became apparent that Web access is a fundamental parameter which in many cases acted as a deciding factor in other design issues.

3.3.4 Specifications

In order to begin the software design, preliminary specifications were developed. The preliminary design documents are contained in Appendix A of this thesis. They consist of a System Definition and a high level Design Specification; both documents were written using a modified version of the formats outlined in [Fai85].

The first design specification to be created was the System Definition. In this document, the high-level specifications of the software are spelled out. The problem definition, system justification, goals, constraints, and functions, user characteristics, development and operation environments, and acceptance criteria are described. The document is a first attempt to quantify the requirements of the overall system, and as such it describes many functions and goals which were not implemented in the first prototype.

The Design Specification deals more directly with Version 1.0, the first prototype of the software. It describes in more detail the functionality of the prototype, and gives flowcharts, object definitions, and screen dumps to give an indication of the appearance of the user interface. Overall, it is a useful outline to assist in prototyping the software.

Design documents such as these are essential when software implementation is being performed by a team, rather than a single designer. Despite this, they still serve a useful function for the individual programmer as well. By using the System Design and Design Specification documents to assist in software implementation, the overall objectives of the implementation can be preserved. The design documents function as a mission statement, focusing the software designer on the objective of the design rather than the specifics of the implementation.

3.3.5 User Interface

Almost all software being developed for the commercial market today features a GUI. As such, the development of a GUI for this application was almost a foregone conclusion. However, a wide range of interface quality is seen within the sphere of the GUI, ranging from the excellent to the unintelligible. In order for this system to gain wide acceptance, it was felt that the GUI design should be an important part of the overall software design process.

The first consideration in the GUI design was clarity and simplicity. If this system was introduced in a working environment, at times the user would be facing considerable stress brought about by the conditions under which they were working at the time - for instance, a nurse may be using the system to check the records of a patient who has apparently had a reaction to a particular medicine, and who may in fact be allergic to that treatment. In this instance, clarity and ease of use may be the deciding factors in the time it takes to identify and treat the condition. Even in typical everyday applications, a user interface whose design is simple and easy to use will be preferred by most users over one which is complicated or cumbersome.

The user interface design objectives described above assist in clarifying the most appropriate type of user interface for this application. In order to achieve these objectives, a "function key" interface was decided on. Under this paradigm, functions are invoked by the user by pressing soft keys [May92]. In the WIMP interface, the mouse-invoked button is analogous to a soft key in the traditional text-based screen interface world. The function key interface was chosen as it was felt that it suited the characteristics of the typical medical user - a person who is not necessarily familiar with computers, who is typically not able to spend a great deal of time learning how to use the application, and who needs the information presented in a clear, unambiguous manner.

3.3.6 Object Definition

The specification of the primary objects or classes is a fundamental step in the process of OOD. In this document, in order to minimize confusion, the following convention will be adopted: a **class** refers to the definition of a fundamental unit in an OO application, and an **object** refers to a specific instance of that unit.

A multitude of semantic models have been developed to symbolically represent the definition of classes and their relationships with other classes. In this document the graphical notation will be eschewed, as it requires the reader to learn the semantic structure of the graphical notation and does not add greatly to the understanding of the design in this instance. The classes that are defined are relatively simple and their relationship will be made clear through the use of text to reveal association. Class descriptions will start with those whose structure is composed solely of fundamental units of the programming language and proceed from there. Complete descriptions of each class will be found in Appendix B.

The fundamental class of the software design is the **Record**. Recalling the patient file described earlier, the fundamental unit of this file is a single piece of information, whether in a textual, audio, image, or video format. Thus the software analog of the Record - a fundamental unit which has informational properties, such as the date of creation, institution and practitioner by which it was created, the file type, and finally the content itself. The Record can be viewed (more correctly, perhaps, would be the term "accessed", since the information may not be visual in nature), modified, instantiated, and deleted. Once the definition of the Record is established, subclasses for each defined information type are defined so that implementations of the methods specific to the media type can be developed.

The next fundamental class is the **PatientFile**. Considering an institution's patient information as the analog of the PatientFile, it is obvious that the fundamental units of information will include first, middle, and last name, date of birth, family doctor, and identification number. Optional units of information may include health insurance information, religion, and other personal information which may be relevant. The PatientFile will also contain a series of Records, containing all information regarding the treatment of the patient.

When using Java applets, the base class of the applet is always a class which extends the class Applet defined in the Java environment. It is this base class which is invoked in the HTML code for the applet's Web page. In this instance the base class is **QB**, which is also the name of the HTML file in which the applet is contained; this is not necessarily the case.

A series of objects are created to interact with the user. The method used by Visual J++ is to create a base class for each window which is then extended by the user; the superclass is then instantiated and manipulated to perform the functions required. In this manner the base class may be manipulated (by changing the size, position or quantity of controls) without affecting the implementation.

3.3.7 Object Oriented Database

Several object-oriented database tools have been developed recently and are available as commercial products. These tools supplant relational databases in an object-oriented milieu. In the case of this prototype, however, it was felt that this type of tool was not necessary. Instead, an object-oriented database was simulated by superclassing the PatientFile and Record objects discussed earlier with the Java Vector class.

The Vector class enables the programmer to define a linked list of objects. It provides methods for search, indexing, adding to, and deleting from the list. Because of the limited capabilities being implemented in the prototype, it was decided that this method of database design would lead to the functionality desired while maintaining simplicity in the code. Nothing in the design of the system using the Vector class would prevent the application from utilizing a more sophisticated database in the future.

3.3.8 Response Time

System response time considerations for a Web application differ from a typical desktop application. Users are becoming familiar with the idea that time to load an application is an acceptable tradeoff when the benefits include a rich information environment once the application download is complete. The concept of the "World Wide Wait" is being supplanted by the idea that a certain amount of waiting is a necessary evil, and that the benefits of waiting outweigh the nuisance factor in many cases. Recall that the traditional patient record is stored in a file folder in a central record archive; access to this record may be a matter of hours or days. When weighed against this, the user will probably be tolerant of a certain amount of transmission delay.

Response time is made up of several components: transmit delay, the delay involved in transmitting and receiving across a communications medium which has a finite transmission speed; server delay, the delay during which the server processes a request, and client delay, during which the client receives, processes, and displays the information retrieved. Transmit delay can be further reduced to two components: transmission facility delay, which will be unchanging for a given facility, and router delay, which will vary depending on the load presented to a router.

There is a great deal of motivation to quantify a maximum acceptable waiting time. While the two second screen update rule specified by designers of text screens

[May92] does not apply here, there is still a maximum acceptable waiting time after which the usefulness of the application begins to degrade. The benefits of an information-rich screen are outweighed if the user could obtain the information more quickly by physically accessing a paper file, for instance.

A search of the relevant literature revealed no download time benchmarks for this type of application. Because the acceptable response time for access to a data file in a Web environment is not currently known, an assumption was made that ten seconds would be an acceptable maximum waiting time for a download of one piece of information. Any longer and it was felt that an unnecessary burden would be placed on the user.

3.4 Implementation

3.4.1 General

The software was written using the Microsoft Visual J++ Professional Edition development environment on an Intel Pentium II 300 MHZ Windows 95 platform. Code for the Web page was written in HTML using a simple text editor.

In total approximately 2000 lines of code were required for the prototype. Of this, windows which were designed using Visual J++'s Resource Editor, which automatically generates source code for layouts specified by the programmer, account for approximately 1200 lines. The remaining 800 lines of code are used to manage the display of patient information.

In developing software using a Microsoft product, care was taken to avoid the use of components which were platform specific to the Microsoft operating system. Microsoft has chosen to supersede the Java specification by providing Application Programming Interfaces (APIs) to components which are specific to Microsoft Windows platforms. Use of these APIs would prevent the application from being used on other operating systems such as Unix or Linux. As one of the key advantages of Java is its portability, use of the Microsoft APIs would defeat this purpose.

3.4.2 User Interface

Figure 5 shows the HTML Web page which greets the user upon accessing the URL of the application. It is designed to be an informational page to let the user know what has been accessed while the Java applet is loading. Once the applet has been loaded, the user will be presented with the window shown in Figure 6. This is the main window to access patient records in the application. In future releases, the user would be permitted to search by name, ID number, or conduct a more advanced search using a query language interface.

Right now, once the patient has been selected by name, the main patient information window shown in Figure 7 is displayed. This window contains the patient's vital statistics (date of birth, full name, etc.), as well as lists of specific records of each modality currently available in the prototype - Audio, Text, and Image. Buttons allow the user to view a specific file selected from the drop-down list, or in future versions, to perform a search of the records of a specific type.

Selecting a record of a specific type allows a window of the type shown in Figures 8, 9, or 11 to be displayed. This contains text information associated with the specific file being accessed, such as creation date and location, practitioner, and so on. As can be seen from the figures, in an effort to maintain consistency in the application, the format of these windows differs only in the display button located in the lower left hand corner, whose label changes depending on the information modality being displayed. This provides a contextual cue to the user which serves to remind them of the record type being considered at a particular instant.

When the record under consideration is Audio, pressing the "Listen to Audio" button shown in Figure 8 will result in the Audio file being downloaded and played. No visual cue is provided at this time to indicate file download status. Depending on the browser used, a status bar may be displayed showing the status. In the case of an Image record, pressing the "Display Image" button will cause a window to open displaying the image, as shown in Figure 12. Pressing the "Display Text" button will likewise open a window containing the associated text as shown in Figure 10.

As implemented, the user interface is simple, unambiguous, and easy to use. The controls are grouped by function to assist in user comprehension.

3.4.3 File Formats

The format in which information associated with the application was stored was largely determined by the limitations of the programming language. By using the multimedia capabilities of Java in developing the prototype, restrictions were placed on the file formats used. The current version of Java is only capable of handling multimedia information in limited formats.

Using Java's AudioClip object, audio must be stored using Sun's proprietary 8 KHz sampled, μ-law encoded AU format [Mor97]. This limits the bandwidth of the audio to 4 KHz, restricting the usability of the audio in applications requiring high audio fidelity. In the case of this application, AU format will most likely be sufficient, as most audio information stored will be voice and will not require high bandwidth.

Java also is only capable of displaying images in JPEG or GIF format, unless a separate Application Programming Interface is used in conjunction with the Visual J++

environment. In this application JPEG format is used exclusively for the storage and display of images. The JPEG format was chosen because its use has been researched in a medical context, and many test images were available already stored in JPEG format.

3.4.4 Video Playback and Retrieval

The ability to retrieve and playback video would be a desirable feature of the prototype. In practice, video playback is not a function of the Java programming environment at this point. Video integration requires the developer to utilize Java Media Framework Application Programming Interface, which is separate from the compiler environment.

Because a functioning prototype which accepted audio and image could be implemented using components present in the Visual J++ environment, video was not incorporated into the prototype at this time. Integration of a component from a separate manufacturer was anticipated to be a lengthy and time-consuming task owing to the probable learning curve involved. With this in mind, it was decided not to include video information, since the concept of an application which incorporates video is not a novelty to most computer users, including those who would later test this prototype. In addition, large files of other media types could be created in order to evaluate downloading time. It was felt that the implementation of audio and image, both of which could be programmed within the Visual J++ environment, would suffice to allow users to understand the concept of what was being proposed.

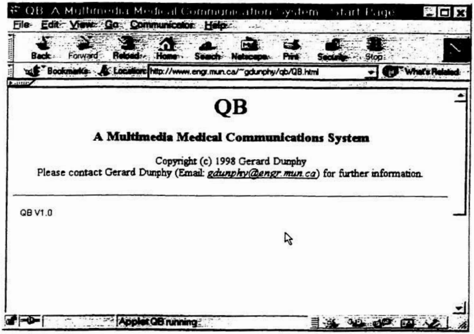


Figure 5. Initial HTML page for QB.

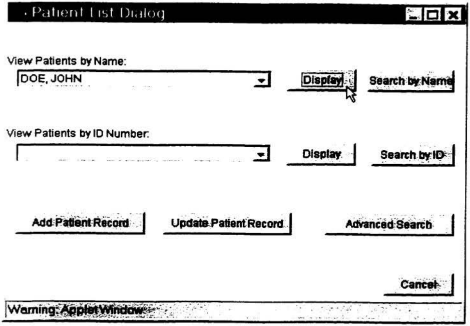


Figure 6. Main Menu.

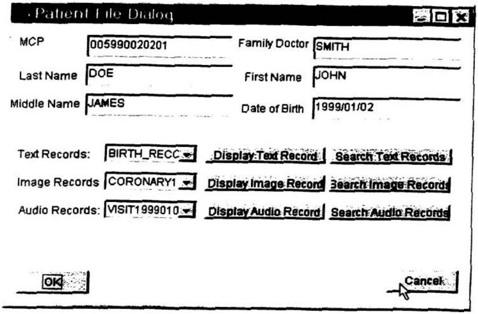


Figure 7. Patient Record Display window.

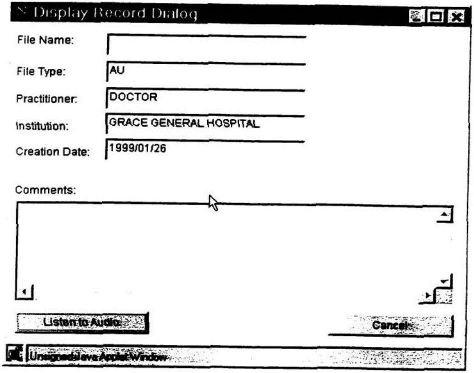


Figure 8. Audio Record Information Display window.

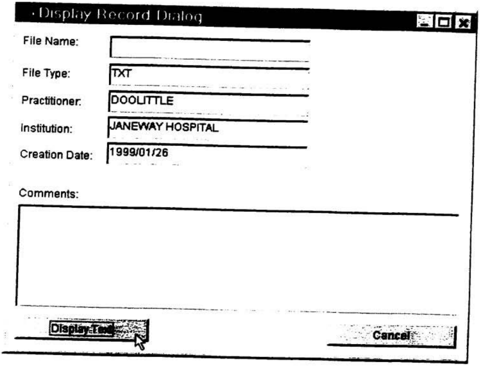


Figure 9. Text Record Display window.

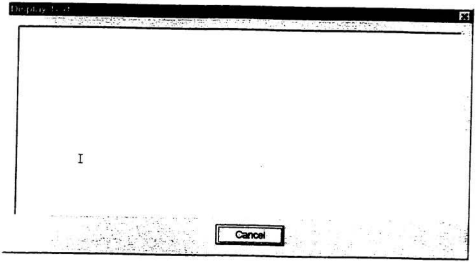


Figure 10. Text Display window.

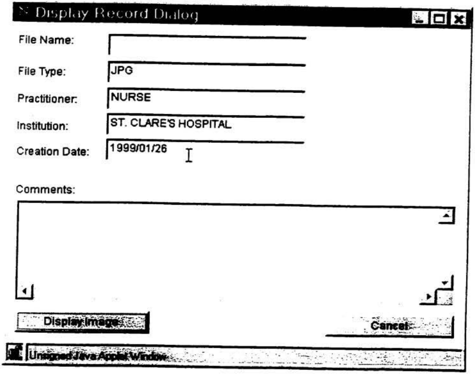


Figure 11. Display Image Record window.

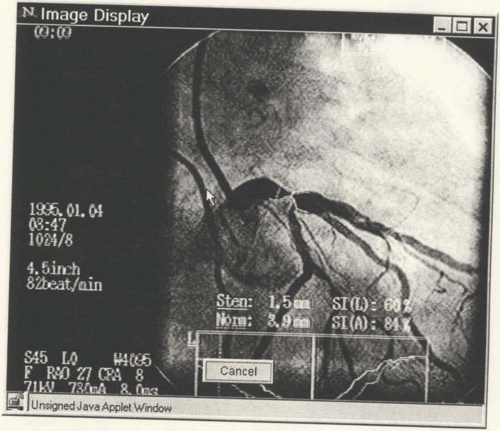


Figure 12. Display Image window.

3.5 Testing

3.5.1 General

Evaluation of the prototype consisted of two parts: user testing, during which the software was evaluated by potential users, and delay testing, during which the performance of the system over a variety of common access media was measured. The result of the testing is anticipated to be a better understanding of the requirements of the software in terms of design and implementation.

3.5.2 User Testing

In order to perform user testing, three health care professionals were contacted and asked to participate. The participants were a radiologist, a neonatologist, and a genetics counsellor. The subjects were chosen based on the fact that they reflected a variety of backgrounds and interests and would provide a range of interpretations and impressions on the application.

User testing is a critical part of a successful software design; its purpose is to gauge the response of potential users to the system. User test methods include the informal interview, questionnaire, observation of computer-user interaction, recording, and simulation [Tre94]. In this instance, the informal interview and questionnaire were used to record user feedback. The advantage of early prototype user testing when using the spiral model as the design paradigm is that the interface can reflect input from the target audience while changes are still relatively inexpensive (in terms of development time) and relatively easy to implement.

User testing consisted of a short interview with the test subject, a demonstration of the application, and a conversation during which the subject's impressions of the application were recorded. The subject was then given a questionnaire which led them through several specific tasks and recorded their impressions. A copy of the questionnaire is contained in Appendix C; the responses of the users are summarized in Chapter 4.

3.5.3 Delay Testing

In order for this system to function in a telemedicine application, it is important that the constraints imposed by Internet access technologies be quantified as closely as possible. Internet access for a single user such as a general practitioner located in a small community would normally be provided through a commercial ISP. Perhaps the single most important factor in determining the usability of the system in an operational setting is the time spent waiting for records to be downloaded to the client from the server.

Because of the many factors influencing the speed of an application running over an Internet TCP/IP connection, it was decided that in this instance the most practical type of testing given the facilities available would be to measure the effects of transmission delay on the application. In investigating transmission delay, an attempt would be made to distinguish between fixed delay caused by transmission facilities and router-induced variable delay. Delay testing would be limited to the time one might expect typical medical files to download, as it was felt that this would be the primary source of user interaction once a user had gained access to the application.

Router delay is variable, as it is a function of the traffic presented to the router. As a router receives IP packets, it must examine the IP header to determine the destination of the packet. As traffic presented to the router increases, packets may be buffered until the router has time to perform this function. It is this buffering which is the source of the variable delay.

If IP traffic is examined by time of day, the effects of router delay may be determined. When traffic is at its lightest, in the early morning hours, then router delay will be minimized. As traffic picks up, delay will increase as the router becomes more active. The difference in delay for a given route can be attributed to the effects of buffering in the router.

While transmission delay is fixed for a given communications facility, there are currently several access technologies by which users may access ISPs. Cable modems, operating using a local, shared 4 Mbit/s Ethernet service, Asymmetrical Digital Subscriber Line (ADSL), operating at variable speeds, but most commonly at 512 kbit/s in the downstream and 128 kbit/s in the upstream, and dial up modem, operating at speeds ranging from 9600 bits/s to 56 kbit/s are commonly used.

In order to examine the effects of time of day variability in delay, a test was set up to ping the host on which the application was running. The test was performed once every 30 minutes, 24 hours a day, for several days, using a variety of communications media. The ping command was followed by a trace route (tracert) command, which was used to verify that changes in delay were not caused by a change in routing.

In order to correlate delay with loading time of files in the application, several tests were performed using each test access medium. A large (650 Kbyte) audio file was created and the download time measured using a PC based stopwatch. This method was made necessary by the restriction on access to the local file system imposed by Java; the applet would have had to be trusted in order to gain access to the local file system. The applet was therefore incapable of writing to the local hard drive. The audio format was chosen because the file is downloaded and then played using two consecutive lines of code:

```
AudioClip myClip =                    //Instantiate an audio object and
      getAudioClip(fileURL);          //retrieve the file
myClip.play();                        //Play the audio
```

Figure 13. Java Source Code to Download and Play an Audio File.

By using the audio file as a measure of download time the effects of processing the application on the local computer were minimized.

A variety of test media were used to measure delay. Tests were performed using 33.6 kbit/s dialup line, simulated BRI ISDN, ADSL, and Thinnet (coax) Ethernet. While a test of cable modem would have been desirable, circumstances conspired to prevent the inclusion of that particular technology.

3.6 Summary

In this chapter, the design and implementation of a multimedia MIS which operates using the WWW has been presented. It has been shown that the a prototype which permits the demonstration of the functionality of an integrated multimedia MIS is feasible using the functionality of the Java programming language.

In the next chapter, the testing performed using the prototype will be presented. The testing was performed in two stages: in the first stage, the prototype was used to permit medical personnel to provide input on the viability of the application, and to provide suggestions for the design of the system. In the second stage, aspects of the performance of the system will be examined. By measuring download time using various types of access technology, an understanding of the ability of a system of this nature to operate in a telemedicine application will be obtained.

Chapter 4

Results and Discussion

4.1 General

The design and implementation of the test software has been previously discussed. It has been shown that there exists a potential niche for a Web based multimedia patient information system. This chapter contains the results of testing which was performed using the prototype in an effort to better understand the constraints placed on the application by the chosen design configuration.

In this chapter the results of user testing and delay testing are presented. The responses of each user polled to the interview and questionnaire will be discussed in detail, both individually and collectively, and the results presented. The delay testing methodology will then be examined, and the results of the tests discussed for each medium tested. The chapter will conclude with a summary of the results.

4.2 Results

4.2.1 User Testing

The purpose of the user testing was to determine the viability of the concept and to obtain input from medical professionals regarding the functionality required from the application. In conducting the user testing, candidates were chosen to provide a range of backgrounds and exposure to computerized record systems. One test subject is a neonatologist with relatively extensive exposure to computerized medical systems, and as such, was thought to be an ideal first candidate to evaluate the application from both a medical and technical perspective. Another is a genetics counsellor with relatively limited exposure to computerized record systems; it was felt that this individual's perspective on the application would be untainted by prior exposure to other systems. The other test subject is a radiologist with considerable exposure to PACS systems and a requirement for extensive interaction with medical images.

4.2.1.1 Questionnaire

A questionnaire was administered, which was designed to elicit written responses from the test users [Tre94]. The questionnaire was designed to gauge the reactions of the users to performing specific activities using the prototype. The response of the users to the prototype was then used to evaluate the usefulness and completeness of the user interface.

The questionnaire consisted of two types of questions: simple yes/no or one word response questions, and questions which required a longer written response from the user. Tables 2 through 6 contain the results of the response to the yes/no questions for each individual; the written response questions will be discussed separately.

Question	Response		
	User A	User B	User C
Is the user interface easy to understand?	Yes	Yes	Yes
Is this application a useful concept?	Yes	Yes	Yes
Is the user interface intuitive?	No	Yes	Yes
Would this type of application be useful in your daily routine?	Yes	Yes	No
Do you think the ability to display video (e.g. ultrasound) would be useful in this application?	Yes	Yes	Yes
Do you think the ability to conduct consultation with another professional would be useful, e.g. videoconferencing?	Yes	Yes	Yes

Table 2. Questionnaire Response - Overall Impressions of the Application.

Question	Response		
	User A	User B	User C
Did the window take a long time to load?	No	No	Yes
Did you consider this too long?	No	No	No
Approximately how long would you consider too long?	5 sec.	2 min.	Min.
Is the function of this screen apparent?	Yes	Yes	Yes
Are the functions of the controls clearly understandable?	Yes	No	Yes

Table 3. Questionnaire Response - Task 1 - Loading the Program.

Question	Response		
	User A	User B	User C
Did the window take a long time to load?	No	No	No
Did you consider this too long?	No	No	No
Approximately how long would you consider too long?	5 sec.	1 min.	Min.
Is the function of this screen apparent?	Yes	Yes	Yes
Are the functions of the controls clearly understandable?	Yes	Yes	Yes
Is there information that should be on this screen but is not?	Yes	Yes	No
Is there redundant information here?	No	No	No

Table 4. Questionnaire Response - Task 2 - Displaying a Patient's Record.

Question	Response		
	User A	User B	User C
Did the window take a long time to load?	No	No	No
Did you consider this too long?	No	No	No
Approximately how long would you consider too long?	10 sec.	1 min.	Min.
Is the function of the "Display Record" window apparent?	No	Yes	Yes
Are the functions of the controls clearly understandable?	Yes	Yes	Yes
Is there information that should be on either window but is not?	Yes	Yes	No
Is there redundant information here?	No	No	No

Table 5. Questionnaire Response - Task 3 - Retrieving an Image.

Question	Response		
	User A	User B	User C
Is the ability to listen to audio (e.g. voice recordings of diagnosis, recordings of heartbeats, etc.) useful?	Yes	Yes	Yes
Is there information that should be in this window but is not?	Yes	Yes	No
Is there redundant information here?	No	No	No

Table 6. Questionnaire Response - Task 4 - Listening to Audio.

In addition to the simple yes/no response questions posed in the questionnaire, the users were asked to relate more detail on specific aspects of the user interface, either in conjunction with a yes/no response, or with a separate, specific question. This method permitted the respondent to elaborate on their response to the yes/no questions, and to respond more fully with details of the features and functionality they feel such a system should contain.

In response to the question "Would this type of application be useful in your daily routine", User A and User C declined to respond with more detail. User B explained that the application "would be a wonderful tool for doctors" and would promote "better understanding and education about the findings on x-ray/nuclear medicine scans".

The next questions related to functionality which was contemplated for an integrated patient record system but not implemented in this prototype. The question "Do you think that the ability to display video (e.g. ultrasound) would be useful in this application?" elicited a response from all users. User A indicated that "even short [video] strips to demonstrate images during audio would help", User B replied that "the doctor ordering the test would be able to see exactly what the radiologist was describing - could also be used to show the patient exactly what was wrong - patient might be more compliant to treatment if they could see what was wrong". User C responded that "a full range of imaging should be included". The question "Do you think the ability to conduct consultation with another professional would be useful, e.g. videoconferencing?" also met with a positive response. User A declined to elaborate; however, User B indicated that "Family doctors and specialists could review what they were seeing and explain and question certain aspects - this kind of consultation is often done ... when neither of them is looking at the x-ray etc.". User C responded that "one of the most useful roles for this software would be remote consultation".

In response to the request to describe other functions a system of this nature should possess, User A indicated that "the interface needs to allow you to visualize the [Table of Contents] for a whole patient chart by time and type". User B replied that confidentiality and security ("limited users...update or edit") were requirements. User C indicated that "comparison of studies from several institutions/sites" would be useful.

The next series of questions were in response to having started the program and viewed the introductory window shown in Figure 6. In response to the question "Is the function of this screen apparent", all users responded positively but declined to elaborate further. In response to the question "Are the functions of the controls clearly understandable", User A and User C gave no further comment; however, User B replied with a query about the "function of the advanced search" as well as showing confusion regarding the "update patient record" functionality.

The second task to be performed in the user test was to display a patient record, causing the window shown in Figure 7 to be displayed. In response to the question "Is the function of this screen apparent", User A and User C did not respond, but User B responded that it is "obvious that all text, image and audio records of patient should be accessible from here".

Responding to the question "Is there information that should be on this screen but is not". User A replied that it requires "an ability to view records chronologically relating text, image and audio to one another, all on one screen for an individual patient". User B stated that it must have "demographics - address, next of kin, phone numbers, etc.", and that "the text records may have to have subdirectories - laboratory, pathology, discharge summaries, etc. - maybe image records as well - MRIs, etc.". User C did not respond.

The third task was to retrieve and display an image, thereby viewing the windows shown in Figures 11 and 12. In response to the question "Is the function of the Display Record window apparent", User A was unsure if "the comments [section is] for me or do they accompany the record", and neither User B nor User C responded. The question "Is there information which should be on this window but is not" was met with a response from all users. User A indicated that it should show a description of the contents of the file, and should display search keywords, as well as an icon indicating the information type; User B indicated that "the patient name and MCP should be displayed on every page, [as] I forget whose record I'm looking at occasionally"; and User C did not respond. In response to the question "Is there redundant information here", User A did not elaborate, User B expressed confusion regarding the file name and file type shown, and User C did not respond.

The final task was to access and listen to an audio record. Because the information screen is essentially the same as that for an image, questions concentrated on the utility of including audio records in the application. The question "Is the ability to listen to audio useful", User A indicated that the record "could contain dictated comments", User B suggested that the ability to store and retrieve heartbeats could be used as a learning tool, and that "audio report of pathology or a specialist's report might be available more readily than text, so reports might be able to be retrieved earlier", and User C did not respond. In response to the question "Is there information which should be in the window but is not", User A stated that it must have the "duration of the message and the ability to stop audio", User B indicated that as before it should include the patient's name and MCP, and User C indicated that it could contain audio records such as "coughs [and] respiratory sounds".

4.2.1.2 Interviews

Informal interviews were used to assist in gauging the opinions of the candidates used in the user testing. The users were queried regarding their overall opinion of the applicability of the Web-accessible concept, the relevance of the integrated patient record concept, and their opinions regarding the prototype and its user interface design. Responses were recorded in point form and summarized herein.

User A was of the opinion that this type of application would be ideal in a telemedicine setting. Several enhancements were mentioned, including voice control of the system for instances when a medical practitioner was unable or unwilling to touch a keyboard (e.g. after washing up for a procedure), a timeline showing a patient's records in chronological order, the ability for a practitioner to automatically retrieve relevant records upon logging into the system (e.g. the list of their patients for the past 2 weeks), the ability to scroll through and bookmark selected records for later display, and content based retrieval of non-text data. The application would also have to feature extensive search capabilities in order to be truly useful in day to day operation.

User B also stated that the application was a practical one. One aspect of the system design not previously considered that was brought out in this interview was the issue of integration with existing text databases, particularly when considering the initial implementation of such a system. An extensive text-based computerized record keeping system already exists in Newfoundland, and presumably this would not be dispensed with overnight. Implementation of a multimedia system such as this would require a coordinated effort to ensure that continuity would be maintained.

User C was of the opinion that in general medical information systems will move to a WWW interface for telemedicine applications. Already vendors have started to move to a browser interface in newer versions of PACS. In addition, digital record storage technologies are well established in the field of radiology. Traditional x-ray film suppliers such as AGFA, recognizing that their established markets are eroding, are beginning to implement computerized solutions to replace film. Digital voice recording technology is also being implemented to replace traditional tape recording. Other initiatives in Newfoundland include a province wide PACS implementation being planned, and a filmless x-ray department being implemented in Gander in the near future. In general, these initiatives point to the increasing need for integrated digital storage and display.

4.2.2 Delay Testing

Delay testing was conducted in two phases: round trip delay testing, and download time measurement. In the round trip delay test, the server on which the application resides was "pinged" once every 30 minutes for several days using a variety

```
echo off                                // stop from echoing commands to display
date < cr.dat >>ping.txt                // append the current date to the file ping.txt
time < cr.dat >>ping.txt                // append the current time also
ping nano.engr.mun.ca >>ping.txt        // append the output from the ping command
tracert nano.engr.mun.ca >>ping.txt     // and the tracert command
time < cr.dat >>ping.txt                // append the current time
```

Figure 14. Ping.bat Batch File.

of access methods - ADSL, simulated BRI ISDN, dialup modem, and Ethernet. The round trip delay measurements of the ping were tabulated and plotted to determine average delay by time of day. The download time measurement attempted to correlate round trip delay to measured download time. In this test, the server was pinged repeatedly in succession to obtain an average delay measurement, and immediately afterwards, the download time of a large data file was measured repeatedly. The testing was performed using the same access media as use in the round trip delay test - dialup modem, simulated BRI ISDN, and ADSL. In order to provide a control measurement, the same tests were performed while the client was directly connected to an Ethernet LAN on the same network as the server on which the application resided.

The purpose of the round trip delay testing was twofold: to determine how delay could vary with time of day in a real world simulation of a working application, and to determine the suitability of various ISP access technologies which might be used to access the application in a telemedicine application. The download time test would then show how the actual download time corresponded to the theoretical time. The intent is to quantify some of the parameters under which the application could be realistically expected to perform.

The round trip delay test was performed by scheduling the client computer to execute a batch file every 30 minutes, 24 hours a day, for the duration of the test. The batch file commands are shown in Figure 14. The **ping** command sends a 32 byte packet to the remote IP address specified. The packet is echoed at the remote address, and the round trip delay measured. The tracert command traces the routing from the client to the remote IP address indicated. Example output from the batch file is shown in Figure 15. The results were tabulated and the average delay by time of day was plotted, separating weekday from weekend results.

The download testing was performed by sending a ping packet to the host 20 times in succession, and averaging the result. Immediately afterwards, an audio file was downloaded 10 times in succession, and the download time averaged. The theoretical delay time was then calculated based on the measured round trip delay, and the results plotted.

All delay testing was performed using a Pentium II, 300 MHZ PC operating Windows 95. Networking access was provided using the drivers supplied with Windows 95. Netscape 4.5 was the browser used for net access.

Because of the limited types of test access technologies available, a BRI ISDN connection was simulated by limiting the throughput on the ADSL modem to 128 kbit/s in both directions, thereby providing a connection speed comparable to the use of two B channels of an ISDN line.

```
Current date is Sat 03-27-1999
Enter new date (mm-dd-yy):
Current time is 10:30:00.10a
Enter new time:

Pinging nano.engr.mun.ca [134.153.12.79] with 32 bytes of data:

Reply from 134.153.12.79: bytes=32 time=114ms TTL=243
Reply from 134.153.12.79: bytes=32 time=105ms TTL=243
Reply from 134.153.12.79: bytes=32 time=101ms TTL=243
Reply from 134.153.12.79: bytes=32 time=106ms TTL=243

Ping statistics for 134.153.12.79:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 101ms, Maximum = 114ms, Average = 106ms

Tracing route to nano.engr.mun.ca [134.153.12.79]
over a maximum of 30 hops:

  1    17 ms    17 ms    16 ms  18warp124.newtel.com [142.163.124.18]
  2    18 ms    16 ms    17 ms  209.128.2.5
  3    20 ms    19 ms    20 ms  1block00.newcomm.net [209.128.0.1]
  4    52 ms    52 ms    55 ms  216.208.252.33
  5    98 ms    98 ms   100 ms  bxnfl-atm2-0.stjohns.canet.ca [205.207.238.70]
  6   100 ms    99 ms    98 ms  gw-nlnet.stjohns.canet.ca [192.68.60.101]
  7   100 ms   105 ms   102 ms  198.165.241.1
  8   102 ms   102 ms   107 ms  198.165.251.246
  9   106 ms   105 ms   113 ms  198.165.251.250
 10   103 ms   107 ms   108 ms  198.165.251.6
 11   105 ms   104 ms   116 ms  nano.engr.mun.ca [134.153.12.79]
 12   110 ms   106 ms   108 ms  nano.engr.mun.ca [134.153.12.79]
 13   115 ms   121 ms   113 ms  nano.engr.mun.ca [134.153.12.79]

Trace complete.
Current time is 10:30:49.65a
Enter new time:
```

Figure 15. Output from ping.bat.

4.3 Discussion

4.3.1 User Testing

4.3.1.1 Questionnaire

As seen in Tables 2 through 6, the overall impression of all users was very positive. All felt that the application was a useful concept, and that the user interface chosen was understandable and relatively clear. Two of the three felt that the application would be useful in their daily routine, an indication that the application fills a niche that is not met with current products. User C indicated that the application would be more useful in telemedicine applications; however, this may have been coloured by the impression that its performance was slow using the hardware/access technology combination that this user had available for evaluation.

Some responses did indicate confusion regarding the intent of controls and presentation fields; however, these responses were infrequent compared to the positive responses. It is likely that user familiarity and a comprehensive user manual, neither of which were present in this instance, would improve the effectiveness of the presentation. In general, users seemed to respond well to the simple interface design chosen.

The responses indicated that the users did not find the download time onerous for the image and audio files chosen. The files were relatively small, however - the image files were JPG files of an approximate size of 50 Kbyte, and the audio files were in the 50 Kbyte-150 Kbyte range. Using these file sizes, maximum download time using a dialup connection is limited to approximately 1 minute, based on repeated trials. The users indicated that a maximum download time on the order of 1-2 minutes would be tolerable, although this opinion did vary from user to user.

4.3.1.2 Interviews

The interview results demonstrate the high degree of acceptance among the user community that this application would expect to generate. Acceptance of some sort of computer as a tool with which to conduct business is becoming commonplace; one would anticipate that medical professionals would be no different. The users ranged, in terms of their exposure to medical information systems, from one exposed on a daily basis to medical imaging using computers to one whose use of medical databases is solely a text-based one. In all cases, the usefulness of the application was readily apparent to the user. All users suggested possible applications beyond those suggested in the original scope of the project, and all were able to make suggestions regarding functionality that would make the system even more useful.

4.3.2 Delay Testing

4.3.2.1 General

The results of the delay testing permit a better understanding of the potential performance of the system under a variety of operating conditions, particularly as they relate to telemedicine applications. In order to use this application in a telemedicine milieu, remote connection to an ISP is a requirement. Establishing performance parameters for the application leads to the ability to benchmark performance and maximize the potential utility of the system.

The traffic performance of a TCP/IP network is somewhat analogous to a telephone network, although there are some significant differences. Unlike isochronous telephone traffic, which is transmitted using discrete channels which are allocated on a per user basis, TCP/IP traffic is asynchronous and may be buffered in multiple locations. In addition, whereas telephone traffic relies on a so-called "nailed-up" (fixed) connection being established between two points for the duration of the session, Internet connections only require a nailed up connection to the ISP; if the client computer is connected to a LAN, there may be no fixed connection beyond the client computer at all. Finally, whereas the traditional Time Division Multiplexing (TDM) telephone switch must process traffic for each established connection regardless of whether the end users are conversing or whether they are silent, processing of TCP/IP traffic by a LAN or router only occurs when an actual request for information or transmission occurs between two locations; at other times, the network is available for traffic between other clients. When the network becomes congested, IP traffic is buffered and, in severe congestion, discarded without notification to the user, whereas with a telephone network, congestion results in denial of service and congestion notification through the use of denial of service tones (e.g. reorder).

4.3.2.2 Probability Distribution

The response time of a ping packet exhibited variation not only by time of day, but also varied for each individual test, during each of which the target was pinged four times. The response time was anticipated to be positive and possibly tailed. It was hypothesized that the delay time of the ping packet at a given time of the day and week would exhibit behavior which would conform to the normal distribution. Figure 16 is a histogram of the number of occurrences of delay measurements for each 10 ms band from 100 to 340 ms.

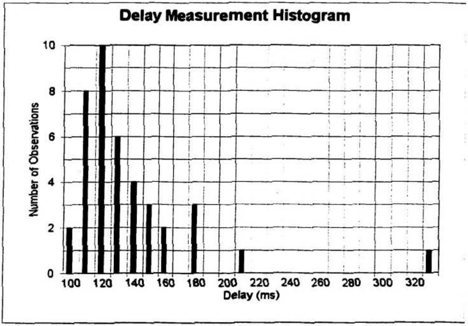


Figure 16. Delay Measurement Histogram, ADSL modem, Weekday, 10-11 am.

In order to test this hypothesis, the responses for a one hour period of one test were examined to see if they fit the normal distribution, by applying the chi-squared test for goodness of fit. The mean and variance of the response times were calculated; from this, the normal distribution curve which fit these parameters was determined. The curve was divided into four equiprobable regions, using a table of normal distribution probabilities [Pfe73]. The observed delay times were matched to the corresponding regions of the curve and the chi-squared test applied to determine goodness of fit [Leo94]. It was found that the observed delay times did not fit the normal distribution. This result is intuitively verified by observation, as well. The normal probability distribution function (pdf) is asymptotic to the ordinate; this implies that the actual observed delay time must be unbounded in either direction with respect to the mean. In fact, the delay time is bounded in the lower limit by the transmission time from the sender to the receiver, which is finite and depends on the transmission facility used and the distance separating the two.

Other pdfs have been shown to be more applicable to bounded data sets. The gamma distribution with 2 degrees of freedom was also tested for goodness of fit. Superficially, the histogram shown above bears a resemblance to the curve of the gamma distribution; however, it was found that the data did not fit in this instance either. As with the case of the normal distribution, a chi-squared test for goodness of fit was applied to the data. In this case, because tables of the probabilities associated with the pdf were not available, the pdf was integrated over four intervals and the data tested using the chi-squared test.

The gamma distribution pdf is given by

$$p(x) = A^2 x e^{-Ax}$$

where

$$A = \frac{2}{\sigma^2}$$

and the expected value E(x) for n degrees of freedom is given by

$$E(x) = \frac{2n}{\sigma^2}$$

Upon integration of the pdf over the four intervals, the expected and actual number of observations in each interval were compared, and, using the chi-squared test for 2 degrees of freedom, the goodness of fit determined as before. Again, the data did not fit the distribution.

We have shown that neither the normal distribution nor the gamma distribution with 2 degrees of freedom applies to the data in this instance. Further investigation was not conducted into the best distribution fit of the data.

4.3.2.3 Time of Day Variation

It was anticipated that the results of the delay testing would show some time of day variation. Because the amount of traffic presented to a router is dependent on the number of users conducting operations at a given time, time of day variation is to be expected as the number of persons using the Internet for various activities changes. This is consistent with what has been demonstrated in telephone networks. In telephone system design, designers consider the daily busy hour when performing capacity engineering. In most cases, the telephone switch will see two daily weekday peaks - one in the morning around 10 am and a second smaller peak at around 3 pm. Weekend traffic does not show as much variation and does not occur to the same degree as weekday traffic. Designing their networks to handle the capacity of these peaks means that telephone network operators will ensure that blockage of telephone calls due to insufficient capacity is minimized.

As can be seen in Figures 17 and 19, consistent weekday time of day variation does exist in transmission delay, regardless of the access technology employed. A traffic peak occurs in mid-afternoon around 3 pm, presumably reflecting business use. A second, smaller peak occurs in the late evening around 9 pm, and probably corresponds with home recreational Internet use. As anticipated, the lowest traffic periods occur in the early morning hours, 3 to 6 am. The data in Figure 18 shows significant oscillation; this is felt to be "noise" caused be a relatively small sample size and is not thought to be relevant.

In order to confirm that weekend access patterns varied from weekday access, the delay test results were compared separately. If the traffic patterns corresponded to the weekday pattern, the plot of response time vs. time of day should show the same type of variation. As shown in Figures 18 and 20, the traffic did not vary nearly as significantly in terms of maximum delay on weekends during the day compared to the results seen for weekday traffic; instead, the traffic variation was consistent regardless of the access technology being tested. This confirms the hypothesis that weekend traffic patterns differ from weekday traffic.

It is important to note that actual delay time is largely a function of the routing between the sender and receiver. A client who is located a long distance from the server site will not only experience greater delays because of distance, but the number of routers, and the traffic on those routers, will also affect the throughput. Measurements of delay are by no means absolute, even when the client and server are located in close proximity. The ISP used can have a significant effect on the delay, depending on the routing configuration by which the client communicates with the server. In order to illustrate this point, two different dialup 33 kbit/s lines were tested. In one instance, the dialup connection was to a local server which was located one hop from the server on which the application resided (Figure 23). The second dialup was to a commercial ISP server located 6 hops from the application server. In each case, the server was pinged 50 times at a low traffic period (around 9 am on a Sunday) and the average delay shown in Table 7. As shown, the average round trip delay in the case of the commercial ISP was considerably higher than that observed when using the local MUN dialup.

A medical information system would be required on a 24 hour, 7 day a week basis. Although delay varied, for the ADSL, simulated BRI ISDN, and Ethernet the variation did not appear excessive. For the dial line, weekend traffic only was tested. As can be seen, the delay always exceeds the 150 ms limit discussed in Chapter 2 by a considerable margin. Further results will also show the inapplicability of dialup modem use for this system.

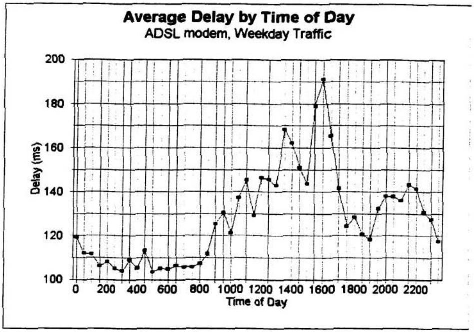


Figure 17. Delay by Time of Day, ADSL, Weekday Traffic.

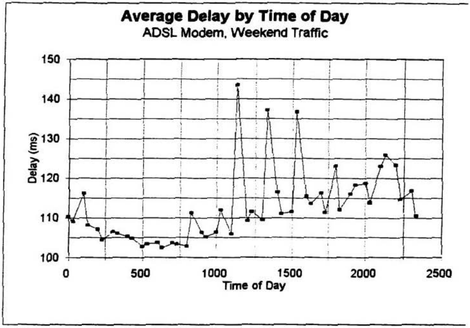


Figure 18. Delay by Time of Day, ADSL, Weekend Traffic.

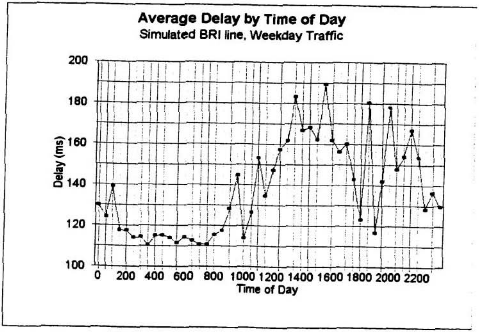


Figure 19. Delay by Time of Day, Simulated BRI ISDN, Weekday Traffic.

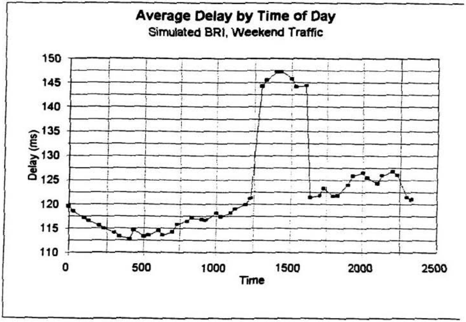


Figure 20. Delay by Time of Day, Simulated BRI ISDN, Weekend Traffic.

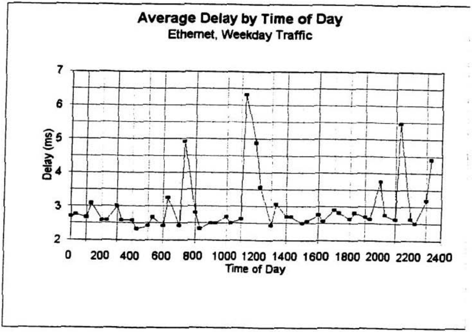


Figure 21. Delay by Time of Day, Ethernet LAN, Weekday Traffic.

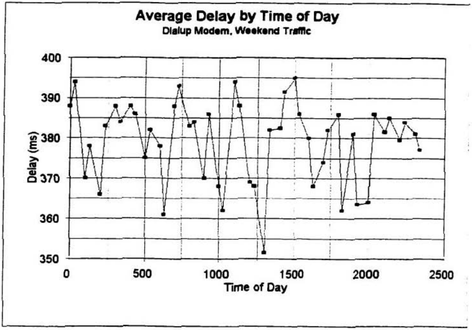


Figure 22. Delay by Time of Day, Dialup Modem, Weekend Traffic.

```
Current date is Sun 04-18-1999
Current time is  8:40:08.88a

Tracing route to nano.engr.mun.ca [134.153.12.79] over a maximum of 30 hops:

  1   398 ms   399 ms   399 ms  draco.engr.mun.ca [134.153.12.27]
  2   397 ms   400 ms   398 ms  nano.engr.mun.ca [134.153.12.79]

Trace complete.
```

Figure 23. Dialup to MUN - tracert result.

```
Current date is Sun 04-18-1999
Current time is  8:50:45.25a

Tracing route to nano.engr.mun.ca [134.153.12.79] over a maximum of 30 hops:

  1   498 ms   500 ms   499 ms  198.165.75.149
  2   492 ms   500 ms   500 ms  gateway.avint.net [198.165.75.1]
  3   500 ms   500 ms   499 ms  198.165.205.17
  4   492 ms   500 ms   500 ms  198.165.251.246
  5   494 ms   500 ms   500 ms  198.165.251.250
  6   492 ms   500 ms   500 ms  198.165.251.6
  7   494 ms   500 ms   500 ms  nano.engr.mun.ca [134.153.12.79]

Trace complete.
```

Figure 24. Dialup to Commercial ISP - tracert result.

Access Location	Average Delay (ms)
MUN	381
Commercial ISP	510

Table 7. Average Delay - Local MUN dialup vs. Commercial ISP.

4.3.2.4 Download Time

As demonstrated in the responses to the questionnaires discussed previously, the time to access and download patient information in this application is of the utmost importance to the user. The tolerance of delay for the files to download varies from seconds to one or two minutes. It is anticipated that a delay on the order of one minute would be the maximum acceptable in practice.

In order to determine the difference between the actual and theoretical download times, repeated trials were conducted to establish the relationship between delay and download time. The server was pinged 30 times in succession and the average delay determined. Then, an audio file measuring 642,818 bytes in size was downloaded. This was the maximum size permitted by the audio editing software used. Trials were performed using the ADSL circuit, simulated BRI ISDN, and dialup modem. An attempt to measure the download time using the direct connection to Ethernet was made, but the measured download time was too small to be meaningful. The results of the download tests are presented in Tables 8 through 10. In each table, the measured average round trip delay is shown, followed by the minimum theoretical download time and the average observed download time.

As shown, there is a poor correspondence between the theoretical download time and the observed download time. In determining the theoretical download time, one must know the configuration of the networking software regarding the setting for the Maximum Transmission Unit (MTU). The MTU is the maximum datagram size transmitted by the sender. In determining the optimum MTU, a tradeoff is made - if the MTU is too small, the throughput is drastically reduced; however, if the MTU is too large, packet losses will increase transmission time as the amount of data to re-send is large. An MTU of 1500 bytes was assumed, as this is the maximum permitted by Ethernet [Fre95]. As shown, the theoretical download times greatly exceeded those expected for the ADSL modems if the MTU of 1500 bytes was used, indicating that the actual MTU must be considerably larger. The same holds true for the dialup modem - the download time is much smaller than if an MTU of 1500 was specified, but larger than if the maximum MTU was implemented. In should be noted that in the calculations, it was assumed that 80 bytes of the datagram were used by the TCP and IP headers.

For some reason the results from the simulated BRI ISDN circuit correspond to a MTU setting of close to 1500 bytes. This is probably due to the configuration change of the ADSL line from 128 kbit/s to 512 kbit/s in the downstream. The 128 kbit/s was probably produced by throttling the data at the service provider, in effect buffering the data in order to simulate a lower speed connection. This would also explain the similar delay measurements seen at 128 kbit/s and 512 kbit/s - the small size of the ping datagram would not incur the same buffering effect as the larger datagrams of the audio file download. This indicates that the simulated ISDN circuit using the ADSL modem is probably not representative of a real world application.

Trial	Avg. Round Trip Delay (ms)	Avg. Measured Download Time (s)	Min. Theoretical Download Time (s)		Difference (%)	
			65535 MTU	1500 MTU	65535 MTU	1500 MTU
1	119.8	15.6	1.2	51.3	1200	30.4
2	121.5	14.7	1.2	52.1	1125	28.2
3	114	14.7	1.1	48.9	1236	30.0

Table 8. Comparison of Actual vs. Theoretical Download time - ADSL.

Trial	Avg. Round Trip Delay (ms)	Avg. Measured Download Time (s)	Min. Theoretical Download Time (s)		Difference (%)	
			65535 MTU	1500 MTU	65535 MTU	1500 MTU
1	115.8	47.4	1.1	52.4	4209	90.4
2	121.6	47.6	1.2	55.0	3866	86.5
3	126.5	48.5	1.2	57.3	3941	84.7

Table 9. Comparison of Actual vs. Theoretical Download time - Simulated BRI.

Trial	Avg. Round Trip Delay (ms)	Avg. Measured Download Time (s)	Min. Theoretical Download Time (s)		Difference (%)	
			65535 MTU	1500 MTU	65535 MTU	1500 MTU
1	336.4	62	3.3	152.3	1778	40.7
2	339.3	61.5	3.3	153.6	1763	40.0
3	336.9	60.9	3.3	152.5	1745	39.9

Table 10. Comparison of Actual vs. Theoretical Download time - Dialup modem.

As an example of the limitations of downloading large multimedia files using the various access technologies discussed, consider an uncompressed grey scale image of 300x300 dots per inch (dpi) resolution, utilizing 256 grey scales (8 bits/pixel) and measuring 8 inches by 10 inches probably approaches the worst case file size for an individual image. This translates into an image file of 7.2 Mbyte. Similarly, a 10 minute consultation audio recording of voice at a sampling rate of 8000 samples/sec and using 8 bit quantization, the current telephone industry Pulse Code Modulation (PCM) standard, would result in a file size of 6.4 Mbyte. In either case, a file size on the order of magnitude of 10 Mbyte is a good worst case benchmark to consider. Based on the observed delays tabulated in the round trip delay testing, best case (i.e. corresponding to the minimum observed delay) and worst case (corresponding to the maximum observed delay) minimum theoretical download times for a 10 Mbyte file are shown in Table 11. As the results for the simulated BRI ISDN simulation are suspect, they are not included.

Access Method	Observed Average Delay (ms)	Min. Download Time (s)	
		65535 MTU	1500 MTU
ADSL	121.3	18.5	852
Dialup modem	337.5	51.4	2373
Ethernet (estimated)	5	0.76	33.3

Table 11. Expected Minimum Delay to Download 10 Mbyte File.

Chapter 5

Conclusions

5.1 General

It has been demonstrated in this thesis that a Web based multimedia medical information system is a viable concept. In order for it to be a practical application, it will have to be implemented using relatively high speed access technology. This chapter will discuss the status of the goals stated earlier, discuss areas where future research could be conducted relative to this thesis, provide recommendations and summarize the conclusions which have been reached herein.

5.2 Goals

In Chapter 3, specific goals for this research were enumerated. In this section, the goals will be re-examined, and the findings relative to each goal discussed as they pertain to the research.

The goals of this research were stated to be the following:

1. To implement a "proof of concept" Web based multimedia MIS.

2. To examine the viability of using the Web for a system of this nature

3. To develop an understanding of the issues surrounding the design of a system of this type.

4. To create a set of performance objectives for a system of this type.

5. To evaluate the potential performance of the system over commonly available access media.

6. To evaluate the suitability of using the Java programming language for the development of a system of this nature.

The implementation of a "proof of concept" multimedia MIS has permitted the evaluation of the performance of such a system, as well as providing an opportunity to dialogue with health care professionals and obtain their input regarding the feasibility of the concept. The implementation was deliberately kept simple, both to minimize development time and expense and to provide a subset of the anticipated functionality of the full implementation which would allow health care professionals to provide feedback while not being overwhelming in scope. The prototype proved to be a useful tool in assessing the performance and constraints of such a system in an operational setting.

The prototype also permitted examination of the WWW as a potential wide area networking topology for the application. It was shown that the Web is potentially acceptable for the implementation of a multimedia MIS, provided the constraints imposed by its public access nature are borne in mind. The advantages of implementing this system using the Web are the inherent client-server architecture, the ubiquity of Web access, the widely available standard browser interface, and the degree to which society has embraced both the concept and the fact of accessible computer networking.

An understanding of the issues surrounding the design of a system of this nature was the next goal of the research. While not addressed in this prototype, issues of security and access time are of paramount importance in an implementation of this type, and will be primary design factors in any real development. The advantages of providing this system using the Web are balanced by the need to design a system bearing in mind the constraints imposed by it; in fact, the choice of the Web for wide area network access may be the single most important design factor in developing a multimedia MIS. Other major factors include the database configuration, the choice of object-oriented or traditional structured software design, and the choice of a GUI.

In designing the prototype and conducting user testing, performance parameters which must be met have been quantified. Access time must be kept low; user interfaces must be clear and easy to understand; network access speeds must be relatively high; patient information must be secure from unauthorized access and be authentic. The performance of such a system will to a large extent determine its acceptance in the user community.

The potential performance of the system was examined in terms of access over the Internet using a standard, commonly available platform, and while performing the most common task to be expected, i.e. downloading multimedia files. It was found that download time could be a limiting factor in the usability of the system. Download time was, as anticipated, affected to a large degree by the type of connection used to gain access to the ISP, but it was also affected by time of day variation and routing from the client to the server. General guidelines may be established to determine expected download times for large files, but these will vary depending on the configuration.

Finally, the Java programming language was evaluated for suitability in this application. It was found that Java is an ideal programming environment in many respects, but it does have limitations which must be dealt with. Java's greatest strength is its platform independence, which permits "write once, run anywhere" software design - at least in theory. In the development of the prototype, it was found that while running in a

Windows 95 environment, no problems were encountered. However, the implementation appearance and functionality differed on other platforms. Any working system would have to be tested thoroughly on as many hardware/software platforms as possible to ensure complete functionality.

In general, the goals outlined were met. The prototype was developed and used to test several aspects of the performance of a working system. It was found that the spiral design paradigm described in Chapter 3 worked well in this instance, permitting refinement of the design of the system while still in an early stage of development.

5.3 Future Directions

5.3.1 General

The changing nature of the environment in which a multimedia MIS would be designed and operated means that the constraints on the system are changing as well. Future developments in the field of multimedia communications will enhance the viability of the concept by ensuring that issues which currently exist regarding the design and implementation of such a system are dealt with in a less onerous fashion than is currently the case.

One aspect of the design of the system which has been alluded to but not dealt with in any formal manner is the issue of compatibility. Many different file formats, storage types, communications protocols, and hardware/software platforms exist in a medical setting, some of which are proprietary, and many of which are incompatible. The design of this system has studiously maintained compatibility with standard formats, but in order for the system to have practical applicability, it will have to deal with other formats as well. Some existing modalities may never be compatible with a standardized system because of their proprietary nature.

Despite this, developments in the medical field have moved toward adopting medical-specific standards such as DICOM, HL7, and CORBAmed. Any system of this type must incorporate these standards in its design in order to maintain compatibility with new applications. Failure to maintain standardization would severely limit the usefulness of the application.

The implemented version of this application would require extensive search and retrieval capabilities. Commercially available methods of search in multimedia databases are limited to textual search using keywords. Research into more dynamic methods of content-based retrieval, using, for instance, multimedia controls to allow the user to specify attributes of the requested records, will be necessary to fully utilize a multimedia database.

As demonstrated in the user testing, video display is a requirement of this application. As discussed, the prototype functioned for its purpose without integration of video; however, a future version should permit video display. The download and display of high quality video may in fact be a limiting factor in determining the extent to which an application of this nature may function as a ubiquitous patient information system. While the use of video is relatively limited in most current diagnostic tools, one may expect that future modalities may rely on video or animation to show bodily functions which vary in time, such as an MRI of a heart while functioning, or to record patient treatment sessions.

Another area in which such a tool is useful, particularly in a telemedicine milieu, is in providing at least a rudimentary conferencing capability, preferably videoconference. Videoconferencing is a tool which is being used increasingly to supplant face to face meetings. In the case of medical consultative or diagnostic session, the ability to look at the person to whom one is speaking imparts an air of personal contact which would otherwise be absent.

5.3.2 Prototype Design Revision

The recursive nature of the spiral design paradigm encourages the development of a more refined prototype based on the lessons learned from a previous version. In this instance, the design, implementation, and testing of the first version prototype demonstrates the areas in which a future prototype can be improved.

The next version prototype must implement more extensive functionality such as the search and retrieval functions. While the initial prototype has demonstrated the "look and feel" of the system, its limited functionality precludes meaningful evaluation of the overall system design. A more extensive implementation will permit the examination of the full capability of the WWW implementation.

The design must be refined to incorporate as fully as possible the appropriate medical standards, such as DICOM and HL7. In this manner, interoperability is ensured and the system functionality is increased. As well, standard file formats for the various media must be used. This may require the incorporation of third party or custom APIs in order to permit file manipulation.

The refined design of the next prototype will allow the development of a more effective version of the application. This in turn will allow the development of a greater degree of sophistication in future implementations.

5.4 Recommendations

In research of this nature, many avenues are left to be explored. This section discusses some of the most relevant. In this instance, relevance is defined in terms of further developing the application, both in its usefulness to the client population and in its functionality.

By using a commercially available multimedia database, the design of a production system may be greatly simplified. Future research may involve the measurement of productivity gains which may accrue from using such a technology. On the other hand, added overhead may mean that access time may be greatly increased in an operational database. This may make the use of such a product less desirable from a performance standpoint.

In addition to transmission delay, significant delay may be experienced if a database server takes a long time to access records. In a production database of this type, one would expect the records for patients to exist in the order of magnitude of thousands or even tens of thousands, while each patient record will potentially contain hundreds of records. Such a system can be expected to handle requests from perhaps hundreds of users simultaneously at peak periods. The design of a server configuration to handle this volume of traffic and still present the requested data in a reasonable amount of time is also a subject of future research.

The scope of application of such a system also remains to be determined. Whether a local unit such as a hospital would contain its own server or servers, or whether a single central server farm would be set up for an entire health authority, is subject to discussion. In the former case, a query from a client to a particular database server could cause an agent to be dispatched to other servers, retrieving patient information from a variety of sources at different locations. This configuration may result in faster access time in the local environment. The relative merits of each configuration would have to be determined before such a decision would be made.

Much research remains to be conducted in the area of content-based retrieval in multimedia databases. Keyword searches are incapable of capturing any but the barest essence of the information contained within a multimedia record.

The effects of the ability to specify a QoS parameter should also be examined. In future, IPv6 will permit the specification of QoS, greatly improving the performance of multimedia transmission over the Internet. It remains to be determined what effect this may have on an application such as this; it may be that the effect will be substantial. The ability to specify QoS may allow use of a system of this type with lower speed access technologies, making it more useful in areas where low speed dialup access remains the sole method of connection of a client to an ISP.

5.5 Conclusion

We have seen that a multimedia medical information system operating over the WWW has applicability in both a telemedicine and local context. The prototype designed and implemented as described in this thesis is consistent with directions in patient information systems, as described in the literature, and has met with approval from medical professionals. It has been shown that within the constraints imposed by the medium, it is conceivable that a system of this type could be implemented in an operational setting.

While the original scope of the project as defined in the Project Definition in Chapter 1 limited the research to telemedicine applications, it has been shown that an application of this nature is not limited to a telemedicine setting. The design of a database server storing all multimedia patient records could be equally useful in a hospital, clinic, or metropolitan setting. Emphasis may be placed on different aspects of the design in each setting; for instance, a telemedicine setting may place relatively more emphasis on videoconferencing than would a clinic. In any instance, the application should provide fast, easy access to a single patient record, providing benefits to the practitioner and the patient as well.

References

[Abo95] N. Abouchi, M. Jourlin, O. Bohbot, C. Faurie, and R. Grisel, "Network Image Data Bank Prototype: the R.S.I. Project", *SPIE*, Vol. 2451, 1995, pp. 360-369.

[Abr98] A. Abrardo, and A.L. Casini, "Embedded Java in a Web-based Teleradiology System", *IEEE Internet Computing*, Vol. 2, No. 3, May-June 1998, pp. 60-68.

[Adj97] D. Adjeroh, and K. Nwosu, "Multimedia Database Management - Requirements and Issues", *IEEE Multimedia*, Vol. 4, No. 3, July-September 1997, pp. 24-33.

[Ahn94] C.B. Ahn, I.Y. Kim, and S.W. Han, "Medical Image Compression Using JPEG Progressive Coding", *1994 Nuclear Science Symposium and Medical Imaging Conference*, Norfolk, Virginia, USA, 1994, pp. 1336-1339.

[Als94] Y. Alsafadi, K. Maloney, K.M. McNeill, M.V. Parra, J. Guice, and R. Martinez, . "Design of the Interface Between the Intensive Care Unit and Hospital/radiology Information Systems at University of Arizona", *1994 Annual International Phoenix Conference on Computers*, Phoenix, AZ, USA, 1994, pp. 404-412.

[Ban94] S. Banerjee, "Multimedia Traffic Analysis and Control in a High-Speed Medical Communications Environment", *1994 Southcon Conference*, Orlando, FL, USA, 1994, pp. 416-420.

[Biz93] Y. Bizais, A.M. Forte, J.P. Guedon, D. Corbard, F. Calmel, and F. Lavarie, "An Object Oriented Image Data Model for a Knowledge Based Image Processing System", *SPIE,*, Vol. 1898, 1993, pp. 702-713.

[Bla94] G.S. Blair, G. Coulson, and N. Davies, "System Support for Multimedia Applications: An Assessment of the State of the Art", *Information & Software Technology*, Vol. 36, No. 4, 1994, pp. 203-212.

[Boo90] G. Booch, *Object-oriented design with applications*, Benjamin-Cummings, Redwood City, CA, USA, 1991, 580 p.

[Bra97] T. Braun, "Internet Protocols for Multimedia Communications", *IEEE Multimedia*, Vol. 4, No. 3, July-September 1997, pp. 85-90.

[Bre94] M. Breeuwer, R. Heusdens, and P. Zwart, "Overlapped Transform Coding of Medical X-ray Images", *SPIE*, Vol. 2164, 1994, pp 264-275.

[Buc94] G. Bucci, R. Detti, and V. Pasqui, "Sharing Multimedia Data over a Client-Server Network", *IEEE Multimedia*, Vol. 1, No. 3, Fall 1994, pp. 44-55.

[Col93] F. Colaitis, and A. Leger, "MHEG: An Emerging Standard for the Interchange of Multimedia and Hypermedia Conferences", *1993 IEEE International Conference on Communications*, Geneva, Switzerland, 1993, pp. 542-546.

[Fai85] R.E. Fairley, *Software Engineering Concepts*, McGraw-Hill, New York, NY, USA, 1985, 364 p.

[Fan93] C. Fan, "Performance Comparison and Analysis of XTP and TCP/IP over the BERKOM Broadband ISDN Network", *IEEE Infocom '93*, San Francisco, CA, USA, 1993, pp. 1154-1161.

[Fla97] D. Flanagan, *Java in a Nutshell*, O'Reilly & Associates, Sebastopol, USA, 610 p.

[For97] D.J. Foran, P.P. Meer, T. Papathomas, and I. Marsic, "Compression Guidelines for Diagnostic Telepathology", *IEEE Transactions on Information Technology in Biomedicine*, Vol. 1, No. 1, March 1997, pp. 55-59.

[Fre95] R. Freeman, *Practical Data Communications*, John Wiley & Sons, New York, NY, USA, 1995, 632 p.

[Fuh94] B. Fuhrt, "Multimedia Systems: An Overview", *IEEE Multimedia*, Vol. 1, No. 1, Spring 1994, pp 47-59.

[Fuh95] B. Fuhrt, "A Survey of Multimedia Compression Techniques and Standards, Part I: JPEG Standard", *Real Time Imaging*, Vol. 1, No. 1, 1995, pp. 49-67.

[Gol94] F. Golshani, and N. Dimitrova, "Retrieval & Delivery of Information in Multimedia Database Systems", *Information & Software Technology*, Vol. 36 No. 4, April 1994, pp 235-242.

[Gol90] M. Goldberg, S. Panchanathan, and L. Wang, "A Comparison of Lossy Coding Techniques for Digitized Radiographic Images", SPIE, Vol.1232, 1990, pp. 327-336.

[Gro94] W. I. Grosky, (1994). "Multimedia Information Systems", *IEEE Multimedia*, Vol. 1, No. 1, Spring 1994, pp. 12-24.

[Hea99] Health Level 7. *Health Level 7 Ballot Draft, Version 2.3*, http://www.mcis.duke.edu:/standards/HL7/pubs/ballot.html, 1999.

[Hic90] N.M. Hickey, J.G. Robertson, and M. Coristine, "Integrated Radiographic Communication System: A Radiology Multimedia Communication System", *Journal of*

Thoracic Imaging, Vol. 5, Issue 1, 1990, pp 77-84.

[Hof97] A. van Hoff, "The Case for Java as a Programming Language", *IEEE Internet Computing*, Vol. 1, No. 1, January-February 1997, pp. 51-56.

[Huh98] M. Huhns, and M. Singh, "All Agents are not Created Equal", *IEEE Internet Computing*, Vol. 5, No. 2, April-June 1998, pp. 94-96.

[Ims99] IMSIG. "HL7 Image Management SIG Goals and Milestones", http://dumccss.mc.duke.edu/standards/HL7/sigs/image-management/HTML/goals.html, 1999.

[Ino94] T. Inoue, and T. Inoue, "ISDN Application for Image Transmission", *Computer Methods & Programs in Biomedicine*, Vol. 43, 1994, pp 139-144.

[Kar90] A. Karmouch, N.D. Georganas, and M. Goldberg, "Multimedia Database Design and Architecture in Radiology Communications", *Proceedings of the Medical Imaging IV Conference*, Newport Beach, CA, USA, 1990.

[Kar90a] A. Karmouch, L. Orozco-Barbosa, N.D. Georganas, and M. Goldberg, "A Multimedia Medical Communications System", *IEEE Journal on Selected Areas in Communciations*, Vol. 8, No. 3, April 1990, pp. 325-339.

[Kil94] A.C.M. Kilsdonk, and A. van der Werff, "Structuring the Patient Record: NUCLEUS", *Computer Methods and Programs in Biomedicine*, Vol. 45, 1994, pp. 127-130.

[Kle94] L. Kleinholtz, and M. Ohly, "Supporting Cooperative Medicine; the BERMED project", *IEEE Multimedia*, Vol. 1, No. 4, Winter 1994, pp. 44-53.

[Leo94] A. Leon-Garcia, *Probability and Random Processes for Electrical Engineering*, 2nd. Ed., Addison-Wesley, Reading, Massachusetts, USA, 1994, 596 p.

[Lit94] H.I. Litt, and J.W. Loonsk, "Graphical Representation of Medical Information in the Visual Chart", *1994 7th Symposium on Computer-Based Medical Systems*, Winston-Salem, NC, USA, 1994, pp 252-257.

[Liu94] P Liu, "ISO Standards from an Information Infrastructure Perspective", *IEEE Multimedia*, Vol. 1, No. 4, Winter 1994.

[Mal97] K. Maly, H. Abdel-Wahab, C.M. Overstreet, J.C. Wild, A.K. Gupta, A. Youssef, E. Stoica, and E.S. Al-Shaer, "Interactive Distance Learning over Intranets", *IEEE Internet Computing*, January-February 1997, pp 60-71.

[Man94] A. Manduca, "Medical Image Compression with Wavelet/subband Coding", *Intelligent Engineering Systems Through Artificial Neural Networks*, Vol. 4, ASME Press, New York, NY, USA, 1994, pp. 645-650.

[May92] D.J. Mayhew, *Principles and Guidelines in Software User Interface Design*, Prentice Hall, Englewood Cliffs, NJ, USA, 1992, 619 p.

[Mor97] Morgan, Bryan, *Visual J++ Unleashed*, 2nd Ed., Sams.net Publishing, Indianapolis, Indiana, USA, 1997, 842 p.

[Nem98] National Electrical Manufacturers Association. Digital Imaging and Communications in Medicine (DICOM), Part 1: Introduction and Overview. National Electrical Manufacturers Association, Rosslyn, VA, USA, 1998.

[Nie96] P. Niemeyer, and J. Peck, *Exploring Java*, O'Reilly & Associates, Sebastopol, USA, 407 p.

[Omg99] Object Management Group "OMG Background Information", http://www.omg.org/omg/background.html, 1999.

]Paz97] P. Pazandak, and J. Srivastava, "Evaluating Object DBMSs for Multimedia", *IEEE Multimedia*, Vol. 4, No. 3, July-September 1997, pp 34-49.

[Pfe73] P. E. Pfeiffer, and D. A. Schum, Introduction to Applied Probability, Academic Press, New York, NY, USA, 1973, 403 p.

[Rig95] M.J. Rigby, R. Roberts, and J.G. Williams, "Objectives and Prerequisites to Success for Integrated Patient Records", *Computer Methods and Programs in*

Biomedicine, Vol. 48, 1995, pp. 121-125.

[Sha95] M.M. Shahsavari, O.M. Ba-Rukuab, and M.N. Chari, "Multimedia Implementation on Ethernet", *IEEE Southcon Conference*, Lauderdale, FL, USA, 1995, pp. 205-209.

[Sod96] J. Sodhi, and P. Sodhi, *Object-Oriented Methods for Software Development*, McGraw-Hill, New York, NY, USA, 1996, 277 p.

[Stu95] H.J. Stuttgen, "Network Evolution and Multimedia Communication", *IEEE Multimedia*, Vol. 2, no. 3, Fall 1995, pp. 42-59.

[Sud93] G.N.M. Sudhakar, A. Karmouch, and N. Georganas, "Design and Performance Evaluation Considerations of a Multimedia Medical Database", *IEEE Transactions on Knowledge & Data Engineering*, Vol. 5, No. 5, Oct. 1993, pp. 888-894.

[Swa93] D. Swanberg, C.F. Shu, and R. Jain, "Architecture of a Multimedia Information System for Content-Based Retrieval", *Network and Operating System Support for Digital*

Audio & Video, ed. P.V. Rangan, Springer-Verlag, 1993.

[Tho97] G.R. Thoma, and R.L. Long, "Compressing and Transmitting Visible Human Images", *IEEE Multimedia*, Vol. 4, No. 2, April-June 1997, pp36-45.

[Tre94] S. Treu, User Interface Evaluation: A Structured Approach, Plenum Press, New York, NY, USA, 1994, 282 p.

[Who88] World Health Organization, Informatics and Telematics in Health, World Health Organization, Geneva, Switzerland, 1988.

[Wil94] M.H. Williams, and J. Hu, "Making Heterogeneous Medical Databases Interoperable", *Computer Methods and Programs in Biomedicine*, Vol. 43, 1994, pp. 275-281.

[Won96] S.T.C. Wong, "A Cryptologic Based Trust Center for Medical Images", *Journal of the American Medical Informatics Association*, Vol. 3, No. 6, November-December 1996, pp. 410-421.

[Won97] S.T.C. Wong, and H.K. Huang, "Networked Multimedia for Medical Imaging", *IEEE Multimedia*, Vol. 4, No. 2, April-June 1997, pp 24-35.

[Yam93] H. Yamaguchi, H. Ichihara, H. Mizumoto, and S. Masaki, "B-ISDN Based Multimedia Communication System Supporting Group Work", *SPIE*, Vol. 1976, 1993, pp. 193-201.

[Yan95] Q. Yang, A. Vellaikal, and S. Dao, "MB+-Tree - A New Index Structure for Multimedia Databases", *International Workshop on Multimedia Database Management Systems*, Blue Mountain Lake, NY, USA, 1995, pp 151-158.

Appendix A

Software Design Specifications

SYSTEM DEFINITION

QB:

A Multimedia Medical Information System for Telemedicine Applications

©Gerard Michael Dunphy, B. Eng., P. Eng.

INTRODUCTION

The purpose of this document is to describe the design objectives of QB, a multimedia medical information system for telemedicine applications. In this document are contained the Problem Definition, System Justification, Goals, Constraints, and Functions, User Characteristics, a description of the Design Environment, and Acceptance Criteria for the system.

PROBLEM DEFINITION

The purpose of this system is provide an environment for the communication of medical information to remote medical personnel. This information may include text, video, graphics, images, audio, and any other format that may be used in the medical field.

It is anticipated that the system will replace existing information storage and retrieval methods which rely on paper, film, and other physical storage media. These record keeping methods have significant deficiencies:

- High information retrieval times;
- Possibility of loss or mishandling;
- Poor information search capability.

The problem is defined as follows:

> *Design a computer based Document Management System (DMS) capable of storing, retrieving, manipulating, and displaying patient information. The system must provide privacy, security, real time information manipulation, and a simple user interface. It should operate under multiple computer platforms and hardware/software configurations.*

SYSTEM JUSTIFICATION

This system addresses significant shortcomings in the present methods of storing, retrieving, and disseminating medical information. The anticipated result of the implementation of this system will be more timely diagnosis, ease of manipulation of the information, and the ability to draw on the resources of remote practitioners in an efficient and cost effective manner.

The traditional patient information record has comprised a series of paper or film records stored in a folder or envelope and located at a central record storage location within a health care facility. This record is maintained by clerks who are responsible for organizing, storing, retrieving, and transmitting the information upon request of a health care professional. There are several disadvantages to this approach:

- Significant time is spent storing, indexing, and retrieving the patient file by records administrators and health professionals;
- There is considerable risk of the patient record being mislaid, damaged, or otherwise mishandled during the transmittal process, resulting in increased treatment costs as repeated examinations may be made necessary;
- Organization of records by a single indexing variable, the patient name, precludes the use of the collected information for other purposes, such as epidemiological studies and medical research;
- The health care practitioner may not have access to patient information located at institutions other than the one at which he/she is located. These records may provide significant information to the practitioner regarding the individual under their care;
- Delays are encountered when information must be transmitted to a specialist for consultation, especially if the specialist is located in another community.

The advantages of a computerized solution to the problems associated with this method of patient information manipulation are:

- Faster document retrieval time;
- Faster document transmission to and from remote sites;
- Less downtime due to record mishandling;
- Integration of patient records stored or generated at different locations;
- Search and retrieval capability using relatively sophisticated criteria.

SYSTEM GOALS

The goal of this system is to provide a useful, simple, powerful tool for medical practitioners. The system will allow practitioners to collaborate on the diagnosis and treatment of patients regardless of their geographical proximity. It will provide the practitioner with resources currently either unavailable or with limited availability owing to the distance involved.

The system will permit the storage and organization of patient records in a networked computer platform.

SYSTEM CONSTRAINTS

Many of the constraints imposed on this system are a result of the economic realities of the medical field. Because cost is a primary factor in the implementation of any new technology in the publicly funded Canadian medical environment, one of the primary goals of this system should be to minimize cost in its development, implementation, and maintenance. With this in mind, however, it is important to stress that many other factors determine the success and usability of such a system.

The primary design constraints on this system are as follows:

1. Privacy - The system must at all times maintain the privacy of the person whose records are being manipulated. This applies equally during retrieval, display, transmission, or modification of the records. Privacy may be maintained by the judicious use of passwords, encryption, encoding, and other methods.

2. Accuracy - At all times the system must maintain a faithful representation of the information as it was originally stored; so, for instance, the act of compression and decompression must not introduce artifacts into a diagnostic image. Likewise, a practitioner who attempts to add information to a patient record must have authorization to do so, and every effort must be made to ensure that the information being added is relevant to the record.

3. Ease of Use - The system will be used by a variety of persons with a range of computer skills. As a result, the user interface must be designed in such a manner that it is understandable and meaningful to the user.

4. Interoperability - The system should not require custom hardware or software, or any equipment that is difficult to obtain or maintain. Because the computing environments in which it is used may vary widely, it should be platform independent and able to function in a variety of combinations of hardware and software with little or no customization.

5. Ease of modification - Because of continuing advancements in the medical and technological fields, the system must be capable of being modified in future. Software should be modular in design and logically organized. Documentation must be complete and accurate.

SYSTEM FUNCTIONS

The system will deal with information which relates primarily to an individual person. The primary level of information management in this system will be the *record*, which will consist of an identifier, such as the individual's name, followed by a series of *components*, multimedia objects which contain information of some type relating to the individual.

In addition, QB will provide facilities for collaborative work, permitting a practitioner to consult with another in real time and exchange patient information.

The functionality of this system is dictated by the requirements of the user group. In order to support medical personnel in the performance of their responsibilities, it is anticipated that the system will provide as a minimum the following functions:

System Level Functions

- User authentication

- Information verification
- Synchronization
- Information query†
- Mail†
- Conferencing/group work†
- Help facility

Record Management Functions

- Creation
- Storage
- Deletion
- Search

Component Manipulation Functions

- Definition
- Creation
- Addition
- Removal
- Modification
- Compression†
- Annotation†

† These functions may be added to a later version of the system.

USER CHARACTERISTICS

The users of this system will primarily be health care workers with a variety of levels of exposure to computerized systems. As a result, the system must be tailored to allow users unfamiliar with computers to obtain results with a minimum degree of training, while allowing more sophisticated users the flexibility to work in a less restrictive environment. The user interface should be customizable to the degree that a user can work more efficiently as his/her level of familiarity with the system grows.

It is anticipated that the typical user would exhibit the following characteristics:

- Little time for training
- Need to obtain clear, unambiguous results in the minimum time necessary
- High degree of familiarity with commonly used paper medical records
- High degree of familiarity with medical terminology
- Normally working in a medical environment
- Familiarity with computers not necessarily great

DEVELOPMENT & OPERATION ENVIRONMENTS

The system will be developed using the object-oriented design (OOD) paradigm. The concepts embodied in the OOD philosophy lend themselves well to a system of this type, permitting rapid development of a relatively complex system.

The system will be programmed using the Java development language and HyperText Markup Language (HTML). Java was chosen because of development speed and its support of platform independence, allowing the code to be developed on any one of a variety of platforms and run under a variety of operating systems. By using HTML, the system will be accessible from any computer equipped with a graphical Web browser such as Netscape™ or Microsoft Internet Explorer™. Development in this environment means that a familiar Graphical User Interface (GUI) will be intrinsic to the design of the system.

ACCEPTANCE CRITERIA

Acceptance of the system will be granted when it is shown to perform its primary functions reliably and accurately. The following tests as a minimum will determine that the system has met a minimum level of acceptance:

- Ability to authenticate a valid user
- Ability to refuse an invalid attempt to gain access to the system
- Ability to create a record
- Ability to delete a record
- Ability to modify a record
- Ability to add a component to a record
- Ability to delete a component from a record
- Ability to annotate a component within a record
- Ability to validate the contents of a component
- Ability to apply compression to a component where applicable

SOFTWARE DESIGN SPECIFICATION

QB:

A Multimedia Medical Information System for Telemedicine Applications

Version 1.0

INTRODUCTION

The purpose of this document is to describe in detail the functional specifications of QB, a medical multimedia information storage and retrieval system. The document will include primary class definitions, data flow diagrams, and example screen dumps.

QB is a computer based Document Management System (DMS) capable of storing, retrieving, manipulating, and displaying medical patient information. The system will provide privacy, security, real time information manipulation, and a simple user interface, and will operate under multiple computer platforms and hardware/software configurations. QB addresses significant shortcomings in the present methods of storing, retrieving, and disseminating medical information. The anticipated result of the implementation of this system will be more timely diagnosis, ease of manipulation of the information, and the ability to draw on the resources of remote practitioners in an efficient and cost effective manner.

QB will be implemented using Java and is designed to be accessible through the Internet or a corporate Intranet using only a standard web browser. The user interface consists of a series of dialog boxes which permit even a user unfamiliar with an online medical database to browse and locate information on a specific patient. Future functionality could include the ability to browse and search through the use of sophisticated search algorithms which could permit audio, image, and video searching as well as text search.

The implementation environment for QB is Microsoft Visual J++ Professional Edition. Visual J++ was chosen because of the sophisticated user interface development tools it provides, permitting rapid development of user interface components of the system.

Current Implementation

A complete System Definition has been developed, which describes the overall operation of QB. This document will focus on a subset of QB which will be implemented at the present time. The purpose of this implementation is to test the performance of the system across a variety of communications facilities. It is anticipated that the system will be tested across a variety of facilities, including shared 10BaseT Ethernet, cable modem (shared 4 Mbps Ethernet protocol), Asymmetrical Subscriber Digital Line (ADSL), dialup

56K Flex, V.90 and V.34 modem. This implementation will be referred to as QB Version 1.0 (V1.0).

QB V1.0 will essentially be a rudimentary record display mechanism. Record input will be performed manually be hard-coding the information prior to program build. User modification of the database will not be permitted. Because search time will largely be a function of the host server and the database size, search modules will not be emphasized at this time. Simple sequential search functions will be implemented to give the user a "look and feel" indication of the program functionality.

The testing will be performed by loading the system with a sample of actual data, and then using medical personnel to assess the performance by measuring such criteria as response time, usability, and accuracy. The performance measurements will serve a twofold purpose: they will establish benchmarks for future testing, and they will allow the system to be iteratively refined to improve performance.

DATA FLOW DIAGRAMS

Appendix A contains two flowcharts for QB V1.0. Flowchart 1 shows the overall program flow. Because the bulk of the program information processing in V1.0 is contained in the Display Record dialog box, Flowchart 2 shows the functionality of this area alone.

The program begins by presenting the user with a Web page which permits the user entry to the QB application. In future releases, the application will be restricted by password entry. Once the user enters, a main menu will be presented which will allow a choice of collaborative work (to be implemented in future), or patient search. The patient search choice will lead to the first search screen, from which the user will search for a patient by name.

Once the patient record is accessed, the user will use the Display Record dialog box to display aspects of the record - audio, text, image, or video data that the patient record may contain. The user will have the capability of choosing from a drop-down box or, for more extensive lists, performing a search through the record. Either choice will eventually display the chosen information using a suitable viewer.

CLASS DEFINITIONS

The primary data constructor of the system is the object which contains the patient information - class patientFile. This object is used as a container for the patient record. It contains text fields for the name and MCP number of the patient, and arrays of text (ASCII text file), audio (wav format), image (bitmap or JPEG compressed), and video (MPEG2 format) for the medical records of the patient.

Other classes are created by the Visual J++ environment and need not be described in detail here.

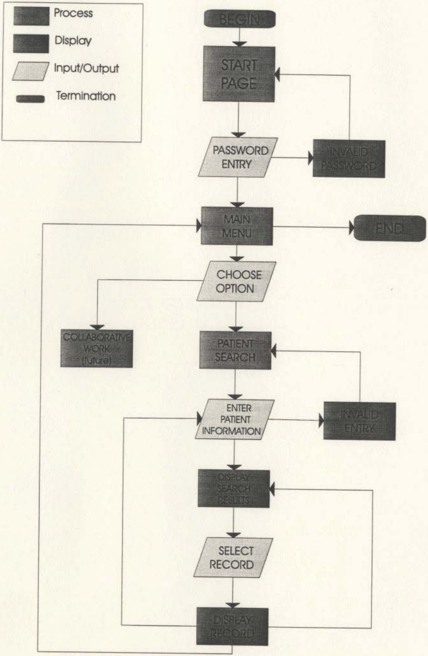
SCREENS

The primary user interface mechanism in QB V1.0 is the dialog box. Dialog boxes were chosen over pull down menus because they require little familiarity on the part of the user to navigate. All options are presented to the user when the dialog box starts up, thereby allowing even users unfamiliar with the program to intuitively navigate through the program.

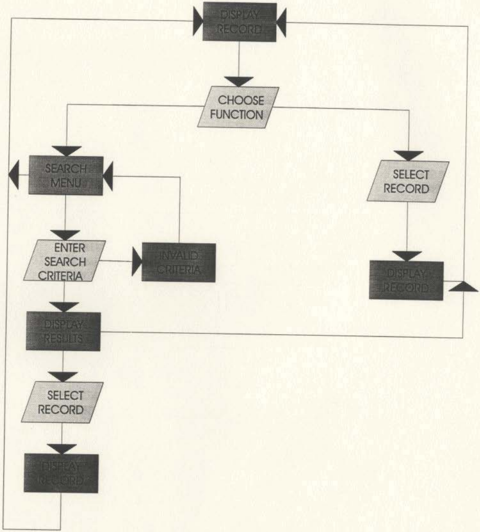
Appendix B contains screen dumps of the QB V1.0 user interface. The screen dumps are presented in the order in which they will appear, as appropriate.

Annex A

Flowcharts



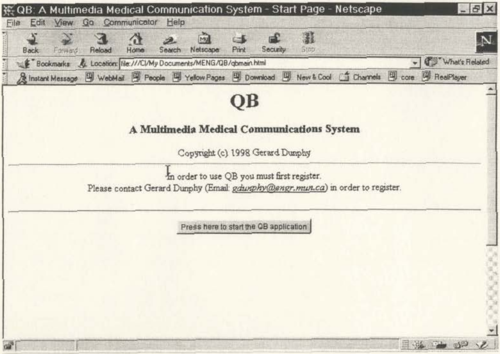
Flowchart]43 Overall Program

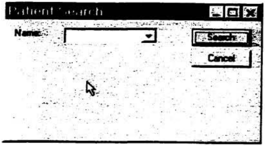


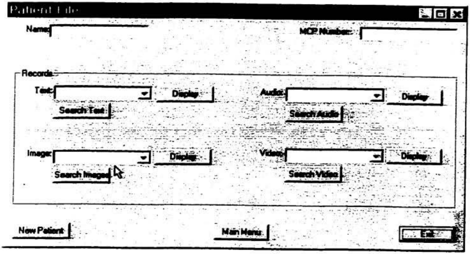
Flowchart 2. Display Record

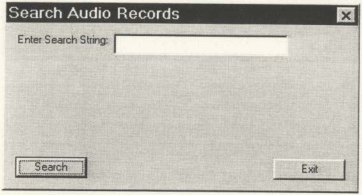
Annex B

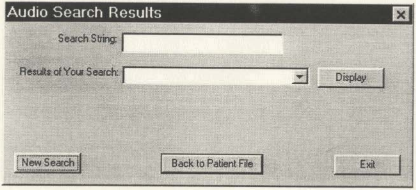
Screen Dumps

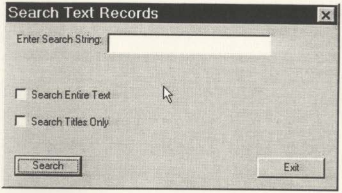

QB: A Multimedia Medical Communication System - Start Page - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Location: file:///C|/My Documents/MENG/QB/qbmain.html
What's Related
Instant Message WebMail People Yellow Pages Download New & Cool Channels core RealPlayer
QB
A Multimedia Medical Communications System
Copyright (c) 1998 Gerard Dunphy
In order to use QB you must first register.
Please contact Gerard Dunphy (Email: gdunphy@engr.mun.ca) in order to register.
Press here to start the QB application

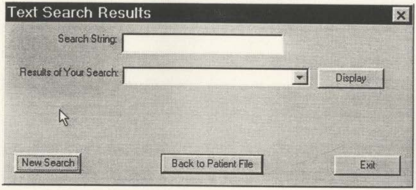

Patient Search
Name:
Search
Cancel

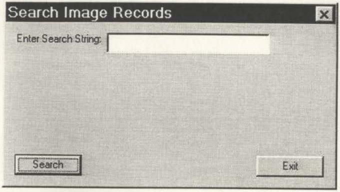

Patient File
Name:
MCP Number:
Records
Text:
Display
Search Text
Audio:
Display
Search Audio
Image:
Display
Search Images
Video:
Display
Search Video
New Patient
Main Menu
Exit


Search Audio Records
Enter Search String:
Search
Exit


Audio Search Results
Search String:
Results of Your Search:
Display
New Search
Back to Patient File
Exit


Search Text Records
Enter Search String:
Search Entire Text
Search Titles Only
Search
Exit


Text Search Results
Search String:
Results of Your Search:
Display
New Search
Back to Patient File
Exit


Search Image Records
Enter Search String:
Search
Exit

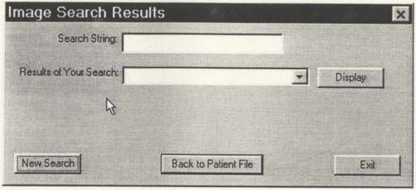
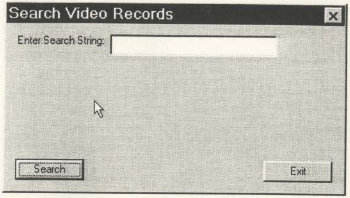
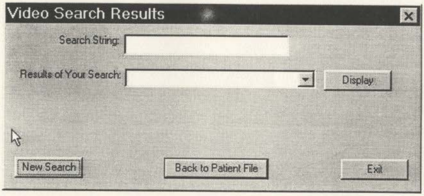


Image Search Results
Search String:
Results of Your Search:
Display
New Search
Back to Patient File
Exit


Search Video Records
Enter Search String:
Search
Exit


Video Search Results
Search String:
Results of Your Search:
Display
New Search
Back to Patient File
Exit

Appendix B

Class Definitions

```
public class QB extends Applet                //Main class for applet
        private PatientListFrame
        private theData  patientData
        public void init()
        public void destroy()
        public void paint(Graphics g)
        public void start()
        public void stop()

class PatientData extends Vector              //The main patient data list
        public String FirstName,
        public String MiddleName,
        public String LastName,
        public String FamilyDoctorName,
        public String MCP;
        public String DateofBirth,
        public String myURL;
        public textRecordVector textRecord;
        public audioRecordVector audioRecord;
        public imageRecordVector imageRecord;

class textRecord extends Record               //Text record list for a patient
                                              //Override text specific methods here
class audioRecord extends Record              //Audio record list for a patient

class imageRecord extends Record              //Image record list for a patient
```

```
class Record extends Vector                    //Base record class; overriden as needed
        public String creationDate;
        public String practitionerName,
        public String institutionName,
        public String Comments,
        public String FileName,
        public String FileType;
        public String myURL;
        public add()
        public delete()
        public find()
        public show()
        public stop()
        public pause()

class PatientListFrame extends Frame                    //Frame to choose patients from
        private IDD_PatientList patientListDialog = null;
        private PatientFileFrame patientFileFrame = null;
        private ListPanicWindow panicWindow = null;
        public Applet myparent = null;
        private PatientData myList;
        public PatientListFrame(Applet parent, PatientData theList)
        public boolean action(Event evt, Object obj)
```

```
class PatientFileFrame extends Frame                    //Frame to display a file
        private IDD_PatientFileDialog2 patientFileDialog = null;
        private PanicWindow panicWindow = null;
        private DisplayRecordFrame displayRecord = null;
        public PatientListFrame myparent = null;
        private URL fileURL = null;
        private PatientData myList = null;
        private int myIndex;
        public PatientFileFrame(PatientListFrame parent, PatientData thePatient, int
        index)
        public boolean action(Event evt, Object obj)

class DisplayRecordFrame extends Frame                  //Generic frame to display a record
        private IDD_DisplayRecord displayRecordDialog = null;
        public PatientFileFrame myparent = null;
        private PatientData myList;
        private String fileURLString, theFileName, theCreationDate,
                thePractitioner, theInstitution;
        private String [] commentArray;
        private DataInputStream in = null;
        private URL textURL, fileURL = null;
        private Record theRecord = null;
        public DisplayRecordFrame(PatientFileFrame parent, Record theRecord)
        public boolean action(Event evt, Object obj)
```

```
class FilePanicWindow extends Frame                      //Frame to display an
                                                         error
        private IDD_PanicWindow panicWindow= null;       //message in a
        private PatientFileFrame myparent = null;        //PatientFileFrame
        public PanicWindow(PatientFileFrame parent)
        public boolean action(Event evt, Object obj)

class ListPanicWindow extends Frame                      //Frame to display an
                                                         error
        private IDD_PanicWindow panicWindow= null;       //message in a
        private PatientListFrame myparent = null;        //PatientListFrame
        public ListPanicWindow(PatientListFrame parent)
        public boolean action(Event evt, Object obj)

class ImageFrame extends Frame                           //Frame to display Image
        private IDD_ImageFrame imageFrame= null;
        private DisplayRecordFrame myparent = null;
        private Graphics g = null;
        private Image myImage = null;
        public ImageFrame(DisplayRecordFrame parent, URL theImageURL)
        public boolean action(Event evt, Object obj)
        public void paint(Graphics g)

class TextFrame extends Frame                            //Frame to display Text
        private IDD_TextFrame textFrame= null;
        private DisplayRecordFrame myparent = null;
        private Graphics g = null;
        private String [] myText = null;
        public TextFrame(DisplayRecordFrame parent, URL theTextURL)
        public boolean action(Event evt, Object obj)
        public void paint(Graphics g)
```

Appendix C

User Test Questionnaire

User Interface Evaluation Document

QB:

A Multimedia Medical Information System

Gerard Dunphy, P. Eng.

March 23, 1999

Introduction

Thank you for taking the time to assist in the evaluation of this software.

The purpose of this document is to describe the evaluation process for Version 1.0 of QB, a prototype multimedia medical information system. Contained are a description of the background and motivation for the development of QB, a description of the program, instructions for accessing and using the program, and an evaluation form to record your thoughts. The results of this evaluation will be used to refine the design of QB in a more fully functional version.

QB Version 1.0 is designed as a "proof of concept" prototype of a medical information system that could be used as a hospital information system, storing all information regarding an institution's patients. It is designed to be used across the World Wide Web (WWW), and to be accessible from any off the shelf Web browser.

Background

Considerable research has been conducted regarding the use of information systems in the storage and manipulation of patient information. Many institutions already have a text based system to record patient information. Likewise, several systems for the manipulation of images such as X-rays, MRI, CT, and ultrasound have been implemented. In general, though, implementation of these systems is limited to a single institution such as a hospital or large clinic.

It was felt that a medical information system which could be accessed from any location, and with a non-proprietary user interface, might be able to assist in the demands of remote diagnosis/evaluation and telemedicine, while at the same time addressing the needs of a local institution for an integrated information system that would store not only text, but audio, image, and video information - in other words, all information which might be obtained from the various diagnostic modalities contained in a typical medical environment. In this manner, an **integrated patient record** could be created, whereby all information collected on an individual from "cradle to grave" could eventually be accessible from a single location.

Instructions

You are requested to examine this software from the point of view of the big picture - not in terms of the specifics of the current implementation, but rather in terms of the viability of the concept. Please consider how useful an application of this type would be, what could be implemented to make it more useful, and what you would need in the execution of your daily routine to make a system of this type a practical tool.

Please go through the specific tasks described on the following pages. Note that many functions you see on the screen may not be implemented, so behavior of the software may be unpredictable if you do not follow the instructions exactly. If something strange happens, close all the application windows (using the "Cancel" button) and reload the page.

Evaluation Form

Evaluator: ____________________

Date: ____________________

Overall Impressions (Please complete this section AFTER completing the Specific Tasks)

Is the user interface easy to understand? Yes ____ No ____

Is this application a useful concept? Yes ____ No ____

Is the user interface intuitive? Yes ____ No ____

Would this type of application be useful in your daily routine? Yes ____ No ____
Please explain:

____

____

Do you think the ability to display video (e.g. ultrasound) would be useful in this application? Yes ____ No ____
Please explain:

____

____

Do you think the ability to conduct consultation with another professional would be useful, e.g. videoconferencing? Yes ____ No ____
Please explain:

____

____

Please describe other functions you feel a system of this nature should possess:

____

____

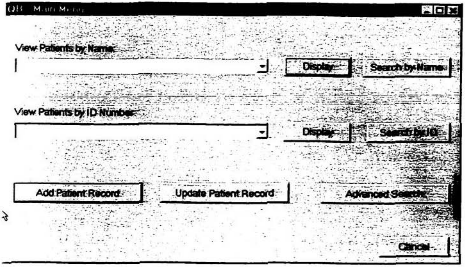
____

____

Specific Tasks

Task 1: Loading the program

1. Start your Internet browser. You must be connected to the Internet to access QB.
2. Enter the URL: http://www.engr.mun.ca/~gdunphy/qb/QB.html (Note: the URL is case-sensitive).
3. When the program has finished loading, you will see a window like this one:



Did the window take a long time to load?	Yes ____	No ____
Did you consider this too long?	Yes ____	No ____
Approximately how long would you consider to be too long?	____ seconds minutes (Please circle one)	
Is the function of this screen apparent?	Yes ____	No ____

Please explain:

____

____

Are the functions of the controls clearly understandable?	Yes ____	No ____

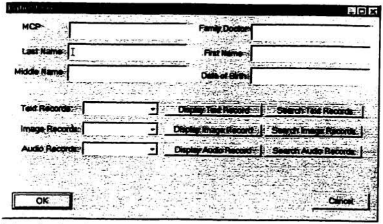
Please explain:

____

____

Task 2: Displaying a Patient's Record

1. The list of patients can be seen in the dropdown list below the heading "View Patients by Name:". Press the arrow to the right of the name shown to see the entire list. Right now, leave "John Doe" in the window and press the button labelled "Display" immediately to the right of the list. Note that pressing any other button will close the window. If this happens, press the "reload" button on your browser. If you press the correct key, you will see a window like the following:



Did the window take a long time to load? Yes _____ No _____

Did you consider this too long? Yes _____ No _____

Approximately how long would you consider to be too long? _____ seconds minutes (Please circle one)

Is the function of this screen apparent? Yes _____ No _____

Please elaborate:

____

____

Are the functions of the controls clearly understandable? Yes _____ No _____

Please elaborate:

____

____

Is there information that should be on this screen but is not? Yes _____ No _____

What should be included that is not here?

____

____

Is there redundant information here? Yes ____ No ____

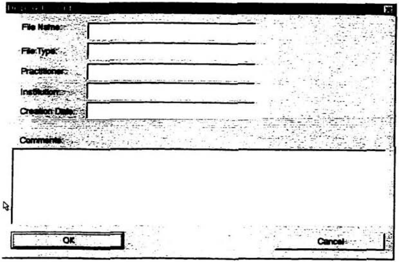
What information can be removed?

____

____

Task 3: Retrieving an image

1. In this application, an image and its associated text data (date of creation, etc.) are displayed separately. First, you must bring up a window that gives information regarding an image; then, the image itself can be displayed. Choose an image from the dropdown box to the right of the label "Image:" and press the button labelled "Display Image Record". You should see a window similar to this one:



2. Now the image can be loaded. Press the "Display Image" button. You will see a window with a medical image in it. To close the image window press the "Cancel" button.

Did the window take a long time to load?	Yes _____	No _____
Did you consider this too long?	Yes _____	No _____
Approximately how long would you consider to be too long?	_____ seconds minutes (Please circle one)	
Is the function of the "Display Record" window apparent?	Yes _____	No _____

Please elaborate:

____

____

Are the functions of the controls clearly understandable?	Yes _____	No _____

Please elaborate:

____

____

Is there information that should be on either window but is not? Yes _____ No _____

What should be included that is not here?

____

____

Is there redundant information here? Yes _____ No _____

What information can be removed?

____

____

Task 4: Listening to Audio

1. Close the image window by pressing the "Cancel" button. Return to the Patient Information window by pressing "Cancel" in the "Display Record" window.

2. Retrieve an audio file by repeating the steps in Task 3, except choose a file from the list next to the label "Audio" and press the "Display Audio Record" button.

3. When the Audio Record information window is displayed, press the "Listen to Audio" button to hear the audio file. Depending on the file chosen, you may hear a short piece of music or a quick quotation from the cartoon "The Jetsons". Do not take the instructions you may hear literally! Note that you will not see another window.

Is the ability to listen to audio (e.g. voice recordings of diagnosis, recordings of heartbeats, etc. useful? Yes ____ No ____

What type of information could be recorded?

____

____

Is there information that should be in this window but is not? Yes ____ No ____

What should be included that is not here?

____

____

Is there redundant information here? Yes ____ No ____

What information can be removed?

____

____

When you are finished, simply close all windows using the "Cancel" button. When all windows are closed and you are back to your browser screen, the program has been terminated.

Thank you for your input!. Please complete the "Overall Impressions" section of this document when you are finished.